

IMVIP 2026

IRISH MACHINE VISION AND IMAGE
PROCESSING CONFERENCE



26th August – 28th August 2026

Maynooth University,
Maynooth, Co. Kildare, Ireland.

Proceedings of the Irish Machine Vision and Image Processing Conference 2026

26th to 28th August 2026
Maynooth University, Kildare, Ireland

Organised by:
John McDonald, Rozenn Dahyot, Charles Markham and Gerry Lacey

Published by the Irish Pattern Recognition & Classification Society
iprcs.org
ISBN 978-1-9195641-0-4
©2026

This work is distributed free of charge by the Irish Pattern Recognition & Classification Society on behalf of the Irish Machine Vision & Image Processing Conference, and the contributing authors to this conference. Both organizers and authors own the rights of their contribution to this book.

Preface

The Organising Committee extends a warm welcome to all speakers and delegates of the 2026 Irish Machine Vision and Image Processing Conference (IMVIP 2026). This year it is hosted at the Maynooth University under the organization of the department of Computer Science. The IMVIP Conference is Ireland's primary meeting for those researching in the fields of machine vision and image processing. The conference has been running since 1997 and provides a forum for the exchange of ideas and the presentation of research conducted in Ireland and worldwide. IMVIP is a single track conference consisting of high quality previously unpublished contributed papers focusing on both theoretical research and practical experiences in all areas. After a rigorous review process, 35 papers were selected for presentation at the conference. We wish to sincerely thank the members of the Programme Committee for generously giving their time, effort and expertise in reviewing the submissions, and the session chairs for their contributions to this event.

Continuing the tradition of inviting high-profile speakers to IMVIP, we are delighted to have two high-profile speakers give keynote talks: Prof. Binh-Son Hua from Trinity College Dublin, with a talk entitled *"Toward a Synergy of Generative AI and Computer Graphics"*, and Prof. Ganesh Sistu from Valeo and University of Limerick with a talk entitled *"Evolution of Autonomy Stacks: From Rules to Physical AI"*.

IMVIP 2026 is run in association with the Irish Pattern Recognition and Classification Society (IPRCS), a member of the International Association for Pattern Recognition (IAPR) and the International Federation of Classification Societies (IFCS). We thank IPRCS for its support in organizing IMVIP 2026, its travel bursaries and paper awards.

We warmly thank Prof. Neil Trappe, head of Department of Physics at Maynooth University, for organizing a very instructive and entertaining visit to the [National Science Museum at Maynooth](#) for all conference attendees. Finally, IMVIP 2026 would not have been possible without the incredible support and know-how of Mrs Phil Dully in the Department of Computer Science at Maynooth University.

Prof. John McDonald
Prof. Rozenn Dahyot
Prof. Charles Markham
Prof. Gerry Lacey

Maynooth University, Kildare, Ireland
September 2026

Acknowledgments

The Microsoft CMT service was used for managing the peer-reviewing process for this conference. This service was provided for free by Microsoft and they bore all expenses, including costs for Azure cloud services as well as for software development and support.

Committee Members

Organising Committee:

Prof. John McDonald
 Prof. Charles Markham
 Prof. Rozenn Dahyot
 Prof. Gerry Lacey

Programme Committee:

Aidan Murphy, University College Dublin
 Brian Mac Namee, University College Dublin
 Bryan Scotney, Ulster University
 Derek Molloy, Dublin City University
 Dermot Kerr, Ulster University
 Francois Pitie, Trinity College Dublin
 Ganesh Sistu, Valeo Vision Systems
 George Moore, Ulster University
 Hideo Saito, Keio University
 Jane Courtney, TU Dublin
 John McDonald, Maynooth University
 Katterine N Rios Peralta, University College Dublin
 Kislay Raj, Dublin City University
 Mark Halton, University of Limerick
 Michael Manzke, Trinity College Dublin
 Michael Schukat, University of Galway
 Niall McLaughlin, Queen's University Belfast
 Richard Gault, Queen's University, Belfast
 Roshan George, University of Galway
 Rozenn Dahyot, Maynooth University
 Ruairi D O'Reilly, Cork Institute of Technology
 Sean Mullery, Atlantic Technological University
 Shane Gilroy, Atlantic Technological University
 Sonya Coleman, School of Computing and Intelligent Systems, University of Ulster
 Soumyabrata Dev, University College Dublin
 Tony Scanlan, University of Limerick
 Victoria M.A. Porter, Queen's University, Belfast
 Vladimir Krylov, Dublin City University
 Zi Ye, Maynooth University

Session chairs:

Rozenn Dahyot, Maynooth University
 Gerry Lacey, Maynooth University
 Ken Dawson-Howe, Trinity College Dublin
 Sonya Coleman, Ulster University
 John McDonald, Maynooth University

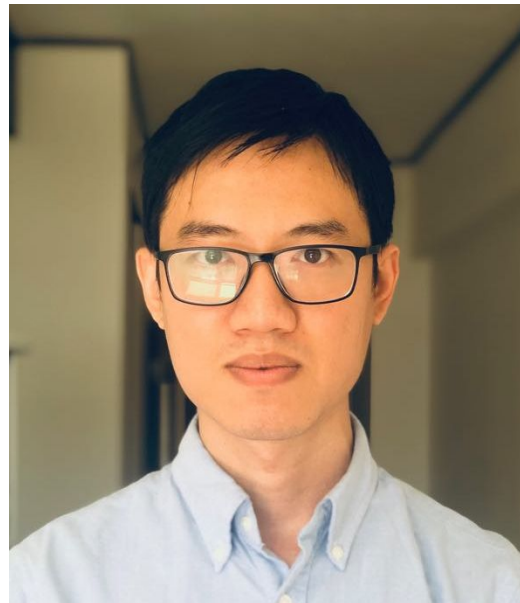
Charles Markham, Maynooth University
Dermot Kerr, Maynooth University
Richard Gault, Queen's University Belfast
Bryan Scotney, Ulster University

Keynote Speaker: Prof. Binh-Son Hua

Title: Toward a Synergy of Generative AI and Computer Graphics

Abstract: The field of computer graphics is changing fast in the era of generative AI. New generative modeling techniques are transforming multiple stages of the graphics pipeline, from image and video synthesis to 3D content generation, stylization, and material editing. This keynote will present our recent work in generative graphics, focusing on 3D generation and appearance control. We will then examine the evolving role of generative AI in computer graphics and explore the prospect of physically based generative rendering through light transport simulation in autoencoder latent spaces.

Biography: Binh-Son Hua is an Assistant Professor of Computer Science at Trinity College Dublin. His research interests are at the intersection of computer graphics, computer vision, and machine learning with a focus on generative AI in the 3D space, particularly on 3D content generation and rendering using generative AI techniques. He received his PhD in Computer Science from National University of Singapore, and spent his postdoctoral research at The University of Tokyo, Japan and Singapore University of Technology and Design. He was recognized with the Best Paper Honorable Mention award at International Conference on 3D Vision 2016, and as an Outstanding Reviewer at CVPR 2023. He has been actively serving as an area chair for CVPR, NeurIPS, BMVC, WACV, an associate editor for TVCG, a technical program committee member for SIGGRAPH Asia, and a senior program committee member for AAAI. His research is funded by Research Ireland under the Frontiers for the Future Programme.



Keynote Speaker: Prof. Ganesh Sistu

Title: Evolution of Autonomy Stacks: From Rules to Physical AI

Abstract: Autonomous driving has evolved from a collection of limited highway-assistance features, such as Lane Keeping Assist and Adaptive Cruise Control, to increasingly capable systems supporting urban driving, automated parking and eyes-off driving. A key driver of this transformation has been the evolution from traditional rule-based software built from millions of lines of explicitly engineered code to AI-driven architectures powered by billions of learned neural-network parameters. This shift is fundamentally changing how autonomous systems perceive, reason, plan, and act in the physical world. In this talk, Ganesh will provide a bird's-eye view of this evolution, tracing the journey from modular autonomy architectures and end-to-end systems to emerging Vision-Language-Action Models and Physical AI systems. The talk will explore how advances in perception, sensing, representation learning, and multimodal foundation models are coming together to create increasingly general and efficient autonomy stacks.

Biography: Ganesh Sistu is leading multiple global research teams focusing on automated driving and parking at Valeo Ireland. With over 15 years of experience in computer vision and machine learning, he has contributed to 40+ publications in top-tier conferences such as ICCV, ICRA, and WACV, garnering over 2,000 citations and 30+ US and EU patent filings. He also plays a pivotal role in guiding the future of AI education as an Adjunct Assistant Professor at the University of Limerick and as an industry board member for Research Ireland's Data Science PhD and the National MSc in AI programs at the University of Limerick, blending academic insight with industry expertise.



Contents

Preface	ii
Committee Members	iii
Keynote Speaker: Prof. Binh-Son Hua	v
Keynote Speaker: Prof. Ganesh Sistu	vi
Contents	vii

Medical Image Analysis

Compact CNNs for Chest X-ray Screening: Accuracy, Explainability, and Containerised Deployment, Christopher Anich (CCT College Dublin), Kislay Raj (Dublin City University)	1
Motion-Based Detection and Tracking of Trypanosoma cruzi Parasites in Microscopy Videos, Senem Aktas (Bahcesehir University), Carlos Brito-Loeza (Universidad Autónoma de Yucatán), Hugo Ruiz-Piña (Centro de Investigaciones Regionales Dr. Hideyo Noguchi), Lavdie Rada (Bahcesehir University)	9
Robust Breast Ultrasound Classification Through Dataset Auditing, Preprocessing Standardization, and Ensemble Fusion, Sana Siddique (Bahçeşehir University), Lavdie Rada (Bahcesehir University)	17

Machine Learning Techniques and Applications I

MicroBird: Embedded Bird-Audio Classification with Microjoule-Range CNN Inference on the MAX78002, Justyna Przyborska (University of Limerick), Salaheddin Alakkari (University of Limerick) - IMVIP 2026 Best Student Paper Award	26
Towards Long-Term Pig Re-Identification: Investigating Human ReID Methods in Pigs, Yiyuan Wang (Queen's University Belfast), Ilias Kyriazakis (Queen's University), Jian Gao (Queen's University Belfast), Niall McLaughlin (Queen's University Belfast)	34
Model Compression for Very Short-Term Solar Irradiance Forecasting Using Infrared Sky Images, Ifran Rahman Nijhum (University College Dublin), Arnav Jaitly (University College Dublin), David Lillis (University College Dublin), Soumyabrata Dev (Trinity College Dublin)	42
Comparative Study of Parameter-Efficient Fine-Tuning on Low-Resource Vision-Language Modelling: Burmese-CLIP, Htiike Thu Kaung (University of Limerick) & Salaheddin Alakkari (University of Limerick)	50
Perspective-Aware Billboard Corner Detection with a Web-Based End-to-End System, Sukriti Dhang (University College Dublin), Tingyu Zhang (University College Dublin), Mimi Zhang (Trinity College Dublin), Soumyabrata Dev (Trinity College Dublin)	58

Aerial and Geospatial Image Processing

Automask: Promptable Concept Segmentation with SAM3 for Glacier 3D Reconstruction from UAV Imagery , SiCheng Zhao (University College Dublin), Matthew Byrne (University College Dublin), Soumyabrata Dev (Trinity College Dublin)	66
Low-Data RGB-to-NIR Translation with Augmentation and Pseudo-Label Learning , Md Yearat Hossain (University College Dublin), Arpit Mirdha (University College Dublin), Riju Das (University College Dublin), Gavin McArdle (University College Dublin), Soumyabrata Dev (Trinity College Dublin) ...	74
Geo-referenced small object detection in oblique aerial imagery , Paolo Moloney (Maynooth University), Aidan Magee (Maynooth University), Daire Walsh (Maynooth University), Tim McCarthy (Maynooth University), John McDonald (Maynooth University)	82
Similarity-Augmented Roadside Object Triangulation from street level images , Waqar Ahmad (Dublin City University), Vladimir Krylov (Dublin City University) - IMVIP 2026 Jonathan Campbell Best Paper Award	90

Posters

Efficient automatic traffic light alteration for sparse traffic flow , Afif Imtiyaz Khan Deshmukh (Trinity College), Kenneth Dawson-Howe (Trinity College)	98
Automatic Passing Vehicle to Cyclist Distance Estimation , Steven Bondaruk (Trinity College), Kenneth Dawson-Howe (Trinity College).....	102
A Computer Vision Tool to Aid Beginner Electric Bass Players with Fretboard Navigation , Zemyna Kulikausk (Trinity College), Kenneth Dawson-Howe (Trinity College)	106
A Hybrid Deployment Framework for Industrial Quality Control Using GelSight Tactile Sensing , Kaviyashree Damodaran (Ulster University), Antonio Gambale (Ulster University), Dermot Kerr (Ulster University), Sonya Coleman (Ulster University)	110
Parameter-Efficient Dense Prediction for Cross-View Geo-Localisation in Dense Urban Environments , Lucas Scott (Ulster University), Sonya Coleman (Ulster University), Dermot Kerr (Ulster University), Philip Vance (Ulster University), Yunzhou Zhang (Ulster University)	117
Transformers Cross-Attention Audio-Visual Fusion , Lucas Hindes (Ulster University), Nazmul Siddique (Ulster University), Philip Vance (Ulster University), Sonya Coleman (Ulster University)	124
WeldNet: Vision-Based Weld Inspection for Intelligent Manufacturing Systems , Sushil Sharma (University of Limerick), Oran Connors (University of Limerick), Reenu Mohandas (University of Limerick), Joe Brennan (OMC Technologies Limited), Ciarán Eising (University of Limerick)	132
Enhancing an Emotion-Labelled Video Dataset with Pose Estimation Key-points for Face, Body and Hands , Ciara Sempey (Ulster University), Bryan Scotney (Ulster University), Shuai Zhang (Ulster University), Matthew Burns (Ulster University), Jorge Martinez (Ulster University)	136

3D Vision, Robotics, and Autonomous System I

Bundled Feature Encoding: A Shift-Tolerant Image Representation for Hyperdimensional Computing , C.T. Levins (Ulster University), D. Kerr (Ulster University), S. Coleman (Ulster University), J.P. Quinn (Ulster University)	144
---	-----

KeyJoint-STGCN: Keypoint-Based Spatio-Temporal Graph Learning for Monocular Trajectory Prediction , Patitapaban Palo (Valeo), Satyajit Nayak (Valeo) , Kwanit Gupta (Valeo), Satarupa Uttarkabat (Valeo), Ganesh Sistu (Valeo), Mohamed Tarek (University Limerick)	152
Multi-session Online Open-Vocabulary Object Mapping , Frank Gallagher (Maynooth University), Marco Cognetti (CNRS), John McDonald (Maynooth University)	160
Towards ISP-Free Fisheye Camera Perception: An Evaluation in Image and BEV Space , Krishna Prasath Kizhpada Vasudevan (University of Limerick), Ganesh Sistu (Valeo Vision Systems), Ciarán Eising (University of Limerick), Venkatesh Satagopan (Valeo), Mihai Ilie (Valeo), William O'Grady (Valeo), Praveen Narasappareddygar (Valeo), and Hariharan Ragunathan (Valeo)	168
A Domain and Sensor-Adaptive Metric Road Width Estimation from Dual Rear Cameras via Transfer Learning , Zuha Hussain (University of Galway), Waqar Shahid Qureshi (University of Galway), Nazre Batool (University of Galway), Syed Ibrahim Hassan (Pavement Management Services Ltd), David Power (Pavement Management Services Ltd), Kieran Feighan (Pavement Management Services Ltd)	176

3D Vision, Robotics, and Autonomous System II

SCULPT: Training Edge Vision Models for Post-Training Quantization Readiness , Bharadwaj Kavuri (Valeo), Sourav Babu-PK (Valeo), Varadhraj Ellapan (Valeo), Pullarao Maddu (Valeo), Prasad Deshpande (Valeo Vision Systems)	184
Pix2Repair: Implicit Shape Restoration from Images , Nikolas Lamb (Clarkson University), Xinchao Song (Rochester Institute of Technology), Natasha Kholgade Banerjee (Wright State University), Sean Banerjee (Wright State University)	192
ARAS-FCWNet: Low-cost and Light-weight Model for Forward Collision Warning (FCW) System in Unstructured ARAS , Satyajit Nayak (Valeo India Pvt. Ltd.), Lakshman-Prabhu Balasubiramanian (Valeo India Private Limited), Rakshita Joshi (Valeo), Ganesh Sistu (Valeo Vision Systems)	200
Vision-Based Robotic Hot-Water Weeding for Precision Agriculture , Gregory Mantel (Maynooth University), James Florin Petri (Maynooth University), Edward Jakunskas (Maynooth University), Gerard Lacey (Maynooth University)	208
Training-Free Geometric Stabilization for Monocular 3D Detection in Advanced Rider-Assistance Systems , Lakshman Prabhu Balasubiramanian (Valeo India Private Limited), Satyajit Nayak (Valeo India Pvt. Ltd.), Cyril Poulet (Valeo), Ganesh Sistu (Valeo Vision Systems)	216

Machine Learning Techniques and Applications II

Diffusion-Based Saliency Map Generation from RGB Images , Mark Langtry (University of Limerick), Oisin O'Sullivan (University of Limerick), Salaheddin Alakkari (University of Limerick)	224
Thumbnail Guided Tissue Compaction (TGTC) - Whole Slide Image Preprocessing for Lossless Image Compression , Christopher Smith (Queen's University Belfast), Leo Galway (Queen's University Belfast), Richard Gault (Queen's University Belfast)	232
Body Mass Index (BMI) estimation from a single frontal face image using a Bayesian-ViT architecture , Carlos Naranjo (Universidad Adolfo Ibañez), Francisca Pérez (Universidad Adolfo Ibañez), Rodrigo Wagner (Universidad Adolfo Ibañez), Claudia Arellano (Universidad Adolfo Ibañez)	240
Are Action Units Priors or Pathways? A Diagnostic Study of Temporal Valence-Arousal Estimation , Riju Das (University College Dublin), Soumyabrata Dev (Trinity College Dublin)	248
Parameter-Efficient Adaptation of V-JEPA 2 for Egocentric Action Anticipation , Ayoub Sahlaoui (Sidi Mohamed Ben Abdellah University), Salaheddin Alakkari (University of Limerick), Younes Dhassi (Sidi Mohamed Ben Abdellah University)	256

Compact CNNs for Chest X-ray Screening: Accuracy, Explainability, and Containerised Deployment

Christopher Anich¹, Kislay Raj²

¹*CCT College Dublin, Ireland*

²*School of Computing, Dublin City University, Ireland*

Abstract

Lightweight convolutional networks are widely promoted for medical-image triage on resource-constrained hardware, on the assumption that smaller models translate into faster, cheaper deployment. We test this in a pilot study on a three-class chest X-ray task (Normal, Pneumonia, Tuberculosis) assembled from four public sources, with patient-grouped splits verified disjoint by path, content hash, patient group and pairwise pixel correlation. A custom 6,627-parameter CNN (*Victorio*) is compared against MobileNetV2, EfficientNet-B0 and ResNet-50 under two initialisation regimes on a held-out test set of 480 images. The three backbones are statistically indistinguishable at 97.3–98.3% accuracy across a tenfold parameter range, while *Victorio* reaches 84.2%, a 14 pp gap far outside the ± 1.6 pp resolution of the test set; pretraining helps every backbone by 1.7 to 6.6 pp. A Grad-CAM versus LIME Agreement Index, computed from raw attribution arrays and stable under three mask thresholds, ranges from 0.13 to 0.24 and varies with architecture and regime rather than being invariant to capacity. A source classifier recovers the acquisition dataset within a fixed diagnosis at 0.98 balanced accuracy, confirming a source confound; performance nonetheless holds on a single-source subset. A deployment benchmark (5,040 requests) exposes an architecture-dependent CPU/GPU crossover: GPU serves ResNet-50 at $2.1\times$ the CPU median, ties on MobileNetV2, and loses to CPU by $2.5\times$ on the compact network.

Keywords: Chest X-ray, CNN, Transfer learning, Explainability, Kubernetes deployment.

1 Introduction

Chest radiography remains the most common front-line imaging modality for respiratory disease, and convolutional neural networks (CNNs) routinely match or exceed expert performance on curated benchmarks [Rajpurkar et al., 2017]. Lightweight architectures such as MobileNetV2 [Sandler et al., 2018] and EfficientNet [Tan and Le, 2019] are increasingly favoured for clinical deployment, with the implicit claim that fewer parameters translate into faster, cheaper inference. In practice, end-to-end response time in a containerised service depends on preprocessing, HTTP, orchestration and model compute combined; the parameter-count advantage of a minimal network can be absorbed by the surrounding stack.

This paper is a pilot study addressing two hypotheses: **(H1)** that an ultra-compact custom CNN matches transfer-learning baselines on a balanced three-class CXR task; and **(H2)** that the CPU/GPU latency advantage is architecture-dependent, with GPU acceleration amortising its per-request launch overhead on heavier backbones but losing that advantage on compact models. We make four contributions:

- A three-class CXR corpus from four public sources with patient-grouped splits, verified disjoint at four levels including pairwise pixel correlation over all 5.17 M cross-split pairs.
- A capacity comparison of a 6,627-parameter custom CNN against backbones spanning 2.2 M to 23.5 M parameters, with confidence intervals so the resolution limit of the test set is explicit.
- A source-confound analysis: a within-class pixel signature test, a source-discriminating classifier, and an evaluation on a single-source subset in which the confound cannot operate.

- A latency characterisation of the deployed service on a local Kubernetes cluster (5,040 requests, CPU and GPU pods).

2 Related Work

CNNs for chest radiography. Modern CXR pipelines almost universally rely on backbones pretrained on ImageNet, with reported AUROC values approaching radiologist performance for pneumonia and tuberculosis screening [Rajpurkar et al., 2017; Irvin et al., 2019; Liu et al., 2020]. Whether natural-image pretraining is the right prior for radiology is contested: RadImageNet [Mei et al., 2022] shows that pretraining on radiological images can outperform ImageNet initialisation on medical tasks, and comparisons of compact backbones on multi-label CXR classification [Ucan et al., 2025] report similar accuracy across the EfficientNet family, consistent with the capacity saturation we observe. CheXpert [Irvin et al., 2019] and TBX11K [Liu et al., 2020] are typically studied in isolation; cross-dataset fusion introduces source-class confounds [Zech et al., 2018] that must be measured rather than asserted away. Other work on medical-imaging CNN training addresses a different configuration axis: Parise and Mac Namee evaluate representation, model capacity and query strategy jointly, optimising labelling efficiency rather than the deployment characteristics studied here [Parise and Mac Namee, 2023].

Lightweight architectures and serving. Inverted residuals [Sandler et al., 2018] and compound scaling [Tan and Le, 2019] reduce parameter count by an order of magnitude while preserving accuracy on natural images, but reported improvements usually refer to model FLOPs rather than end-to-end response times. Serving systems such as Clockwork [Gujarati et al., 2020] target predictable per-request latency under containerised deployment, without decomposing it by architecture and backend at the granularity relevant to single-image medical inference.

Explainability. Grad-CAM [Selvaraju et al., 2017] and LIME [Ribeiro et al., 2016] are the two most common post-hoc attribution methods in radiology research. Quantitative comparisons between them are rare, and Adebayo et al. [2018] show that saliency methods can produce plausible maps that are insensitive to the model being explained, so any agreement metric requires a sanity check before it can be interpreted.

3 Materials and Methods

3.1 Dataset construction

The corpus draws on *four* public sources. The TBX11K release [Liu et al., 2020] bundles three further tuberculosis datasets under `imgs/extra/`: Montgomery County and Shenzhen [Jaeger et al., 2014], and DA and DB [Chauhan et al., 2014]. Table 1 gives the composition. Pneumonia is drawn from CheXpert-Small [Irvin et al., 2019] frontal views. The corpus is balanced at 1,072 images per class, 3,216 in total, split 2,255 / 481 / 480 into training, validation and test, with the test set balanced at 160 per class.

Class	TBX11K	Montgomery/Shenzhen	DA/DB	CheXpert
Normal	985	57	30	—
Tuberculosis	800	198	74	—
Pneumonia	—	—	—	1,072

Table 1: Corpus composition by class and acquisition source, 1,072 images per class. Pneumonia originates from a single source, which is the basis of the confound analysed in Section 5.3.

Class definition. The tuberculosis class comprises radiographs carrying any TB annotation. TBX11K distinguishes active from obsolete (healed, inactive) disease: of the 1,072 TB images, 58.8% are annotated active, 13.0% obsolete, 2.8% both, and 25.5% come from sources that record TB positivity without activity sub-classification. Reported tuberculosis recall is therefore sensitivity to TB-associated radiographic findings, not to active infection. Figure 1 shows representative radiographs from each class.

Split integrity. CheXpert is split by patient (901 patients, 1.19 images each); the remaining sources contribute one image per subject. Splits were verified disjoint by file path, SHA-256 content hash, patient group, and

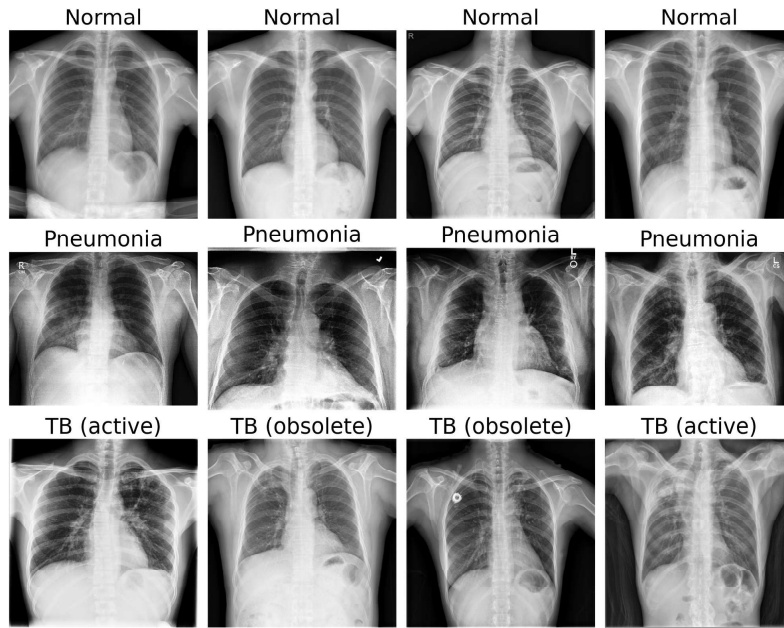


Figure 1: Representative radiographs from the test split, four per class. The tuberculosis panels are annotated with their activity label, illustrating that the class contains both active and healed disease.

pairwise Pearson correlation of z-scored 64×64 images over all 5,169,720 cross-split pairs. The maximum cross-split similarity is 0.9722, against approximately 0.9999 for a re-encoded duplicate, and the pair concerned is retained and disclosed rather than removed, so the test set remains exactly reproducible from the build script.

3.2 Architectures

MobileNetV2 [Sandler et al., 2018] (2,227,715 parameters), EfficientNet-B0 [Tan and Le, 2019] (4,011,391) and ResNet-50 [He et al., 2016] (23,514,179) were used with the final classification head replaced by a 3-way linear layer; counts are for the 3-class configuration rather than the 1000-class ImageNet original. These three are referred to as the backbones throughout; Victorio is treated separately. **Victorio** is a custom minimal CNN of 6,627 parameters: two 3×3 convolutional blocks ($3 \rightarrow 16 \rightarrow 32$ channels), each followed by ReLU and 2×2 max-pooling, adaptive average pooling to 4×4 , and a 3-way linear head, with no batch normalisation or dropout. The design intentionally explores the lower bound of CNN capacity for this task.

3.3 Training regimes

Each architecture was trained under two regimes. The **ImageNet** regime initialises the three backbones with ImageNet weights and uses ImageNet normalisation. The **CXR** regime initialises from random weights and applies fixed dataset-specific normalisation (per-channel mean 0.534, standard deviation 0.266), applied identically to training, validation and test. Images are single-channel grayscale replicated across three RGB channels.

The two quantities a regime fixes, initialisation and normalisation, are logically separate and coincide only where pretrained weights exist. No ImageNet weights exist for Victorio, so both of its runs begin from random initialisation and differ in normalisation alone. Its regime comparison therefore answers a different question from that of the three backbones, and its rows in Table 2 must not be read as evidence about pretraining. A run manifest recording the initialisation and normalisation actually used is written alongside every checkpoint.

4 Experimental Setup

4.1 Training protocol

All models were trained with identical hyperparameters: input 256×256 , batch size 128, 60 epochs, AdamW ($lr = 10^{-4}$, weight decay 10^{-4}), cosine schedule with 3-epoch warm-up, label smoothing 0.05, mixed precision, early stopping (patience 8). Augmentation was random horizontal flip only. Resizing used aspect-

ratio-preserving letterbox padding to avoid clipping costophrenic angles. All trainable parameters were updated end-to-end. Best-validation checkpoints were retained. Using one hyperparameter set across architectures spanning four orders of magnitude in capacity is a deliberate control for fairness, but it disadvantages models whose optimum lies elsewhere; this is revisited in Section 6.

4.2 Evaluation and explainability

Accuracy, per-class precision, recall and F1, and one-vs-rest macro AUROC were computed on the held-out test set ($n = 480$). Every proportion carries a Wilson score interval, preferred over the normal approximation because the values lie close to 1.0 with small n . The interval half-width is approximately ± 1.6 pp overall and ± 2.9 pp per class at the observed accuracies, so differences below roughly 3 pp are not resolvable.

Grad-CAM [Selvaraju et al., 2017] maps were taken as the raw normalised activation array of the last activated convolutional block; the mask retains the highest-activating 20% of pixels by rank. LIME [Ribeiro et al., 2016] explanations used 1,000 perturbations per image, and the mask is the union of superpixels with positive fitted weight, read from the explanation object. Neither mask is derived from a rendered figure. The Agreement Index is the intersection over union of the two masks, computed over 20 test images per class for each of the eight model–regime pairs, retaining those the model classifies correctly, and repeated at mask thresholds of 10%, 20% and 30% to test threshold dependence.

4.3 Deployment protocol

Each checkpoint was packaged in a containerised FastAPI service with a single `/predict` POST endpoint, all eight served from one pod via a model query parameter. CPU (`python:3.10-slim`) and GPU (`nvidia/cuda:12.8.0-runtime-ubuntu22.04`) images were built with Podman and deployed as separate Pods on a local Minikube cluster (Docker driver, 6 vCPU, 16 GB RAM; GPU pod on a host-attached NVIDIA RTX 5090 via the device plugin). For each (backend, model, image) triple, 5 warm-up requests followed by 100 measured requests were issued *sequentially*, that is at concurrency one: exactly one request is in flight at any moment, so the measurement isolates per-request latency and excludes queuing effects. Latency was the client-side wall-clock HTTP round trip, reported as median and tail percentiles in line with established practice [Dean and Barroso, 2013]. The data set totals 5,040 requests with no failures.

5 Results

5.1 Classification performance

Table 2 reports test-set metrics. The three backbones cluster at 97.3–98.3% under the ImageNet regime; every pairwise difference between them has a confidence interval spanning zero, so they are statistically indistinguishable despite a tenfold range in parameter count. Victorio reaches 84.2%, a gap of 14.1 pp to the best backbone, far outside the interval. Hypothesis H1 is not supported.

Pretraining improves accuracy for all three backbones, by 6.6 pp (MobileNetV2), 2.9 pp (ResNet-50) and 1.7 pp (EfficientNet-B0). The direction is consistent, but only the MobileNetV2 difference is clearly outside sampling error; the ResNet-50 difference is marginal and the EfficientNet-B0 difference is within it. The Victorio pair, which differs in normalisation alone, separates by 0.4 pp and serves as a negative control.

Per-class results expose an asymmetry, visible in Figure 2. Under the ImageNet regime all three backbones classify Pneumonia perfectly (160/160, precision and recall 1.000), and all residual errors fall on the TB/Normal boundary. Pneumonia is also the only class drawn from a single source, which motivates Section 5.3. Tuberculosis recall differs little between active and obsolete disease for the backbones (for example 0.979 and 1.000 respectively for MobileNetV2 under the ImageNet regime), though the 26 obsolete test cases support only an indicative estimate.

5.2 Explainability and Agreement Index

Table 3 reports the Grad-CAM versus LIME Agreement Index at the 20% mask threshold. The rank ordering across model–regime pairs is preserved at 10%, 20% and 30% (Spearman $\rho \geq 0.976$, $p < 0.001$), so the index reflects a property of the attributions rather than of the threshold.

Agreement is *not* invariant to capacity. It varies from 0.132 to 0.243, and the ImageNet regime yields higher

Model	Regime	Accuracy [95% CI]	Macro F1	AUROC	TB recall
ResNet-50	ImageNet	0.983 [0.967, 0.992]	0.983	0.999	0.988
MobileNetV2	ImageNet	0.981 [0.965, 0.990]	0.981	0.999	0.981
EfficientNet-B0	ImageNet	0.973 [0.954, 0.984]	0.973	0.998	0.944
EfficientNet-B0	CXR	0.956 [0.934, 0.971]	0.956	0.994	0.925
ResNet-50	CXR	0.954 [0.932, 0.970]	0.954	0.988	0.969
MobileNetV2	CXR	0.915 [0.886, 0.936]	0.915	0.981	0.881
Victorio	ImageNet	0.842 [0.806, 0.872]	0.839	0.933	0.694
Victorio	CXR	0.838 [0.802, 0.868]	0.836	0.935	0.706

Table 2: Test-set performance ($n = 480$, 160 per class) with Wilson score intervals. Rows are ordered by accuracy; the ordering is not a ranking, as differences below roughly 3 pp are not resolvable at this sample size. Victorio has no ImageNet weights, so its two rows compare normalisation only.

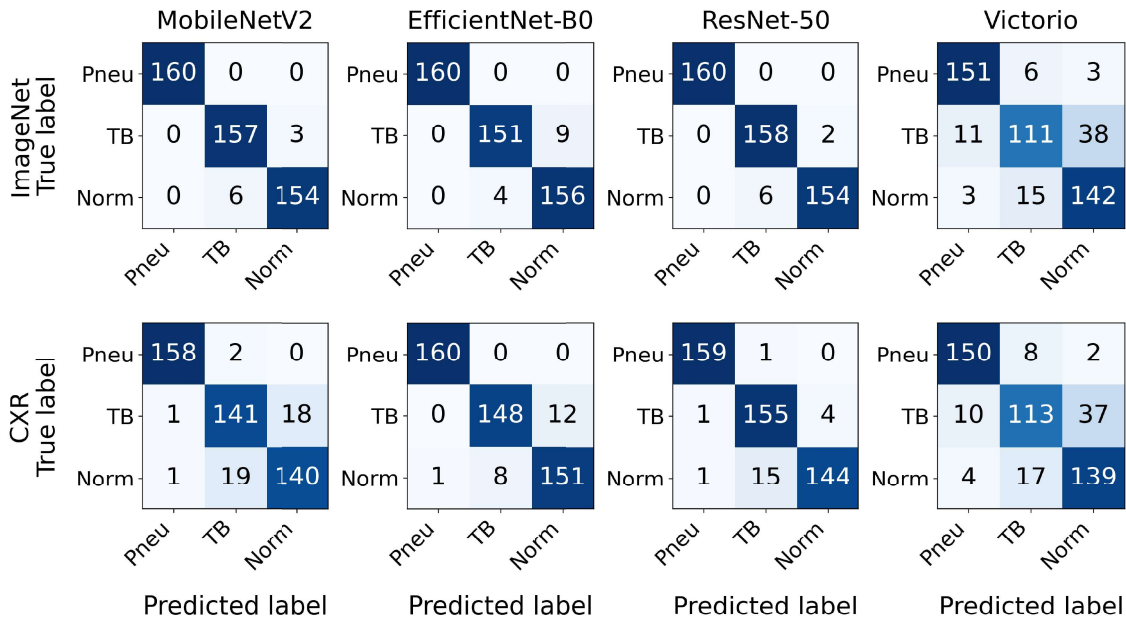


Figure 2: Confusion matrices on the held-out test set ($n = 480$). Top row: ImageNet regime; bottom row: CXR regime. Columns: MobileNetV2, EfficientNet-B0, ResNet-50, Victorio. Pneumonia is perfectly separated by all three backbones under the ImageNet regime, and residual errors concentrate on the TB/Normal boundary. Axis labels are abbreviated: Pneu (Pneumonia), Norm (Normal).

agreement than the CXR regime for every backbone. Victorio, whose two regimes differ only in normalisation, returns 0.143 in both, again acting as a negative control. Absolute values must be read against the mask areas: the Grad-CAM mask covers 20% of the image by construction while the LIME mask covers 55–85%, so the attainable maximum is the ratio of the two areas, approximately 0.29, rather than one. Agreement is therefore close to its ceiling for the pretrained backbones and materially below it for the compact network. By class, agreement is highest for Normal (0.199), then Pneumonia (0.187), then TB (0.155), consistent with focal tuberculosis lesions being a harder target for two methods to localise identically. Figure 3 shows both attributions for a representative case.

5.3 Source confound

Three measurements quantify the extent to which acquisition source, rather than pathology, is available to the models.

First, with the diagnosis held fixed, images sharing an acquisition source are more similar in raw pixels than images from different sources: within Normal, mean correlation 0.711 against 0.619 ($t = 153.1$); within TB, 0.604 against 0.574 ($t = 67.2$). Source leaves a signature independent of pathology.

Regime	MobileNetV2	EfficientNet-B0	ResNet-50	Victorio
ImageNet	0.216 ± 0.011	0.243 ± 0.011	0.199 ± 0.008	0.143 ± 0.014
CXR	0.178 ± 0.017	0.183 ± 0.011	0.132 ± 0.025	0.143 ± 0.013

Table 3: Grad-CAM versus LIME Agreement Index (IoU) at the 20% mask threshold, mean $\pm$ 95% confidence interval over correctly classified test images ($n = 52\text{--}60$ per cell). The attainable maximum is approximately 0.29 given the mask areas involved.

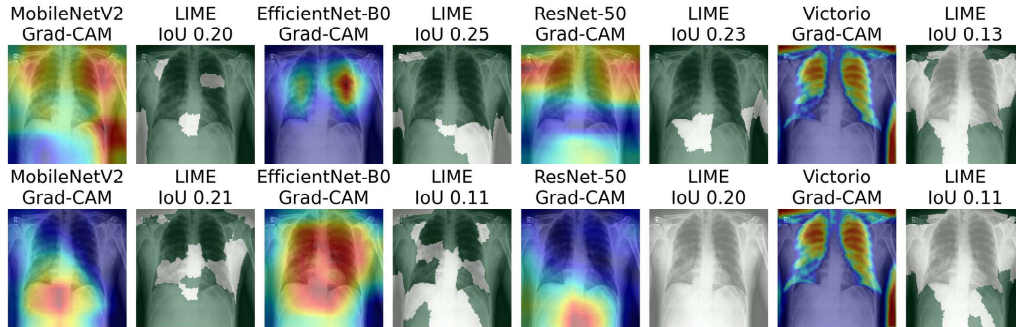


Figure 3: Grad-CAM and LIME on a representative Pneumonia test case. Top row: ImageNet regime; bottom row: CXR regime. Each pair of panels shows the Grad-CAM heatmap and the LIME positive-superpixel mask for the same case, with the Agreement Index for that case in the panel title.

Model	Regime	CPU pod		GPU pod	
		P50 (ms)	P95 (ms)	P50 (ms)	P95 (ms)
MobileNetV2	CXR	101	924	93	262
MobileNetV2	ImageNet	69	507	93	273
EfficientNet-B0	CXR	161	1010	110	300
EfficientNet-B0	ImageNet	188	1159	112	301
ResNet-50	CXR	517	1430	250	432
ResNet-50	ImageNet	398	1445	248	431
Victorio	CXR	19	113	48	204
Victorio	ImageNet	19	111	47	210

Table 4: Warm-state per-request latency at concurrency one (300 measured requests per row). Median (P50) and 95th percentile (P95) in milliseconds.

Second, a MobileNetV2 classifier trained to predict source rather than diagnosis separates CheXpert from the rest perfectly, but that is an upper bound rather than a measure of confounding, because CheXpert membership and the Pneumonia label partition the corpus identically. The informative test holds diagnosis fixed: within the tuberculosis class, TBX11K is separated from Montgomery, Shenzhen, DA and DB at 0.982 balanced accuracy and 0.999 AUROC against a 0.781 majority baseline. The equivalent test within Normal reaches 1.000 but rests on only 10 non-TBX11K test images.

Third, the checkpoints were evaluated where source is constant: TB against Normal, both from TBX11K, 275 test images. Performance is retained (ResNet-50 ImageNet 0.986, MobileNetV2 0.989, EfficientNet-B0 0.975, Victorio 0.836) and TB recall is essentially unchanged (0.984 against 0.988). This two-class problem has a 0.545 majority baseline and is not comparable with the three-class headline, but it establishes that TB/Normal discrimination does not depend on identifying the dataset of origin.

5.4 Deployment latency

Table 4 reports warm-state per-request latency for all eight pairs on both backends, aggregated over 100 measured requests per (backend, model, image) cell across three representative test images. Predicted classes

were identical across CPU and GPU for every cell, confirming functional equivalence of the deployed models across devices.

The CPU-versus-GPU comparison is architecture-dependent. GPU halved the median latency of ResNet-50 (250 ms against 517 ms under the CXR regime, a $2.1\times$ speed-up) and gave a smaller but consistent advantage on EfficientNet-B0 (110 ms against 161 ms, roughly $1.5\times$). For MobileNetV2 the two backends were within tens of milliseconds and effectively tied. Only for the compact custom CNN did CPU clearly win, at 19 ms against 48 ms, a $2.5\times$ inversion. Tail behaviour also diverged: CPU P95 reached up to three times its own median (1,430 ms against 517 ms for ResNet-50 CXR), whereas GPU P95 stayed within roughly twice the median across all models.

6 Discussion

Capacity saturates, and pretraining helps. H1 is not supported: a 6,627-parameter network trails the backbones by 14 pp. The more useful observation is where the gap is not: MobileNetV2 at 2.2 M parameters matches ResNet-50 at 23.5 M to within sampling error, so capacity beyond roughly two million parameters buys nothing measurable on this task. Pretraining raises accuracy for every backbone, though only the MobileNetV2 difference is clearly resolvable at this sample size. Read alongside Section 5, the practical implication is that the choice worth making is pretrained-versus-scratch, not large-versus-small.

The source confound is real, and partly bounded. The corpus is not source-balanced, and Section 5.3 shows the consequence is not hypothetical: acquisition source is recoverable from raw pixels with the diagnosis held fixed, and a classifier recovers it at 0.982 balanced accuracy within the tuberculosis class. Performance nonetheless holds on a single-source subset, so TB/Normal discrimination reflects radiographic content rather than dataset identity. Pneumonia is the unresolved case: it is drawn entirely from CheXpert, no within-source control is available, and its perfect classification may in part reflect acquisition characteristics. Resolving this requires a pneumonia set from a second source, or domain-adversarial training [Ganin et al., 2016], and is beyond the scope of this study. The reported accuracies should be read as an upper bound on clinical performance.

Explainer agreement is not invariant to capacity. Agreement varies with both architecture and initialisation, and is consistently higher for pretrained models. That the Victorio pair returns an identical value under both regimes, while every backbone pair does not, indicates the variation tracks the representation rather than the measurement. Interpreted against the attainable ceiling of approximately 0.29, the pretrained backbones agree close to that limit while the compact network does not, which is consistent with a less spatially specific representation. We report this as an observation about two attribution methods, not as evidence that either localises pathology correctly; the sanity-check literature [Adebayo et al., 2018] cautions against the stronger reading.

Backend choice is architecture-dependent (H2). H2 is partially supported: the conventional expectation that CPU serving suits small models holds only at the lower bound of capacity. GPU per-request overhead (kernel launch, host-to-device transfer, stream synchronisation) is roughly constant, so it is amortised by the heavier forward passes of ResNet-50 and EfficientNet-B0 but dominates the compute of the compact network, inverting the ordering. For low-throughput single-image screening the guidance is split: pair compact backbones with CPU pods, retain GPU serving for ResNet-class architectures. The CPU pod also showed a heavy right tail, so GPU serving is preferable when worst-case rather than typical latency is the constraint.

Limitations. This is a pilot study on a single workstation, trained on one dataset combination. The 480-image test set limits resolution to roughly 3 pp, so the three backbones cannot be separated. All models share one hyperparameter configuration, which controls for tuning effort but disadvantages architectures whose optimum lies elsewhere; the from-scratch ResNet-50 run showed unstable validation behaviour, and each configuration was trained once, so no seed variance is reported. The tuberculosis class mixes active and healed disease. The benchmark ran at concurrency one on a single-node Minikube cluster; concurrent workloads would likely shift the CPU/GPU balance. Clinical translation would require validation under regulatory frameworks (for example EU MDR and GDPR).

7 Conclusion

Three findings stand out from this pilot. Capacity saturates early: MobileNetV2 at 2.2 M parameters is indistinguishable from ResNet-50 at 23.5 M, while the 6,627-parameter network trails by 14 pp. Grad-CAM and LIME agreement varies systematically with architecture and initialisation rather than being invariant to capacity, and is stable across mask thresholds. The deployment benchmark shows an architecture-dependent crossover, with GPU $2.1\times$ faster than CPU on ResNet-50 and $2.5\times$ slower on the compact network, so compact backbones should be paired with CPU pods while GPU serving remains right for heavier architectures. The corpus is not source-balanced, and the reported accuracies are an upper bound on clinical performance.

References

- Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. (2018). Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, pages 9505–9515.
- Chauhan, A., Chauhan, D., and Rout, C. (2014). Role of gist and PHOG features in computer-aided diagnosis of tuberculosis without segmentation. *PLoS One*, 9(11):e112980.
- Dean, J. and Barroso, L. A. (2013). The tail at scale. *Communications of the ACM*, 56(2):74–80.
- Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016). Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35.
- Gujarati, A., Karimi, R., Alzayat, S., Hao, W., Kaufmann, A., Vigfusson, Y., and Mace, J. (2020). Serving DNNs like clockwork: Performance predictability from the bottom up. In *Proceedings of the 14th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, pages 443–462.
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.
- Irvin, J., Rajpurkar, P., Ko, M., Yu, Y., Ciurea-Ilcus, S., Chute, C., Marklund, H., Haghighi, B., Ball, R., Shpanskaya, K., et al. (2019). CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 590–597.
- Jaeger, S., Candemir, S., Antani, S., Wang, Y.-X. J., Lu, P.-X., and Thoma, G. (2014). Two public chest x-ray datasets for computer-aided screening of pulmonary diseases. *Quantitative Imaging in Medicine and Surgery*, 4(6):475–477.
- Liu, Y., Wu, Y.-H., Ban, Y., Wang, H., and Cheng, M.-M. (2020). Rethinking computer-aided tuberculosis diagnosis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2646–2655.
- Mei, X., Liu, Z., Robson, P. M., Marinelli, B., Huang, M., Doshi, A., Jacobi, A., Cao, C., Link, K. E., Yang, T., Wang, Y., Greenspan, H., Deyer, T., Fayad, Z. A., and Yang, Y. (2022). RadImageNet: An open radiologic deep learning research dataset for effective transfer learning. *Radiology: Artificial Intelligence*, 4(5):e210315.
- Parise, A. and Mac Namee, B. (2023). Exploring optimal configurations in active learning for medical imaging. In Bramer, M. and Stahl, F., editors, *Artificial Intelligence XL: 43rd SGAI International Conference on Artificial Intelligence, AI 2023*, volume 14381 of *Lecture Notes in Computer Science*, pages 75–88. Springer, Cham.
- Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., Lungren, M. P., and Ng, A. Y. (2017). CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep learning. *arXiv preprint arXiv:1711.05225*.
- Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). “Why should I trust you?”: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1135–1144.
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520.
- Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 618–626.
- Tan, M. and Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 6105–6114.
- Ucan, M., Kaya, B., Aygun, O., Kaya, M., and Alhaji, R. (2025). Comparison of EfficientNet CNN models for multi-label chest x-ray disease diagnosis. *PeerJ Computer Science*, 11:e2968.
- Zech, J. R., Badgeley, M. A., Liu, M., Costa, A. B., Titano, J. J., and Oermann, E. K. (2018). Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLOS Medicine*, 15(11):e1002683.

Motion-Based Detection and Tracking of *Trypanosoma cruzi* Parasites in Microscopy Videos

Senem Aktas¹, Carlos Brito-Loeza², Hugo Ruiz-Piña³, Lavdie Rada¹

¹ *Faculty of Engineering and Natural Sciences, Bahcesehir University, Istanbul, Turkey*

² *Universidad Autónoma de Yucatán, Merida, Mexico,*

³ *Centro de Investigaciones Regionales Dr. Hideyo Noguchi, Mérida, México.*

Abstract

Chagas disease, caused by the parasite *Trypanosoma cruzi*, poses a substantial threat to global health, particularly in Latin America. Traditional diagnostics involve the time-consuming and error-prone process of manual analysis of stained blood samples under a microscope. In recent years, the diagnosis process has been significantly enhanced by learning-based algorithms, which analyze digital images of stained blood smears automatically. Nevertheless, computational algorithms can turn ineffective as a result of blood staining being defective by a variety of factors, such as smear preparation, staining, and interpretation. In this collaborative work, we propose an approach for monitoring motile *Trypanosoma cruzi* parasites in our microscopy videos. The pipeline focuses on movement, contours, and background contrast as primary cues, mainly relying on dense optical flow and a custom point-based tracking algorithm that is both scalable and dependency-free. Code and sample outputs are available at ¹.

Keywords: Chagas disease, Fast moving object, Point tracking, Dense optical flow

1 Introduction

Chagas disease, which is caused by the *Trypanosoma cruzi* (*T. cruzi*) parasite, is a global health problem with a prevalence of millions in Latin America, particularly Mexico. It is transmitted by domestic pets, wild mammals, and insects that feed on blood. Transmission can occur through infected blood transfusions, organ transplants, pregnancy, and contaminated food or drink. Reliable risk assessment, early diagnosis, and optimal treatment strategies are among the challenges that persist, despite the progress that has been made [Rassi and Marin-Neto, 2010, Nagajyothi et al., 2012, Rojo-Medina et al., 2018].

The first learning-based methods for detecting *T. cruzi* in images involve the use of conventional image processing techniques, such as color segmentation, texture analysis, and morphological operations; these methods derive features from microscopic blood smear image samples. Subsequently, these features are processed using machine learning algorithms, including AdaBoost, Support Vector Machines, and Artificial Neural Networks [Soberanis-Mukul et al., 2013, Uc-Cetina et al., 2015]. Deep learning, especially Convolutional Neural Networks (CNNs), has transformed parasite identification in images. CNNs provide end-to-end classification and segmentation solutions without manual feature extraction. Deep learning models can accurately discriminate infected and uninfected blood smear images. These models work directly with raw images to learn hierarchical features [Ojeda-Pat et al., 2022, Pereira et al., 2022, Martins et al., 2023, Rada et al., 2024].

Although identifying the Chagas parasite in videos has shown promising results by using optical flow [Martins et al., 2021, Martins et al., 2024], tracking its movement presents a greater challenge due to its rapid and erratic movement. Here, we enumerate some of the difficulties:

¹ <https://github.com/senemaktas/Parasite-Chagas>

1. High velocities lead to significant blurring and, large displacement between consecutive frames complicate the position prediction [Rozumnyi et al., 2021, Aktas et al., 2025].
2. The tiny size of the parasite offers limited distinguishability due to low resolution and poor contrast against their background [Haalck et al., 2024].
3. These parasites shows irregularities and transformations in appearance can complicate detection [Chen et al., 2020]
4. Curvilinear trajectories and irregular motions of the parasite can exacerbate the tracking challenges.

Popular tracking libraries, such as DeepSort [Wojke et al., 2017] and StrongSort [Du et al., 2023], perform poorly because *T. cruzi* parasite is tiny and Fast-Moving Object (FMO). We present a method (see Section 3) specifically designed for the automated, label-free detection and tracking of *T. cruzi* parasites in standard microscopy videos (Section 2). The method is designed as a low-cost, easily deployable pipeline that can be integrated into existing microscopy workflows without requiring specialized hardware.

2 Dataset

The dataset collection study was conducted using experimentally infected albino ICR (Institute for Cancer Research) strain mice, which were all in the acute phase of Chagas disease. ICR mice are widely used in scientific research due to their genetic variability, docility, and good reproductive characteristics. Blood samples were collected directly from the tail of each infected animal during the peak of parasitemia. Microscopy was performed using a Motic BA300 microscope, with data acquisition performed at a magnification of 40x. A Canon EOS Rebel T1i, Model 500D digital camera was used for video recording, which was manually controlled by the user rather than through microscopy image analysis software. The camera was connected to the microscope using a C-mount adapter that did not include any optics, and it operated at a resolution of 1280×720 pixels. Three mice were used per experimental replicate. Video recording was performed twice, with a ~30-day interval between sessions. 24 short videos were recorded at 20 FPS, totaling 32 minutes of footage. Some sections of the recordings contain depictions of *T. cruzi*, and some others do not. Furthermore, videos exhibit a wide range of approaches, varying illuminations, and different levels of motion when capturing the parasites. Furthermore, experts extracted 70×70-pixel ROIs from the 24 videos, creating an image dataset of 1,161 positive (parasite present) and 1,697 negative samples.

3 Methodology

An overview of the applied methodology is shown in Figure 1. The experimental methodology steps are explained in the following subsections and have been evaluated on the dataset with the custom metrics.

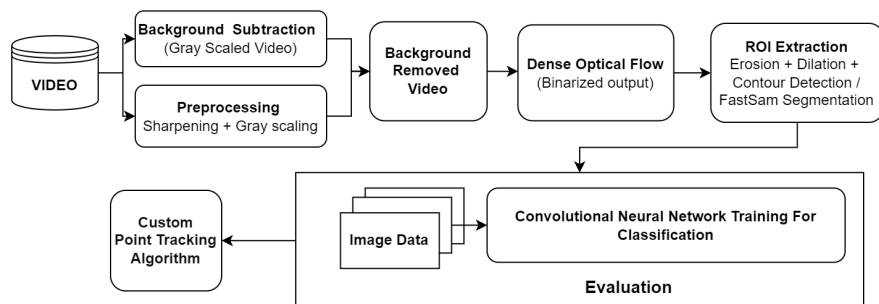


Figure 1: Flowchart of the proposed methodology for monitoring the *Trypanosoma cruzi* parasite.

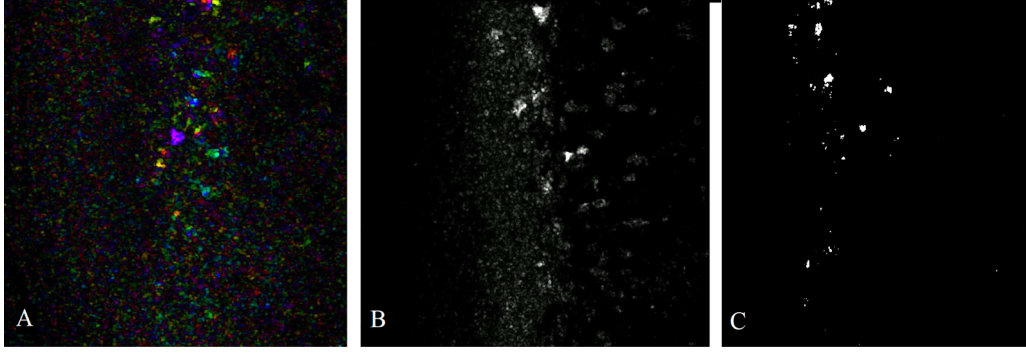


Figure 2: Exploration of dense optical flow output: (A) color shows the direction and magnitude of movement (B) grayscaling preserves motion integrity (C) binarization emphasize potential regions of parasites.

3.1 Pre-processing and Background Subtraction

Preliminary analysis revealed that chromatic information is non-discriminative for parasite localization. So frames are converted to grayscale and convolved with the Laplacian kernel to enhance edges. To avoid per-video parameter tuning under varying backgrounds, we apply background subtraction [Paul et al., 2017]: Algorithm 1 builds an adaptive background, and its absolute difference from each frame isolates moving objects.

Algorithm 1 Background Estimation via Difference-Weighted Averaging

```

1:  $sum \leftarrow 0$ ,  $count \leftarrow 0$  ▷ running weighted sum of frames, number of accepted frames
2: for all frames  $frame$  of video whose index is divisible by frame stride do
3:   if  $count = 0$  then
4:      $sum \leftarrow frame$  ▷ seed with the first frame
5:   else
6:      $background \leftarrow sum / count$  ▷ current mean background
7:      $diff \leftarrow (frame, background) / 255$ 
8:      $weight \leftarrow 1 - diff$  ▷ low weight where the frame disagrees
9:      $sum \leftarrow sum + frame * weight$ 
10:  end if
11:   $count \leftarrow count + 1$ 
12:  if  $count > maxFrames$  then break
13: end for
14: return  $(sum / count)$ 

```

3.2 Dense Optical Flow

The living *T. cruzi* parasites shows rapid movement, distinguishing them from other entities and organisms. Dense optical flow algorithm [Farneback, 2003] employed to identify areas of interest where the parasite may reside. Output is visualized as a color-coded image, where each pixel is assigned a specific color that reflects its magnitude and direction.

3.3 Region of Interest (ROI) extraction

To exclude irrelevant objects, morphological operations are applied to the optical flow outputs. Small regions are removed via erosion with 8×8 kernel. Subsequently, broad regions suspected of parasites are enhanced and refined through dilation with 12×12 kernel. Kernel sizes were chosen to avoid over-dilating minor features while preserving candidate parasite regions. Then filtered output is binarized. Figure 2 illustrates the steps.

We applied two different methods for localizing parasites. Contour detection [Suzuki et al., 1985] approach is a border-following technique intended to extract contours from binary images. FastSAM [Zhao et al., 2023]

algorithm is a simplified variant of the Segment Anything Model that facilitates rapid segmentation of objects in images. Among the two methods, our customized contour detection method tends to identify a wider range of regions, particularly those that may not be optically evident while the FastSAM method seems to adopt a more conservative approach.

3.4 Evaluation

Using image dataset described in Section 2; various CNNs including VGG16 [Simonyan and Zisserman, 2014], InceptionV3 [Szegedy et al., 2016], and Xception [Chollet, 2017] were fine-tuned to confirm selected ROIs whether they truly corresponds to a *T. cruzi* parasite. Our experiments revealed that, aside from VGG16, the other models struggled to effectively classify the ROIs. The VGG16 model was trained for 15 epochs using 70×70 inputs, an SGD optimizer ($\eta = 0.001$), and binary cross-entropy loss with sigmoid activation.

3.5 Custom Point Tracking

To tackle the *T. cruzi* tracking challenge, the proposed technique is depicted in Figure 3.

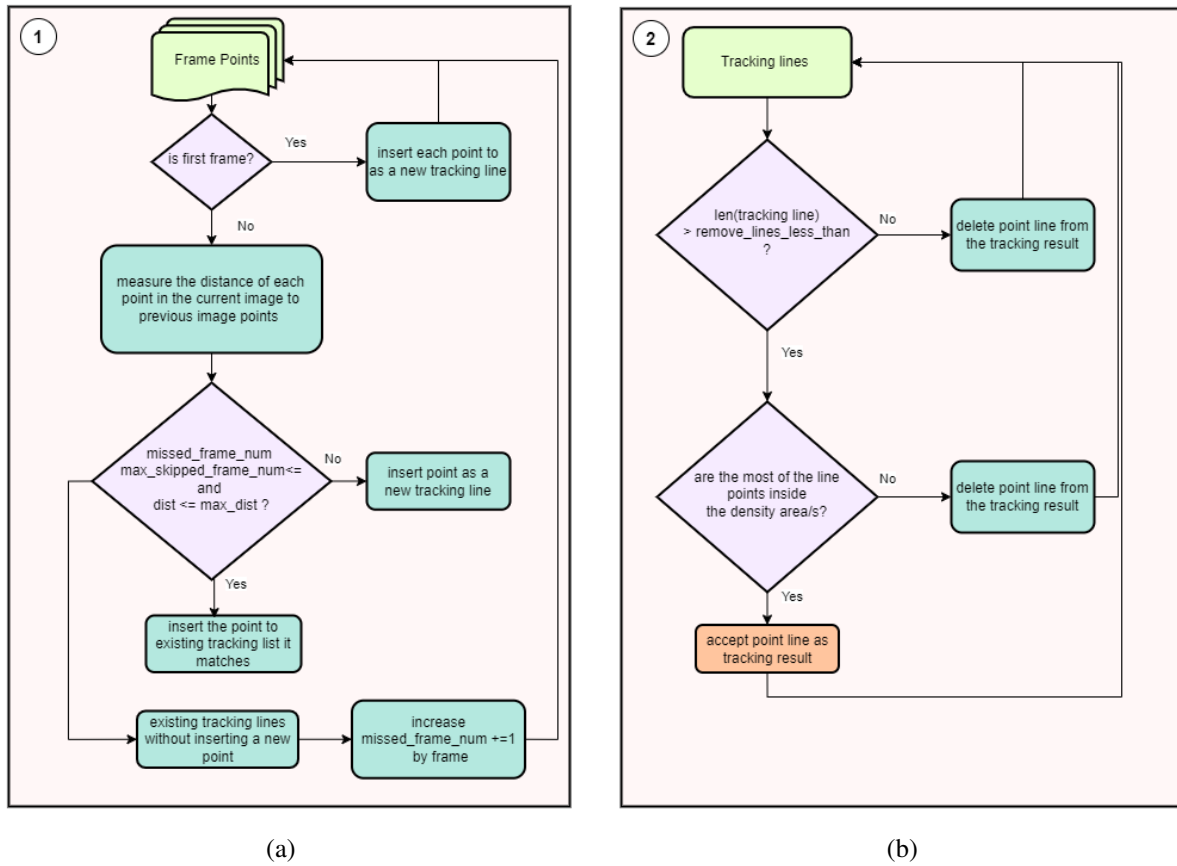


Figure 3: Custom Tracking Algorithm Flow. (a) covers the Steps 1 and 2. (b) covers the Steps 3 and 4.

- **Step 1:** In the initial frame, each extracted ROI centre is assigned a unique tracking ID. For each ID, alongside object locations, "*maximum shortest distance*", "*current frame number*", "*start frame number*", and "*end frame number*" values are recorded to facilitate tracking analysis.
- **Step 2:** In the subsequent frames, new objects' points are compared with the recorded tracking points to either align with existing IDs or assign new IDs if no correspondence is found. This step continues until all frames in the video have been thoroughly evaluated.

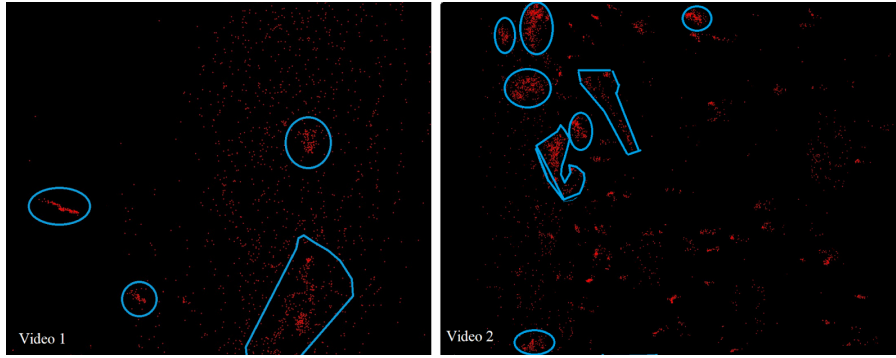


Figure 4: Parasite-dense areas (blue) are mapped by plotting the all ROIs midpoints (red) onto a single image.

1. If the distance between two objects points being compared is less than the "*maximum shortest distance*" variable, the new object point is linked to the compared object's tracking ID. The "*current frame number*" is updated to the "*end frame number*", while the "*start frame number*" remains unchanged which defines when ID tracking started.
 2. If any recorded tracking ID is not assigned in the current frame, the *maximum skipped frame number* is incremented by 1. If this value exceeds a predetermined threshold (e.g., 25), no new object will be assigned to the tracking ID, regardless of the distance between points. This decision is based on the assumption that the object is either no longer alive or out of view.
 3. If the new objects do not align with any existing tracking ID, create their own distinct tracking IDs.
- **Step 3:** Next, tracking IDs with fewer detections than the threshold defined by "*remove lines less than*" (e.g., 15) are removed, assuming parasites did not survive such brief periods. This eliminates erroneous or incomplete trackings, ensuring the analysis focuses on reliable data.
 - **Step 4:** During experiments, Figure 4 was produced by plotting each ROIs midpoints onto a single image, where dense areas indicate high parasite prevalence. Trackings outside these dense boundaries were excluded for robustness. This optional step, interchangeable with Step 3, ensures that even if a tracking ID exceeds the threshold, will be removed if it is not located within dense area(s).

Figure 5 shows tracking results across sequential frames for a video at several-second intervals. Each number represents a unique tracking ID marking the location of a parasite. By following the frame order (from A to D), changes in the ID values can be observed. If the same parasite is tracked in the subsequent frame, the same ID value will appear in that frame as well. For instance, the parasite marked with ID 103 appears in frames A & B but is no longer found or tracked in frames C & D. ²

Tracking parameters were tuned qualitatively by selecting values that yielded the best performance across various videos. *T. cruzi* parasites defined as FMOs, "*maximum shortest distance*" should not be set too low due to large object displacements between consecutive frames. Too high a value may cause inaccuracies by linking points that belong to different objects. Similarly, parasites must appear in at least a minimum number of frames (e.g., 15) for effective tracking, controlled by the "*remove lines less than*" parameter. Setting this value too high risks losing valid tracks, while too low a value has negligible effect.

4 Experimental Results

Table 1 shows the tracking results for the *T. cruzi* parasites across the 24 videos in our dataset. **Label** is the identifier assigned to each video. The **number of tracked parasites** (NTP) refers to the distinct parasites identified in each video that the algorithm monitored. The **mean of frames** (MF) represents the average

²For a better understanding we share the code and some of the processed videos in Github repository.

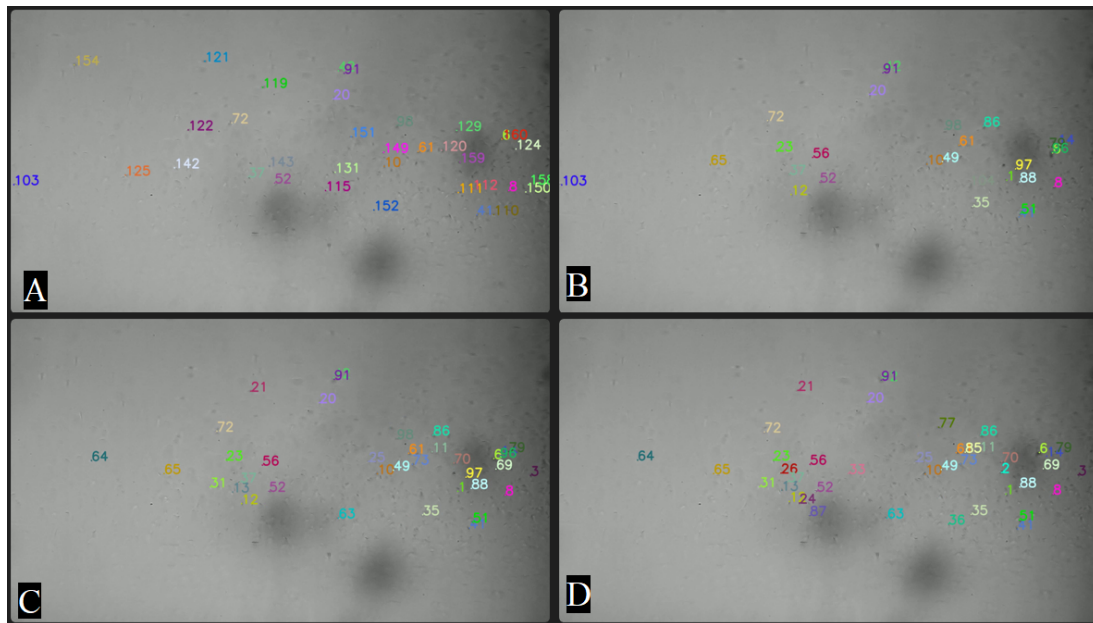


Figure 5: Sample Output of Custom Tracking with FastSam: locations marked with tracking IDs on the sequential images (in order of A-B-C-D) with a few second time intervals, indicate the presence of parasites.

number of frames in which algorithm tracked all detected parasites in the video. In 15 of the 24 videos analyzed, our method utilizing **contour detection/tracing (CT)** segmentation outperformed the other methods by successfully tracking the highest number of parasites. The runner-up was our approach employing FastSAM segmentation, significantly lagging behind the DeepSort and StrongSort methods. Conversely, both techniques, especially StrongSort, significantly outscored our custom method regarding the mean of frames variable. These findings indicate that our method can track a larger quantity of parasites, albeit over a limited number of frames, whereas deep learning techniques are more selective, tracking fewer parasites but for extended time frames.

Label	Our Method				DeepSort				StrongSort			
	CT		FastSAM		CT		FastSAM		CT		FastSAM	
	NTP	MF	NTP	MF	NTP	MF	NTP	MF	NTP	MF	NTP	MF
20	22	42.04	21	47.52	3	145.33	4	119.75	3	155.67	2	179.5
33	135	80.86	170	70.91	40	89.82	49	66.65	23	182.26	21	208.42
36	235	80.23	270	80.11	74	94.67	70	93.01	62	133.98	58	137.67
38	85	60.55	88	68.00	61	41.54	55	44.58	21	175.52	15	201.73
42	134	66.28	163	59.91	39	80.66	42	80.59	26	178.42	24	178.50
44	60	57.20	72	67.82	50	43.20	45	45.68	18	164.28	19	149.73
45	92	65.08	89	69.32	55	33.21	44	39.63	14	176.54	12	196.58
46	63	55.59	53	72.49	13	113.23	12	92.91	10	215.20	10	212.80
47	102	63.53	97	65.93	43	38.23	40	41.50	15	152.80	8	175.75
48	65	50.17	73	42.86	12	74.83	13	67.61	10	175.00	8	180.12
49	96	55.35	103	50.25	18	88.17	19	77.84	11	188.91	13	117.46
50	329	78.51	208	60.65	92	35.67	44	62.20	26	247.26	20	253.40
57	447	103.82	526	83.04	197	58.98	215	62.95	48	180.27	29	230.55
58	304	89.59	234	78.73	59	72.81	43	88.11	36	171.67	25	140.56
59	307	64.41	253	46.51	40	44.82	31	48.42	13	135.15	11	89.36
60	151	63.91	138	53.71	35	43.88	27	46.67	12	203.75	9	213.22
61	456	49.74	175	28.26	37	69.78	10	85.70	11	96.36	5	137.60
62	966	62.73	470	31.12	113	64.75	52	54.48	95	190.80	88	158.04
67	447	22.32	335	20.18	31	76.54	32	77.37	26	153.73	31	146.84
68	375	73.00	343	49.98	40	77.00	44	55.86	43	91.60	30	95.60
69	389	64.59	311	48.99	53	61.22	43	52.14	40	98.12	37	77.38
70	253	46.90	215	30.56	29	49.07	22	49.73	11	228.00	7	225.28
71	195	25.60	140	25.98	12	88.66	12	85.16	11	191.72	8	174.00
72	140	64.96	172	68.78	96	53.54	101	49.06	44	179.47	42	165.78

Table 1: Results for all tested methods. MF values can be converted to seconds using 20 FPS. Bold numbers indicate the highest values for the NTP and italic numbers indicate the highest values for the MF variable.

5 Conclusion

We propose an approach that enables monitoring motile *T. cruzi* parasites in microscopy videos. Current pipeline focuses on movement, contours, and background contrast as primary cues. We encountered challenges in accurately counting the parasites using popular tracking libraries due to their fast movement, small size, and unpredictable behavior. While our approach showed promise, our current model was trained and validated on our specific dataset. Performance could be sensitive to imaging artefacts (e.g., variations in illumination, camera noise) and may not generalize directly to other parasite strains, or microscopy setups. Potential integration with molecular and immunological tools, could create a more robust, multi-modal diagnostic pipeline. The motion-based assay could serve as a rapid, low-cost, and stain-free initial screening tool. Samples flagged as positive could be confirmed with a specific molecular test (e.g., qPCR).

Acknowledgments

This work is supported in part by the Academy of Medical Sciences Award (NGR1\1718). The Institutional Animal Care and Use Committee of the University (IACUC-03-2021) gave their approval to all of the approaches that were utilized for the treatment of animals and the collection of samples. Video footage was captured at the Dr. Hideyo Noguchi Regional Research Center, which is affiliated with the Autonomous University of Yucatan.

References

- [Aktas et al., 2025] Aktas, S., Markham, C., McDonald, J., and Dahyot, R. (2025). Benchmarking sam2-based trackers on fmox. In *33rd International Conference on Artificial Intelligence and Cognitive Science (AICS 2025)*, pages 1–12, Dublin, Ireland.
- [Chen et al., 2020] Chen, Z., Zhong, B., Li, G., Zhang, S., and Ji, R. (2020). Siamese box adaptive network for visual tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6668–6677.
- [Chollet, 2017] Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258.
- [Du et al., 2023] Du, Y., Zhao, Z., Song, Y., Zhao, Y., Su, F., Gong, T., and Meng, H. (2023). Strongsort: Make deepsort great again. *IEEE Transactions on Multimedia*, 25:8725–8737.
- [Farnebäck, 2003] Farnebäck, G. (2003). Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer.
- [Haalck et al., 2024] Haalck, L., Thiele, S., and Risse, B. (2024). Tracking tiny insects in cluttered natural environments using refinable recurrent neural networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7126–7135.
- [Martins et al., 2023] Martins, G. L., Ferreira, D. S., Carneiro, C. M., and Bianchi, A. G. (2023). Trypanosoma cruzi detection using lstm convolutional autoencoder. In *Simpósio Brasileiro de Computação Aplicada à Saúde (SBCAS)*, pages 443–454. SBC.
- [Martins et al., 2024] Martins, G. L., Ferreira, D. S., Carneiro, C. M., Nogueira-Paiva, N. C., and Bianchi, A. G. (2024). Trajectory-driven computational analysis for element characterization in trypanosoma cruzi video microscopy. *Plos one*, 19(6):e0304716.
- [Martins et al., 2021] Martins, G. L., Ferreira, D. S., and Ramalho, G. L. (2021). Collateral motion saliency-based model for trypanosoma cruzi detection in dye-free blood microscopy. *Computers in Biology and Medicine*, 132:104220.

- [Nagajyothi et al., 2012] Nagajyothi, F., Machado, F. S., Burleigh, B. A., Jelicks, L. A., Scherer, P. E., Mukherjee, S., Lisanti, M. P., Weiss, L. M., Garg, N. J., and Tanowitz, H. B. (2012). Mechanisms of trypanosoma cruzi persistence in chagas disease. *Cellular microbiology*, 14(5):634–643.
- [Ojeda-Pat et al., 2022] Ojeda-Pat, A., Martin-Gonzalez, A., Brito-Loeza, C., Ruiz-Pina, H., and Ruz-Suarez, D. (2022). Effective residual convolutional neural network for chagas disease parasite segmentation. *Medical & Biological Engineering & Computing*, 60(4):1099–1110.
- [Paul et al., 2017] Paul, N., Singh, A., Midya, A., Roy, P. P., and Dogra, D. P. (2017). Moving object detection using modified temporal differencing and local fuzzy thresholding. *The Journal of Supercomputing*, 73(3):1120–1139.
- [Pereira et al., 2022] Pereira, A. S., Mazza, L. O., Pinto, P. C., Gomes, J. G. R., Nedjah, N., Vanzan, D. F., Pyrrho, A. S., and Soares, J. G. (2022). Deep convolutional neural network applied to trypanosoma cruzi detection in blood samples. *International Journal of Bio-Inspired Computation*, 19(1):1–17.
- [Rada et al., 2024] Rada, L., Kumar, P., Martin-Gonzalez, A., and Brito-Loeza, C. (2024). Chagas parasite classification in blood sample images using different machine learning architectures. *Medical & Biological Engineering & Computing*, 62(1):195–206.
- [Rassi and Marin-Neto, 2010] Rassi, A. and Marin-Neto, J. A. (2010). Chagas disease. *The Lancet*, 375(9723):1388–1402.
- [Rojo-Medina et al., 2018] Rojo-Medina, J., Ruiz-Matus, C., Salazar-Schettino, P. M., and González-Roldán, J. F. (2018). Enfermedad de chagas en México. *Gaceta médica de México*, 154(5):605–612.
- [Rozumnyi et al., 2021] Rozumnyi, D., Matas, J., Šroubek, F., Pollefeys, M., and Oswald, M. R. (2021). Fmodetect: Robust detection of fast moving objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3541–3549.
- [Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- [Soberanis-Mukul et al., 2013] Soberanis-Mukul, R., Uc-Cetina, V., Brito-Loeza, C., and Ruiz-Piña, H. (2013). An automatic algorithm for the detection of trypanosoma cruzi parasites in blood sample images. *Computer methods and programs in biomedicine*, 112(3):633–639.
- [Suzuki et al., 1985] Suzuki, S. et al. (1985). Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing*, 30(1):32–46.
- [Szegedy et al., 2016] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.
- [Uc-Cetina et al., 2015] Uc-Cetina, V., Brito-Loeza, C., and Ruiz-Piña, H. (2015). Chagas parasite detection in blood images using adaboost. *Computational and Mathematical Methods in Medicine*, 2015(1):139681.
- [Wojke et al., 2017] Wojke, N., Bewley, A., and Paulus, D. (2017). Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE.
- [Zhao et al., 2023] Zhao, X., Ding, W., An, Y., Du, Y., Yu, T., Li, M., Tang, M., and Wang, J. (2023). Fast segment anything. *arXiv preprint arXiv:2306.12156*.

Robust Breast Ultrasound Classification Through Dataset Auditing, Preprocessing Standardization, and Ensemble Fusion

Sana Siddique^{*1} and Lavdie Rada^{†1}

¹Bahcesehir University, Istanbul, Türkiye, lavdie.rada@bau.edu.tr

Abstract

Deep learning models for breast ultrasound classification frequently report competitive benchmark performance on individually collected private or public datasets, yet this paper demonstrates that such results remain limited by dataset artifacts, inconsistent preprocessing, and weak external generalization. This study investigates breast ultrasound malignancy classification using an artifact-aware pipeline that combines dataset auditing, controlled preprocessing, backbone comparison, and ensemble fusion. Eight public breast ultrasound datasets were audited for duplicate or near-duplicate images, color Doppler overlays, and measurement markers. The audit revealed substantial non-anatomical content across datasets, including extensive duplicates in BUSDB2023, universal Doppler overlays in BUS-UCLM, and universal measurement markers in BRA, indicating a high risk of shortcut learning and biased evaluation. Three convolutional backbones, VGG-16, ResNet-50, and EfficientNetV2S, were evaluated under raw, median-filtered, CLAHE-enhanced, and full artifact-reduction preprocessing conditions using a unified training and threshold-selection protocol. Results showed that preprocessing strongly affected sensitivity, specificity, and AUC, often as much as architecture selection. Models that achieved high single-dataset performance showed reduced AUC under multi-dataset evaluation, highlighting limited generalizability. Ensemble experiments demonstrated that ResNet-50 and EfficientNetV2S provide complementary decision behavior: weighted fusion achieved the highest specificity, while attention-based fusion achieved the highest sensitivity and F1-score. These findings suggest that deployable breast ultrasound AI should be developed as a complete clinical pipeline rather than as an isolated model-selection task. Dataset curation, artifact removal, transparent preprocessing, calibration, and clinically aligned threshold selection are essential for reliable evaluation and future multi-center deployment.

Keywords: breast ultrasound; breast cancer classification; dataset auditing; image artifacts; preprocessing; ensemble learning; sensitivity; specificity.

1 Introduction

Breast cancer remains one of the most common malignancies among women worldwide, and early detection is essential for improving clinical outcomes [World Health Organization, 2023]. Ultrasound is an important modality in breast assessment because it is non-invasive, real-time, comparatively inexpensive, and useful for evaluating dense breast tissue [Al-Dhabyani et al., 2020]. These advantages have motivated extensive research on computer-aided breast lesion analysis using convolutional neural networks (CNNs), including VGG-style models, residual networks, and compound-scaled architectures [Simonyan and Zisserman, 2015, He et al., 2016, Lee and Choi, 2025]. However, a critical gap exists between reported benchmark performance and practical clinical deployability. Many studies report high accuracy or area under the receiver-operating-characteristic curve (AUC) on public datasets, but these results are often obtained under favorable experimental conditions, such as single-dataset evaluation, limited external validation, and

^{*}ORCID: [0009-0008-3436-5602](https://orcid.org/0009-0008-3436-5602)

[†]ORCID: [0000-0002-2688-4962](https://orcid.org/0000-0002-2688-4962)

inconsistent preprocessing [Zhang et al., 2020a, Thomas et al., 2023]. In clinical practice, ultrasound images vary across scanners, operators, hospitals, annotation conventions, and storage pipelines. A model that performs well on a curated benchmark may therefore fail when exposed to images acquired under different conditions. Two methodological issues motivate this study. First, public breast ultrasound datasets can contain non-anatomical visual information, including measurement calipers, color Doppler overlays, text labels, annotation marks, and duplicate or near-duplicate images. If these cues are correlated with dataset source or diagnostic label, a classifier may learn shortcut features that inflate retrospective performance without improving lesion characterization. Second, architecture comparisons are frequently confounded by uncontrolled preprocessing and dataset composition. Differences attributed to model design may instead reflect artifact prevalence, class imbalance, contrast normalization, image resolution, or annotation practice. The clinical consequence is most evident in the sensitivity–specificity trade-off. A classifier with high specificity but poor sensitivity may appear acceptable according to aggregate metrics, yet it would be unsafe for screening if it misses many malignant lesions. Conversely, very high sensitivity with poor specificity can overload radiologists with false positives and unnecessary follow-up. Therefore, reliable evaluation requires reporting threshold-dependent clinical metrics under clearly specified preprocessing conditions, not only headline AUC. This paper investigates breast ultrasound classification through a pipeline that combines dataset auditing, artifact-aware preprocessing, controlled backbone evaluation, and ensemble fusion. The principal contributions are as follows:

1. **Dataset audit.** Eight public breast ultrasound datasets are quantitatively inspected for duplicate or near-duplicate images, color Doppler overlays, markers with color indicating cancer, and measurement markers that may bias training and evaluation.
2. **Preprocessing analysis.** Raw, median-filtered, CLAHE-enhanced, and full artifact-reduction pipelines are compared under a common training and threshold-selection protocol.
3. **Single-Dataset vs. Multi-Dataset Comparison.** We show that models achieving 88-95% AUC on single datasets drop to 70-83% AUC when tested on multi-dataset combinations, revealing substantial overfitting and dataset-specific bias.
4. **Complementary ensemble learning.** Averaging, weighted, stacked, and attention-based fusion strategies are evaluated to combine the high-sensitivity behavior of ResNet-50 with the stronger specificity of EfficientNetV2S.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 describes the datasets, preprocessing pipeline, model architectures, training protocol, and ensemble methods. Section 4 presents quantitative results. Section 5 discusses clinical implications and limitations and concludes the study.

2 Related Work

In this study, we considered publicly available open-source datasets suitable for further research. The literature includes eight datasets containing breast ultrasound images: BrEaST/TCIA [Pawłowska et al., 2024], BUSI2020 [Al-Dhabyani et al., 2020], BUSDB2023 [Abbasian Ardakani et al., 2023], UDIAT2021 [Yap et al., 2018], OASBUD2017 [Piotrkowska-Wróblewska et al., 2017], BUS-UCLM [Vallez et al., 2025], Y.Guo [Guo et al., 2021], and BRA [Gómez-Flores et al., 2019]. These datasets were collected from different hospitals using different imaging tools. One of the main issues with such datasets is that some images have been marked by clinicians with red color overlays (e.g., color Doppler overlays) or cross-lines used to measure the extent of a lesion. We refer to such images as *potentially marked* images, and images without such markers or annotations as *unmarked* images. One baseline framework in the literature, presented by Zhang et al. [Zhang et al., 2020b], evaluated several models on a private breast ultrasound dataset, where InceptionV3 achieved the highest AUC of 0.905, outperforming VGG16, ResNet50, and VGG19, which obtained AUC values of

0.866, 0.851, and 0.847, respectively. Notably, their reported performance relied on images containing intact medical annotations and unstandardized contrast levels, which masks the true generalizability of the models, as such images would be classified as ‘potentially marked’ rather than ‘unmarked’ under our definition. A recent study by Koshy et al. [Koshy et al., 2026] investigates and compares EfficientNetB7, DenseNet121, ConvNeXtTiny, XGBoost Ensemble, and Soft Voting Ensemble. In our replication of this methodology, we substitute EfficientNet-V2-S for the original EfficientNet-B7 backbone, to keep this baseline comparison consistent with the EfficientNet variant used throughout the remainder of this paper’s own architecture and ensemble experiments. Across all three datasets, the Soft Voting Ensemble achieved the best overall performance, suggesting that ensemble-based approaches, particularly soft voting, can improve classification performance compared with individual architectures. For example, on the BUSI dataset, it obtained an accuracy of 88.46%, precision of 88.49%, recall of 88.46%, F1-score of 88.45%, and AUC of 95.38%. Similarly, for the BUS-UCLM and UDIAT datasets, the Soft Voting Ensemble achieved accuracy, precision, recall, F1-score, and AUC values of at least 90%. These results were obtained on ‘potentially marked’ datasets, without removing markers or crosses indicating cancer regions. To assess how much results differ when applying the methodology of Koshy et al. [Koshy et al., 2026] to the eight datasets collected for this study, we evaluated the architectures under the ‘potentially marked’ scenario rather than the fully ‘marked’ one, with results shown in Tables 1 and 2, respectively. We observed that performance drops significantly when evaluated on datasets lacking medical annotation attributes, indicating a high risk of model bias and data leakage, as networks may learn to classify based on artifacts rather than actual pathology. Datasets with available patient identifiers were split into training and test sets at the patient level to avoid leakage; datasets lacking this metadata were assigned entirely to the test set (stratified by cancer/non-cancer class), and all eight datasets were then combined into a single pooled cohort. Table 1 reports all five architectures evaluated under the original,

Table 1: Performance of the five baseline classification models, implemented following the methodology of Koshy et al. [Koshy et al., 2026], on the original datasets before artifact removal.

Model	Accuracy	F1-Score	Sensitivity	Specificity	AUC
Soft_Voting_Ensemble	0.8924	0.8522	0.8305	0.9293	0.9680
ConvNeXt_Tiny	0.9051	0.8718	0.8644	0.9293	0.9632
DenseNet121	0.8671	0.8073	0.7458	0.9394	0.9539
EfficientNetV2S	0.8734	0.8276	0.8136	0.9091	0.9462
XGBoost_Ensemble	0.5696	0.1282	0.0847	0.8586	0.4181

pre-artifact-removal setting. After applying the artifact-removal and preprocessing pipeline described in Section 3, EfficientNetV2S and the Soft Voting Ensemble retained the strongest performance, while the remaining three architectures showed substantially greater degradation. These two models were therefore carried forward to the subsequent ensemble experiments (Table 2); the XGBoost Ensemble was additionally unstable even before artifact removal (AUC 0.4181), indicating it failed to learn a usable decision boundary on this data. After artifact removal, both models show a marked drop in performance across all metrics

Table 2: Performance of the two best-performing models after artifact removal and preprocessing.

Model	Accuracy	F1-Score	Sensitivity	Specificity	AUC
EfficientNetV2S	0.7421	0.6734	0.6837	0.7792	0.7662
SoftVoting_Ensemble	0.7262	0.6230	0.5816	0.8182	0.7435

relative to Table 1, indicating that the original scores were inflated by non-anatomical artifacts rather than true lesion signal.

3 Materials and Methods

3.1 Dataset Audit

The eight aforementioned public breast ultrasound datasets were audited to verify dataset composition and quantify visible sources of potential bias. For each dataset, images were re-examined to estimate the total number of images, the proportion of cancer cases, duplicate or near-duplicate image groups, color Doppler overlays, and measurement markers. The audit was designed as a quality-control step before model training, with the goal of identifying possible leakage and shortcut-learning signals.

Duplicate detection. Perceptual hashing was used as an initial screen for near-duplicate content. Each image was resized to 8×8 pixels, converted to grayscale, and represented by a binary hash

$$h(i) = \begin{cases} 1, & I(i) > \mu, \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

where $I(i)$ is the intensity of pixel i in the resized image and μ is the mean intensity. Candidate pairs with Hamming distance below 5 were then validated using the structural similarity index measure (SSIM). Pairs with SSIM greater than 0.95 were treated as duplicate or near-duplicate images.

Artifact detection. Color overlays, including Color Doppler visualizations and colored annotations (e.g., red, blue, or yellow boxes and markers), were detected by analyzing differences between the RGB color channels. Since grayscale ultrasound images exhibit similar intensities across all three channels, images with large channel differences were flagged as containing color overlays.

3.2 Preprocessing Pipeline

The preprocessing pipeline was designed to reduce non-anatomical information while preserving lesion morphology. Colored overlays and markers were first segmented in HSV color space. Red regions were defined by $H \in [0, 10] \cup [160, 180]$ with saturation and value greater than 100, and blue regions were defined by $H \in [100, 130]$. The resulting binary masks were refined with a 5×5 morphological kernel and inpainted using Telea’s fast marching method [Telea, 2004]. Median filtering with a 5×5 kernel was then applied to reduce isolated residual artifacts while preserving boundaries. Finally, CLAHE was applied to the luminance channel in LAB color space with clip limit 2.0 and tile grid size 8×8 . Four preprocessing conditions were evaluated: raw images, median-filtered images, CLAHE-enhanced images, and the full artifact-reduction pipeline. This design allowed the effect of each preprocessing regime to be compared under a common model and evaluation protocol. These values follow common defaults in the ultrasound and general image-restoration literature (e.g., typical CLAHE clip-limit ranges and Telea’s original parameterization) and were checked on a held-out validation subset to confirm they did not distort lesion boundaries before being fixed for all experiments.

3.3 Classification Architectures

Three CNN backbones were evaluated: VGG-16, ResNet-50, and EfficientNetV2S. The backbones were paired with a common attention-augmented classification head to reduce differences introduced by the final classifier. Attention was implemented using a convolutional block attention module (CBAM) [Woo et al., 2018], which combines channel and spatial attention. For a feature tensor F , channel attention was computed using global average pooling followed by a two-layer multilayer perceptron:

$$s = \sigma(W_2 \text{ReLU}(W_1 \text{GAP}(F))), \quad \tilde{F} = s \odot F, \quad (2)$$

where σ denotes the sigmoid function and $\odot$ denotes element-wise multiplication. Spatial attention was then computed using a 7×7 convolution over concatenated average-pooled and max-pooled feature maps. The resulting representation was passed through batch-normalized fully connected layers of sizes 512 and 256 with dropout probability $p = 0.3$, followed by a sigmoid output neuron.

3.4 Training and Evaluation Protocol

All models were trained using the Adam optimizer with transfer learning from ImageNet-pretrained weights. The backbone was initially frozen while the classification head was trained with learning rate 10^{-4} ; subsequent fine-tuning used learning rate 10^{-5} . Training augmentation included horizontal flipping, rotations up to $\pm 40^\circ$, and color jitter. Class imbalance was addressed using oversampling within the training set only. Because missing malignant cases is clinically more costly than false-positive referral, training used a sensitivity-oriented loss combining focal loss [Lin et al., 2017] with a false-negative penalty:

$$\mathcal{L} = -\alpha_t(1 - p_t)^\gamma \log(p_t) + \lambda_{\text{FN}} \mathbb{E}[y(1 - \hat{p})], \quad (3)$$

where p_t is the probability assigned to the true class, $\hat{p}$ is the predicted malignancy probability, $y \in \{0, 1\}$ is the ground-truth label, $\alpha_t = 0.25$, $\gamma = 2.0$, and $\lambda_{\text{FN}} = 6.0$. Decision thresholds were selected using Youden's index on the ROC curve. Performance was reported using AUC, sensitivity, specificity, and F1-score.

3.5 Ensemble Methods

ResNet-50 and EfficientNetV2S were selected for ensemble analysis because their individual results showed complementary behavior: ResNet-50 tended to improve malignant-case detection, whereas EfficientNetV2S tended to preserve specificity. Four fusion strategies were evaluated.

Averaging. The predicted malignancy probability was computed as

$$\hat{p}_{\text{avg}} = \frac{\hat{p}_{\text{R}} + \hat{p}_{\text{E}}}{2}, \quad (4)$$

where $\hat{p}_{\text{R}}$ and $\hat{p}_{\text{E}}$ denote the ResNet-50 and EfficientNetV2S predictions.

Weighted fusion. Learned softmax-normalized weights were used to combine probabilities:

$$\hat{p}_{\text{w}} = w_{\text{R}}\hat{p}_{\text{R}} + w_{\text{E}}\hat{p}_{\text{E}}, \quad w_{\text{R}} + w_{\text{E}} = 1. \quad (5)$$

Stacked fusion. A two-layer meta-classifier with architecture $2 \rightarrow 64 \rightarrow 32 \rightarrow 1$ was trained on frozen base-model predictions.

Attention fusion. Feature-level fusion was implemented using multi-head attention with eight heads. ResNet-50 and EfficientNetV2S feature vectors were projected into a common embedding space, and cross-attention was used to learn task-dependent interactions before classification. This design allows the fused model to weight complementary representations differently across images rather than using a fixed probability average. More details are shown in Figure 1.

4 Results

4.1 Dataset Audit

Table 5, added at the end of this paper, summarizes the dataset audit. The results show that non-anatomical content is common across public breast ultrasound datasets. BUSDB2023 contained the largest number of

duplicate or near-duplicate groups and color Doppler overlays. BUS-UCLM contained color Doppler overlays in all audited images, and BRA contained measurement markers in all audited images. These findings indicate that image artifacts should be treated as a central part of the experimental design rather than as incidental noise. The audit has two implications for model evaluation. First, duplicate and near-duplicate images can create leakage if visually similar samples are split across training and testing. Second, overlays and markers may act as label-correlated shortcuts. Even if such cues are clinically meaningful in some contexts, their inconsistent presence across datasets makes them unreliable for a B-mode lesion-classification model intended for deployment across institutions.

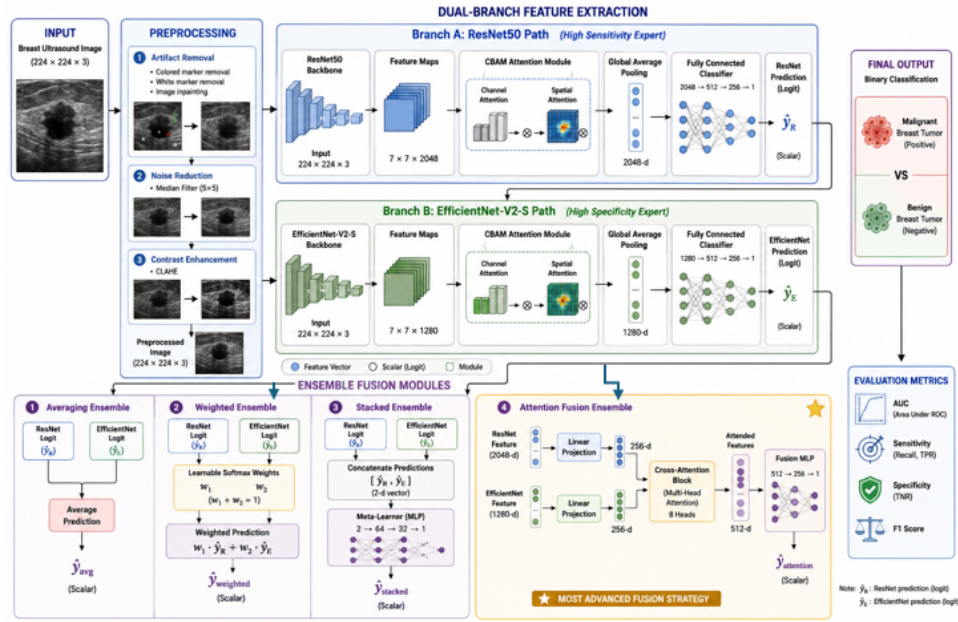


Figure 1: Architecture of the proposed dual-branch ensemble pipeline for breast ultrasound classification. The dual-branch design, backbone selection, CBAM attention integration, and all four fusion strategies were conceived and specified entirely by the authors; a generative AI diagramming tool was used only to assist with rendering and visually refining the final figure layout. Raw ultrasound images are first processed using the three-stage preprocessing pipeline described in Section 3: HSV-based segmentation and inpainting to remove colored markers and Doppler overlays, 5×5 median filtering to suppress residual noise, and CLAHE-based contrast enhancement to improve consistency in tissue appearance across the pooled source datasets. Each preprocessed image is then passed through two parallel convolutional backbones, ResNet-50 and EfficientNet-V2-S. Both branches incorporate a CBAM module for combined channel and spatial attention, followed by global average pooling and a dense classification head. The outputs of the two branches are subsequently combined using one of four fusion strategies evaluated in this study: simple averaging, learned weighted averaging, a trained stacking meta-classifier, or cross-attention feature fusion. The resulting fused output produces the final benign/malignant classification. Fusion performance is evaluated using AUC, sensitivity, specificity, and F1-score, as reported in Table 4.

4.2 Ensemble Results

Table 4 compares the ensemble strategies. All ensemble methods improved at least one clinically relevant metric relative to the individual base models, supporting the hypothesis that ResNet-50 and EfficientNetV2S provide complementary information.

4.2.1 Inference Latency

To assess clinical deployment feasibility, inference latency was benchmarked for each backbone and fusion strategy on a CUDA GPU, using a warm-up period followed by 100 timed forward passes per model (batch size 1, 224×224 input). Results are reported in Table 3. As expected, ResNet50 alone is fastest, and attention fusion

Table 3: Inference latency per model/ensemble strategy (CUDA, batch size 1).

Model	Mean (ms/image)	Median (ms/image)	Throughput (img/s)
ResNet50	6.58	6.49	151.9
EfficientNetV2S	17.99	17.61	55.6
Averaging Ensemble	24.97	24.37	40.0
Weighted Ensemble	23.98	23.88	41.7
Stacked Ensemble	24.50	24.50	40.8
Attention Fusion	24.83	24.64	40.3

is among the more computationally expensive strategies due to the added cross-attention computation, though all ensemble variants remain within a similar range (24–25 ms/image) since backbone inference dominates total latency. These figures reflect GPU (CUDA) latency; CPU-only deployment timing was not evaluated in this study and is left for future work.

Table 4: Ensemble comparison using ResNet-50 and EfficientNetV2S. Attention fusion achieved the highest sensitivity and F1-score; weighted fusion achieved the highest specificity.

Model	AUC	Sensitivity	Specificity	F1
ResNet-50	0.777	0.735	0.682	0.692
EfficientNetV2S	0.810	0.755	0.695	0.708
Averaging ensemble	0.811	0.776	0.714	0.712
Weighted ensemble	0.822	0.745	0.799	0.726
Stacked ensemble	0.813	0.806	0.701	0.722
Attention fusion	0.823	0.888	0.662	0.733

Simple averaging improved sensitivity, specificity, and F1-score relative to both base models, indicating partially independent error patterns. Weighted fusion achieved the strongest specificity, 0.799, while preserving AUC 0.822. This operating point is better suited to confirmation settings in which false positives should be reduced. Stacked fusion increased sensitivity to 0.806, but it did not match the AUC or specificity of weighted fusion. Attention fusion produced the most screening-oriented behavior, with sensitivity 0.888 and F1-score 0.733. This sensitivity represents a substantial improvement over the individual ResNet-50 and EfficientNetV2S models. The trade-off is lower specificity than weighted fusion, which may be acceptable when the system is used as a second reader or triage aid rather than as an autonomous diagnostic device. This ablation reflects preprocessing sensitivity at inference time only, using backbones trained on the full pipeline rather than retrained per condition. We attempted a fully independent ablation, retraining each backbone from scratch under all four preprocessing conditions, but this coincided with corrections to the dataset composition and an upgrade of the Python/deep-learning library environment. The resulting AUC values were inflated and inconsistent across conditions, itself illustrating this paper’s central argument: reported AUC is sensitive to the full experimental pipeline, including dataset version and software environment, not only to architecture and preprocessing choice. Since these runs were not obtained under a fixed, reproducible pipeline, we omit their numeric values to avoid overstating the ablation’s conclusions. A corrected ablation, run under a single fixed dataset version and environment, is planned as immediate follow-up work.

5 Discussion and Conclusion

This study demonstrates that robust breast ultrasound classification requires a holistic pipeline approach rather than isolated model selection, as systematic dataset audits revealed that public cohorts contain significant non-anatomical artifacts (e.g., duplicates, calipers, and Doppler overlays). Our experiments show that artifact-suppressing preprocessing shifts clinical operating characteristics as strongly as backbone

Table 5: Combined Ultrasound Public Dataset audit results. Counts summarize visible duplicate or near-duplicate groups, color Doppler overlays, and measurement markers identified during quality control.

Metric	BrEaST/TCIA	BUSI	BUSDB	UDIAT	OASBUD	BUS-UCLM	Y.Guo	BRA
Total images	252	780	232	163	100	264	192	1875
Cancer (%)	38.9	32.5	53.0	33.1	52.0	34.1	12.0	32.4
Duplicates	0	98	1	4	0	21	5	0
Doppler overlays	0	6	0	0	0	36	2	0
Measurement markers	0	749	361	0	101	11	5	1875

selection, revealing whether networks rely on true lesion morphology or shortcut signals. Furthermore, leveraging the complementary inductive biases of ResNet-50 (which enhances malignancy detection) and EfficientNetV2S (which preserves specificity) through learned fusion yields clinically tailorable workflows; specifically, cross-attention fusion maximizes sensitivity whereas weighted fusion optimizes specificity. While these findings underscore the value of pipeline-wide optimization, several limitations remain. This work relies on retrospective public datasets, employs heuristic parameters for preprocessing (such as CLAHE limits) and artifact detection, and introduces additional computational overhead via attention-based fusion. Our attempted preprocessing ablation (Section 4) further underscored this fragility in practice, as dataset and software-environment changes alone produced unstable AUC shifts comparable in magnitude to those attributed to preprocessing itself. Additionally, despite incorporating CBAM and attention mechanisms, the interpretability of these ensemble decisions requires further refinement prior to clinical adoption. Future work will validate this framework on prospective multicenter cohorts with consistent acquisition metadata, develop adaptive quality-control models, incorporate calibration-aware thresholding, extend the system from binary to multi-class, BIRADs-oriented decision support, and conduct a fully independent per-condition preprocessing ablation that re-trains each backbone separately to isolate its contribution to performance. We are also collecting a new breast ultrasound dataset with a clinical partner hospital – prospectively acquired, single-institution, with consistent imaging protocols and verified groundtruth labels – to validate this framework beyond the retrospective public datasets used here.

References

- [Abbasian Ardakani et al., 2023] Abbasian Ardakani, A., Mohammadi, A., Mirza-Aghazadeh-Attari, M., and Acharya, U. R. (2023). An open-access breast lesion ultrasound image database: Applicable in artificial intelligence studies. *Computers in Biology and Medicine*, 152:106438.
- [Al-Dhabyani et al., 2020] Al-Dhabyani, W., Gomaa, M., Khaled, H., and Fahmy, A. (2020). Dataset of breast ultrasound images. *Data in Brief*, 28:104863.
- [Gómez-Flores et al., 2019] Gómez-Flores, W., Gregorio-Calas, M. J., and Pereira, W. C. d. A. (2019). Bus-bra: A breast ultrasound dataset for assessing computer-aided diagnosis systems. *Medical Physics*, 46(9):4058–4067.
- [Guo et al., 2021] Guo, Y., Duan, X., Wang, C., and Guo, H. (2021). Segmentation and recognition of breast ultrasound images based on an expanded u-net. *PLOS ONE*, 16(6):e0253202.
- [He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.
- [Koshy et al., 2026] Koshy, S. S., Anbarasi, L. J., Narendra, M., and Singh, R. K. (2026). Hed-net: a hybrid ensemble deep learning framework for breast ultrasound image classification. *Frontiers in Artificial Intelligence*, 8. Published 23 January 2026.

- [Lee and Choi, 2025] Lee, H. and Choi, M. (2025). Efficientnet for breast ultrasound classification focusing on calcifications. *Scientific Reports*, 15:27149.
- [Lin et al., 2017] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988.
- [Pawłowska et al., 2024] Pawłowska, A., 'Cwierz Pie'nkowska, A., Domalik, A., Jagu's, D., Kasprzak, P., Matkowski, R., Fura, Ł., Nowicki, A., and Zolek, N. (2024). A Curated Benchmark Dataset for Ultrasound Based Breast Lesion Analysis (Breast-Lesions-USG). Dataset.
- [Piotrkowska-Wróblewska et al., 2017] Piotrkowska-Wróblewska, H., Dobruch-Sobczak, K., Byra, M., and Nowicki, A. (2017). Open access database of raw ultrasonic signals acquired from malignant and benign breast lesions. *Medical Physics*, 44(12):6105–6109.
- [Simonyan and Zisserman, 2015] Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*.
- [Telea, 2004] Telea, A. (2004). An image inpainting technique based on the fast marching method. *Journal of Graphics Tools*, 9(1):23–34.
- [Thomas et al., 2023] Thomas, C., Byra, M., Marti, R., Yap, M. H., and Zwiggelaar, R. (2023). Bus-set: A benchmark for quantitative evaluation of breast ultrasound segmentation networks with public datasets. *Medical Physics*, 50(6):3574–3588.
- [Vallez et al., 2025] Vallez, N., Bueno, G., Deniz, O., Rienda, M. A., and Pastor, C. (2025). Bus-uclm: Breast ultrasound lesion segmentation dataset. *Scientific Data*, 12(1):242.
- [Woo et al., 2018] Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). Cbam: Convolutional block attention module. In *European Conference on Computer Vision (ECCV)*, pages 3–19.
- [World Health Organization, 2023] World Health Organization (2023). Breast cancer. <https://www.who.int/news-room/fact-sheets/detail/breast-cancer>. Accessed: 2025-01.
- [Yap et al., 2018] Yap, M. H., Pons, G., Marti, J., Ganau, S., Sentis, M., Zwiggelaar, R., Davison, A. K., and Marti, R. (2018). Automated breast ultrasound lesions detection using convolutional neural networks. *IEEE Journal of Biomedical and Health Informatics*, 22(4):1218–1226.
- [Zhang et al., 2020a] Zhang, H., Han, L., Chen, K., Peng, Y., and Lin, J. (2020a). Diagnostic efficiency of the breast ultrasound computer-aided prediction model based on convolutional neural network in breast cancer. *Journal of Cancer*, 11(18):5299–5307.
- [Zhang et al., 2020b] Zhang, H., Han, L., Chen, K., Peng, Y., and Lin, J. (2020b). Diagnostic efficiency of the breast ultrasound computer-aided prediction model based on convolutional neural network in breast cancer. *Computational and Mathematical Methods in Medicine*, 2020:Article ID 8834860.

MicroBird: Embedded Bird-Audio Classification with Microjoule-Range CNN Inference on the MAX78002

Justyna Przyborska Salaheddin Alakkari

Immersive Software Engineering, University of Limerick, Ireland

justyna.przyborska22@gmail.com Salaheddin.Alakkari@ul.ie

Abstract

Automated bird-audio classifiers can support passive acoustic monitoring, but many strong systems rely on cloud inference, global taxonomies, or watt-class edge hardware that is difficult to deploy at scale. This paper presents MicroBird, a region-specific 51-class bird-audio classifier deployed on the MAX78002 microcontroller CNN accelerator for Irish species monitoring. The dataset pipeline derives a regional species list from public occurrence sources, gathers focal recordings from Xeno-Canto, extracts 3-second clips, applies RMS gating and BirdNET teacher filtering, and converts accepted clips to 64×128 log-mel spectrograms. A compact hardware-aware CNN is trained with quantization-aware training (QAT) and synthesized using the ai8x toolchain. On a held-out Xeno-Canto-derived test split of 3,034 recording-ID-disjoint clips, the deployed end-to-end path achieves 79.10% top-1 accuracy, 90.90% top-3 accuracy, macro- F_1 of 0.792, and weighted- F_1 of 0.789. CNN inference latency is 3.5 ms; the full spectrogram-plus-CNN compute path requires 218.8 ms per 3-second clip. From measured durations and fixed power coefficients, CNN inference is estimated at 15.3 μ J and the full compute path at 2.168 mJ. The frontend accounts for approximately 99.3% of estimated compute energy, revealing how accelerator-backed classification shifts the system bottleneck from CNN inference to feature extraction. To the best of our knowledge, this is among the first reports of a deployed multi-class bird-audio classifier with microjoule-scale CNN inference on this hardware class. Evaluation is limited to held-out public-audio clips; long-duration field deployment remains future work.

Keywords: bioacoustics, embedded AI, bird classification, TinyML, MAX78002, passive acoustic monitoring, quantization-aware training

1 Introduction

Bird populations are widely used as indicators of ecosystem condition because changes in abundance and distribution reflect broader pressures such as land-use change, habitat degradation, and climate change [Gregory et al., 2005]. Passive acoustic monitoring extends this idea by allowing autonomous recorders to collect long-duration environmental audio without continuous human survey effort [Shonfield and Bayne, 2017, Hill et al., 2018]. For dense monitoring networks, raw audio alone is not enough: useful deployments also need local species-level screening that limits storage, bandwidth, and maintenance cost.

Large-scale bird-audio classifiers such as BirdNET show that spectrogram CNNs can recognise broad species sets with strong accuracy [Kahl et al., 2021]. However, many monitoring deployments do not require a global classifier; they require reliable screening of locally plausible species on hardware that can remain in the field. MicroBird addresses this narrower setting with a regional Irish species set and a classifier designed for deployment on a microcontroller-class device rather than a single-board computer.

MicroBird tests whether a regional class set and hardware-aware CNN can provide useful multi-class bird-audio accuracy at much lower embedded compute cost. The target hardware is the MAX78002, a microcontroller with an Arm Cortex-M4 host processor and a dedicated low-power CNN accelerator. The accelerator can execute quantized convolutional networks efficiently, but only when the architecture fits the ai8x synthesis flow, supported operators, memory layout, and layer scheduling constraints [Analog Devices, 2022a, Analog Devices, 2026b, Analog Devices, 2026a]. This makes quantization-aware training, spectrogram design, and firmware evaluation part of the same deployment problem rather than separate offline steps.

Accordingly, the evaluation considers model accuracy alongside stage-level latency and estimated compute energy to identify where optimisation effort has the greatest system-level effect.

This paper makes four contributions:

- A reproducible regional bird-audio dataset pipeline using public occurrence/audio sources and BirdNET-guided clip filtering.
- A 51-class hardware-aware spectrogram CNN trained with QAT and synthesized for the MAX78002.
- A deployed embedded pipeline with Cortex-M4 spectrogram generation and MAX78002 CNN accelerator inference.
- An empirical accuracy, latency, and duration-based energy evaluation showing useful multi-class accuracy and that frontend spectrogram computation dominates the deployed compute cost.

2 Related Work

Bird-audio recognition has shifted from hand-engineered acoustic features to spectrogram-based deep learning. BirdCLEF helped establish large-scale public evaluation for avian sound recognition, and by 2016 CNN-based spectrogram systems were already outperforming classical pipelines [Goëau et al., 2016, Sprengel et al., 2016]. BirdNET is the most relevant large-scale reference system: it uses deep spectrogram models to identify hundreds of North American and European species and has become a widely used tool for avian diversity monitoring [Kahl et al., 2021].

Xeno-Canto provides large numbers of community recordings with useful metadata, but labels are typically at recording level and recordings vary in duration, quality, background noise, and co-occurring species [Vellinga and Planqué, 2015]. The BirdSet benchmark paper highlights a broader reproducibility issue in avian bioacoustics: datasets, preprocessing choices, task definitions, and evaluation boundaries are often difficult to compare across studies [Rauch et al., 2025]. MicroBird therefore reports the regional species list, clip filtering rules, spectrogram settings, and recording-ID group split explicitly.

Low-power monitoring systems occupy a different design space from high-capacity server or cloud models. AudioMoth made low-cost acoustic logging practical, but it is primarily a recorder rather than a deployed multi-species classifier [Hill et al., 2018]. Bird@Edge moves recognition closer to the recording site using edge hardware and a distributed monitoring setup, but remains oriented toward more capable edge platforms rather than a microcontroller CNN accelerator [Höchst et al., 2022]. TinyChirp demonstrates that bird-song screening can run on an nRF52840-class microcontroller, but its reported task is binary corn-bunting detection rather than regional multi-species classification [Huang et al., 2024]. WrenNet is closest in spirit to MicroBird because it targets 70-species bird classification on low-power bioacoustic loggers and reports deployment on AudioMoth [Ciapponi et al., 2026]. MicroBird differs by using a dedicated microcontroller CNN accelerator, a 51-class Irish species set, and an explicit split between frontend spectrogram cost and CNN inference cost.

3 Method

3.1 Dataset Construction

The class set contains 50 Irish bird species plus `non_bird`. Candidate species were selected from eBird occurrence data and cross-referenced with BirdWatch Ireland monitoring resources [Sullivan et al., 2009, Lewis et al., 2020], then represented in JSON lists with numeric identifiers, common names, and scientific names. The pipeline is intended to scale to larger regional lists, but the deployed system evaluated here uses this 51-class task.

Recordings were collected from Xeno-Canto using species-level metadata manifests. Candidate recordings with quality ratings A–C were retained, species were capped at 350 recordings, and no more than 50 recordings per recordist were selected to reduce over-representation of a single recording style or environment. Recordings from Ireland, the UK, and nearby regions were preferred.

Each recording was split into non-overlapping 3-second clips after skipping the first second. An adaptive root-mean-square (RMS) gate discarded low-energy windows using the louder of -40 dBFS and the recording’s

30th RMS percentile. Remaining windows were classified with BirdNET as a teacher. A window was accepted as the target species when BirdNET’s top detection matched that species with confidence above 0.3. If no bird was detected above a 0.1 general bird threshold, the window was retained as `non_bird`; if another bird species was detected, the window was dropped. The `non_bird` class is therefore an initial rejection class derived from low-confidence public-recording windows, not a comprehensive environmental soundscape class.

The final spectrogram dataset contains 30,201 clips, split 80/10/10 into 24,147 train, 3,020 validation, and 3,034 test clips. Splitting was grouped by Xeno-Canto recording ID so that clips from the same source recording did not appear in multiple partitions.

3.2 Preprocessing

Audio was loaded as 16 kHz mono. Each 3-second clip was converted to a log-mel spectrogram using a 1024-sample short-time Fourier transform (STFT) window, hop length 256, 128 mel bins from 120 Hz to 8000 Hz, dB scaling relative to the clip maximum, and clamping to $[-80, 0]$ dB. Spectrograms were resized to 64×128 and normalised with a per-sample transform that subtracts the sample mean, divides by sample standard deviation, clamps to $[-3, 3]$, and rescales to $[-1, 1]$.

Training augmentations included SpecAugment-style time and frequency masking [Park et al., 2019], small time shifts, additive Gaussian noise, and mixup during floating-point pre-training only [Zhang et al., 2018]. Mixup was disabled during QAT so that quantization statistics were collected from real, unblended inputs.

3.3 Model, Training, and Deployment

A small architecture search selected the MicroBird model under the MAX78002/ai8x deployment constraints. Candidate models varied base channel width, additional convolution blocks, and residual connections while using the same split, preprocessing, class weighting, augmentation policy, and Adam training schedule. Validation accuracy improved substantially from 8- and 16-channel baselines to 32-channel models; adding moderate depth was more parameter-efficient than width alone at 16 channels, while residual variants did not improve QAT validation accuracy in this setting. The final model was selected because it provided the strongest balance of validation accuracy, synthesis compatibility, and deployment margin.

The MicroBird model is an ai8x-compatible CNN for a single-channel 64×128 spectrogram. It uses a 3×3 stem convolution, three fused max-pool-convolution stages, one additional non-residual 3×3 block, a 1×1 head convolution, fixed average pooling over the final 8×16 feature map, and a 51-output classifier. The synthesized model has seven accelerator layers, 393,160 parameters, and approximately 384 KB of 8-bit weights. Floating-point training used learning rate 10^{-3} ; QAT began at epoch 30 with learning rate 10^{-4} , 8-bit weights, and best-checkpoint selection by validation loss. QAT is used here as a deployment requirement, not only as post-training compression, because integer arithmetic and ai8x layer constraints affect the deployable network.

The selected checkpoint was synthesized using ai8x and integrated into MAX78002 firmware. The Cortex-M4 computes the spectrogram frontend using CMSIS-DSP FFT routines, precomputed Hann windows and mel filters, dB scaling, resizing, int8 packing, and per-sample normalisation [Arm Ltd., 2025]. The CNN accelerator then executes the quantized network and returns top- k predictions. Figure 1 summarises this deployed path. Two evaluation paths were supported: `LOAD_PCM`, which uploads 3-second pulse-code modulation (PCM) audio and exercises the full frontend-plus-CNN path, and `LOAD_SPEC`, which bypasses spectrogram generation for debugging.

3.4 Evaluation

Final accuracy was measured by replaying the full held-out test split through the deployed UART/API evaluation path. Each 3-second PCM audio clip was sent to the board over UART; the firmware then ran the Cortex-M4 spectrogram frontend, executed the CNN accelerator, and returned top- k predictions. The host API was used only for replay and collection, so host-side transfer and control overheads are excluded from the

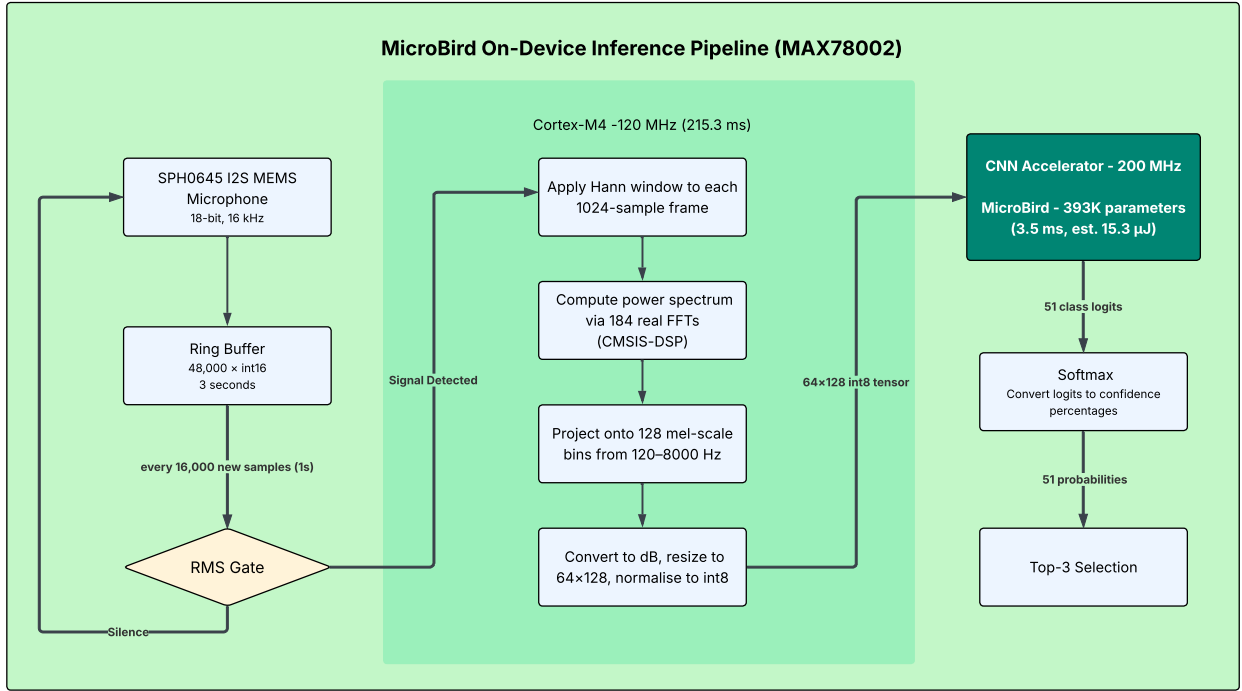


Figure 1: MicroBird deployed inference path. The Cortex-M4 computes the 64×128 log-mel frontend from 3-second audio, while the MAX78002 CNN accelerator runs the quantized classifier and returns top- k predictions. Evaluation replays equivalent 3-second PCM clips over UART.

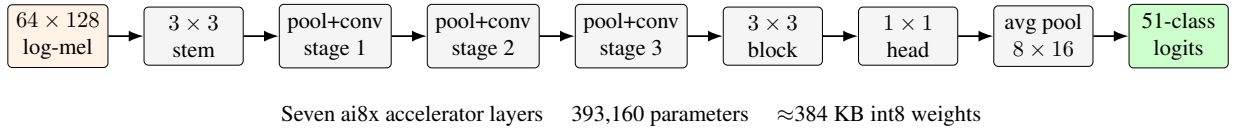


Figure 2: MicroBird CNN architecture synthesized for the MAX78002. The deployed network contains seven accelerator layers, 393,160 parameters, and approximately 384 KB of 8-bit weights.

embedded latency and energy figures. CNN latency was measured around accelerator execution, and spectrogram frontend latency was measured around the Cortex-M4 preprocessing path. The firmware also exposes MAX78002EVKIT PMON trigger windows for system and CNN stages following the vendor benchmarking flow [Analog Devices, 2022b]. Raw PMON serial logs were captured for representative validation runs, but a verified scalar energy value was not obtained from the PMON stream for all reported stages. Consequently, the scalar energy values reported below remain duration-based estimates from measured stage durations and fixed power coefficients. They include spectrogram frontend compute and CNN accelerator inference only, excluding microphone duty cycle, storage, radio transmission, host-side upload overhead, display, and idle-system energy.

4 Results

4.1 Deployed Accuracy

The deployed system achieved 79.10% top-1 and 90.90% top-3 accuracy on the held-out Xeno-Canto-derived test split (Table 1). Macro- F_1 and weighted- F_1 are close, indicating that the result is not driven only by the most frequent classes.

Table 1: Final deployed MicroBird performance on the held-out test split.

Metric	Value
Test clips	3,034
Classes	51
Top-1 accuracy	79.10%
Top-3 accuracy	90.90%
Macro- F_1	0.792
Weighted- F_1	0.789

Per-class results show that errors are concentrated rather than uniform (Table 2). Acoustically distinctive species such as sparrowhawk, kingfisher, buzzard, swift, and stonechat exceed 96% recall. The weakest class is `non_bird`, with 23.33% recall and 29.58% precision, confirming that it should be treated as an initial rejection class rather than a general environmental sound class.

Table 2: Best and worst deployed per-class recall values.

Class	Common name	Recall	n
Best classes			
<i>Accipiter nisus</i>	Sparrowhawk	100.00%	23
<i>Alcedo atthis</i>	Kingfisher	97.06%	34
<i>Buteo buteo</i>	Common Buzzard	96.92%	65
<i>Apus apus</i>	Common Swift	96.36%	55
<i>Saxicola rubicola</i>	Stonechat	96.05%	76
Worst classes			
<code>non_bird</code>	Initial rejection	23.33%	90
<i>Prunella modularis</i>	Dunnock	56.16%	73
<i>Emberiza schoeniclus</i>	Reed Bunting	56.86%	51
<i>Anthus pratensis</i>	Meadow Pipit	57.78%	45
<i>Turdus philomelos</i>	Song Thrush	58.18%	55

Table 3: Most frequent bird-to-bird confusion pairs in the deployed test run.

True class	Predicted class	Count
Dunnock	Kingfisher	10
House Sparrow	House Martin	9
Reed Bunting	Yellowhammer	7
Rook	Hooded Crow	6
Wood Pigeon	Collared Dove	6

4.2 Latency and Duration-Based Energy

Table 4 separates the measured latency and estimated energy of the frontend and CNN stages. CNN inference is cheap at 3.5 ms and an estimated 15.3 μ J. The full spectrogram-plus-CNN compute path is 218.8 ms and an estimated 2.168 mJ per 3-second clip. The frontend accounts for approximately 99.3% of the estimated spec-plus-CNN energy, indicating that spectrogram generation, rather than CNN inference, is the dominant optimisation target. This marks a system-level bottleneck shift in which accelerator-backed classification is no longer the dominant cost, making frontend optimisation more consequential than further CNN compression.

Table 4: Measured latency and duration-based energy estimates. Raw PMON trigger logs were captured for audit, but the scalar energy values here are estimated from measured duration and fixed coefficients.

Stage	Latency	Energy type	Energy
Spectrogram Cortex-M4	215.3 ms	Duration-based estimate	2.153 mJ
CNN accelerator	3.5 ms	Duration-based estimate	15.3 μ J
Total spec+CNN	218.8 ms	Duration-based estimate	2.168 mJ

4.3 Indicative Prior-Work Comparison

Table 5 compares MicroBird with BirdNET [Kahl et al., 2021], TinyChirp [Huang et al., 2024], and WrenNet [Ciapponi et al., 2026]. The comparison is indicative and not benchmark-equivalent because tasks, class sets, clip lengths, hardware, and measurement boundaries differ. MicroBird’s energy value includes spectrogram frontend compute and CNN inference, while the CNN-only estimate is 15.3 μ J.

Table 5: Indicative comparison with related bird-audio systems. Comparisons are not benchmark-equivalent because tasks, metrics, clip lengths, hardware, and measurement boundaries differ.

System	Task	Hardware	Reported result	Energy boundary	Energy/clip
BirdNET	984 spp.	RPi 3B+	Global reference model	Frontend incl.; WrenNet bench.	2,790 mJ
TinyChirp	Binary	nRF52840	Precision >0.98; corn bunting detection	Frontend incl.; reported	42.5 mJ
WrenNet	70 spp.	AudioMoth	70.1% full-task accuracy	Frontend incl.; reported	77 mJ
MicroBird	51 classes	MAX78002	79.10% top-1; 90.90% top-3	Frontend incl.; estimate	2.168 mJ

5 Limitations and Future Work

The main limitation is evaluation scope. A held-out focal-recording test split is a common and useful stage for bird-audio model evaluation, and the split used here is recording-ID-disjoint. However, it is still derived from Xeno-Canto clips rather than from a long-duration outdoor field deployment. The reported accuracy is therefore evidence for deployment feasibility on the target hardware, not a final estimate of long-term field performance under weather, microphone placement, seasonal change, and persistent background noise.

The `non_bird` class is weak and should be expanded with a richer environmental corpus before being treated as robust background rejection. The scalar energy values are duration-based estimates rather than independently parsed PMON energy measurements. Finally, prior-work comparisons are useful for positioning but not benchmark-equivalent.

Future work should prioritise richer non-bird data, long-duration field trials, recordist- or location-disjoint evaluation, fleet-level deployment studies, and frontend optimisation. Since the CNN is already a small fraction of the estimated compute energy, sliding-window reuse, partial STFT reuse, or lower-cost features may improve total system cost more than further shrinking the classifier. The same curation pipeline can also be used to test larger regional species lists under the same hardware constraints.

6 Conclusion

MicroBird demonstrates that a 51-class regional bird-audio classifier can be deployed end to end on the MAX78002 with useful held-out accuracy, microjoule-scale CNN inference estimate, millijoule-scale full compute estimate, and a clear bottleneck shift from classifier inference to spectrogram frontend computation. The result supports hardware-aware regional classification as a practical direction for low-power acoustic monitoring, while leaving field validation and stronger non-bird rejection as necessary next steps.

References

- [Analog Devices, 2022a] Analog Devices (2022a). MAX78002: Artificial intelligence microcontroller with low-power convolutional neural network accelerator. <https://www.analog.com/media/en/technical-documentation/data-sheets/max78002.pdf>. Datasheet, accessed 2026-04-03.
- [Analog Devices, 2022b] Analog Devices (2022b). MAX7800x power monitor and energy benchmarking guide. <https://www.analog.com/media/en/technical-documentation/user-guides/max7800x-power-monitor-and-energy-benchmarking-guide.pdf>. User guide, accessed 2026-04-03.
- [Analog Devices, 2026a] Analog Devices (2026a). ai8x-synthesis: Network synthesis for MAX78000/MAX78002. <https://github.com/analogdevicesinc/ai8x-synthesis>. GitHub repository, accessed 2026-04-03.
- [Analog Devices, 2026b] Analog Devices (2026b). ai8x-training: Training and quantization for MAX78000/MAX78002. <https://github.com/analogdevicesinc/ai8x-training>. GitHub repository, accessed 2026-04-03.
- [Arm Ltd., 2025] Arm Ltd. (2025). CMSIS-DSP software library. <https://arm-software.github.io/CMSIS-DSP/latest/>. Documentation, accessed 2026-04-03.
- [Ciapponi et al., 2026] Ciapponi, S., Mannini, L., Scanferla, J., Anderle, M., and Farella, E. (2026). Enabling multi-species bird classification on low-power bioacoustic loggers. In *Proceedings of the 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. <https://doi.org/10.1109/ICASSP55912.2026.11463679>.
- [Goëau et al., 2016] Goëau, H., Glotin, H., Vellinga, W.-P., Planqué, R., and Joly, A. (2016). LifeCLEF bird identification task 2016: The arrival of deep learning. In *Working Notes of CLEF 2016 – Conference and Labs of the Evaluation Forum*, Évora, Portugal.
- [Gregory et al., 2005] Gregory, R. D., van Strien, A., Voříšek, P., Meyling, A. W. G., Noble, D. G., Foppen, R. P. B., and Gibbons, D. W. (2005). Developing indicators for european birds. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 360(1454):269–288.
- [Hill et al., 2018] Hill, A. P., Prince, P., Piña Covarrubias, E., Doncaster, C. P., Snaddon, J. L., and Rogers, A. (2018). AudioMoth: Evaluation of a smart open acoustic device for monitoring biodiversity and the environment. *Methods in Ecology and Evolution*, 9(5):1199–1211.
- [Höchst et al., 2022] Höchst, J., Bellafkir, H., Lampe, P., Vogelbacher, M., Mühling, M., Schneider, D., Lindner, K., Rösner, S., Schabo, D. G., Farwig, N., and Freisleben, B. (2022). Bird@Edge: Bird species recognition at the edge. In *Networked Systems – 10th International Conference, NETYS 2022*, volume 13464 of *Lecture Notes in Computer Science*, pages 69–86. Springer.

- [Huang et al., 2024] Huang, Z., Tousnakhoff, A., Kozyr, P., Rehausen, R., Bießmann, F., Lachlan, R., Adjih, C., and Baccelli, E. (2024). TinyChirp: Bird song recognition using TinyML models on low-power wireless acoustic sensors. In *2024 IEEE 5th International Symposium on the Internet of Sounds (IS2)*. IEEE.
- [Kahl et al., 2021] Kahl, S., Wood, C. M., Eibl, M., and Klinck, H. (2021). BirdNET: A deep learning solution for avian diversity monitoring. *Ecological Informatics*, 61:101236.
- [Lewis et al., 2020] Lewis, L. J., Coombes, R. H., Burke, B., Tierney, T. D., Cummins, S., Walsh, A. J., Ryan, N., and O’Halloran, J. (2020). Countryside bird survey report 1998–2019. Technical report, BirdWatch Ireland, Wicklow, Ireland.
- [Park et al., 2019] Park, D. S., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E. D., and Le, Q. V. (2019). SpecAugment: A simple data augmentation method for automatic speech recognition. In *Proceedings of Interspeech 2019*.
- [Rauch et al., 2025] Rauch, L., Schwinger, R., Wirth, M., Heinrich, R., Huseljic, D., Herde, M., Lange, J., Kahl, S., Sick, B., Tomforde, S., and Scholz, C. (2025). BirdSet: A large-scale dataset for audio classification in avian bioacoustics. In *The Thirteenth International Conference on Learning Representations (ICLR 2025)*. Spotlight.
- [Shonfield and Bayne, 2017] Shonfield, J. and Bayne, E. M. (2017). Autonomous recording units in avian ecological research: Current use and future applications. *Avian Conservation and Ecology*, 12(1):14.
- [Sprengel et al., 2016] Sprengel, E., Jaggi, M., Kilcher, Y., and Hofmann, T. (2016). Audio based bird species identification using deep learning techniques. In *Working Notes of CLEF 2016 – Conference and Labs of the Evaluation Forum*, Évora, Portugal.
- [Sullivan et al., 2009] Sullivan, B. L., Wood, C. L., Iliff, M. J., Bonney, R. E., Fink, D., and Kelling, S. (2009). eBird: A citizen-based bird observation network in the biological sciences. *Biological Conservation*, 142(10):2282–2292.
- [Vellinga and Planqué, 2015] Vellinga, W.-P. and Planqué, R. (2015). The Xeno-Canto collection and its relation to sound recognition and classification. In *Working Notes of CLEF 2015*.
- [Zhang et al., 2018] Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. (2018). mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*.

Towards Long-Term Pig Re-Identification: Investigating Human ReID Methods in Pigs

Yiyuan Wang¹, Illias Kyriazakis², Jian Gao¹, Niall McLaughlin¹

¹ School of EECS, Queen's University Belfast, UK

² School of Biological Sciences, Queen's University Belfast, UK

{ywang203, j.gao, n.mclaughlin}@qub.ac.uk, I.Kyriazakis@qub.ac.uk

Abstract

In precision livestock farming, early intervention at the individual level can improve health and welfare outcomes. However, automated long-term visual monitoring of individual pigs is challenging due to their lack of distinctive visual features and the scarcity of suitable public datasets. Existing research on this topic is limited to short time periods, of minutes or hours. Yet, to have a commercial impact, accurate monitoring over days to weeks is required. Hence, we study long-term pig re-identification, aiming to recognise individual pigs across multiple weeks despite variation in markings, weight gain, and growth. We evaluate existing human re-identification methods for this task using a newly released long-term pig tracking dataset. Our results show that while existing methods excel at same-day re-identification, there is a sharp drop-off in performance for cross-week scenarios, with many methods approaching chance performance by around six weeks. We also find that standard classification loss generally outperforms specialised re-identification loss functions, such as triplet loss, in the long-term setting.

Keywords: Long-Term Re-Identification, Pigs, Precision Livestock Farming

1 Introduction

With the advancement of precision livestock farming (PLF), it is now possible to monitor animals at the individual level rather than at the group level, allowing farmers to better understand an animal's behaviour, meet their nutritional needs, and identify health issues earlier for intervention. For instance, a pig with reduced feed intake can be identified for further checks to help the pig recover, prevent the spread of potential disease, and minimise financial loss. At the same time, farms are becoming larger whilst labour input per animal is decreasing, making manual individual observation impractical. This raises the problem of how to identify and track each individual pig over long time periods. While ear tags can be used in some circumstances, they are not practical for all scenarios [1]. To address the challenge of long-term non-invasive individual monitoring, computer vision-based multi-object tracking (MOT) algorithms have been explored [2]. However, trackers rely on continuous video input and accurate ID propagation between frames, which may provide a solution for short-term monitoring, but in real-world settings, errors accumulate due to factors such as occlusions and fast movements, leading to tracking failure [3]. A complementary approach is Re-Identification (ReID), which aims to match individuals over long time periods based purely on visual appearance in single images [4].

Re-identification methods were originally developed for matching people moving between cameras; adapting them to pigs poses several challenges. In contrast to human ReID, pig ReID is typically a single-camera task, and humans often wear distinctive clothing, but pigs tend to be visually similar, lacking clear distinguishing visual features [4]. In addition, factors such as illumination, viewing angle, skin stains, dirt, and the pig's growth introduce additional unwanted sources of appearance variation (Figure 1). Existing pig ReID benchmarks [4] are limited to short time periods of several minutes or hours and explore only a narrow range of

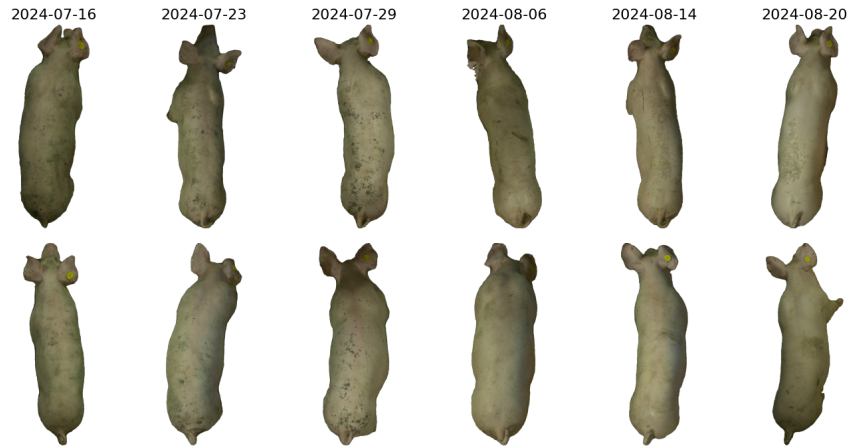


Figure 1: **Appearance variation of pigs over time. Each row represents an individual pig.** Due to the lack of distinct features, as well as minor drift of the camera angle, illuminations, stains and pigs’ growth, identification is difficult. Images extracted from [5]. The images were pre-segmented in the original dataset.

ReID training strategies, leaving long-term performance uncharacterised. Given these challenges, it remains an open question whether specialised ReID methods, mostly designed for humans or visually distinctive animals, remain effective for pigs, especially in long-term scenarios.

In this work, we leverage a recently released dataset for long-term, cross-day pig ReID [5], which provides longitudinal video recordings spanning more than one month on average and globally consistent pig identity labels across different video recording sessions. In the dataset, the pig images for ReID are pre-cropped and segmented to remove the interference of the pen background. We aim to understand how a wide range of ReID methods perform over days or weeks, extending the existing literature that largely focuses on shorter time periods. To achieve this goal, we evaluate the performance of multiple existing state-of-the-art re-identification methods over long time spans. To the best of our knowledge, this is the first time such analysis has been carried out. Hence, our work provides the community with a basis for future research on long-term pig re-identification.

2 Related Work

2.1 Existing Pig Tracking and Re-Identification Datasets

To satisfy the needs for long-term ReID, a dataset should provide i) high-quality ID annotations, such as globally consistent identity annotations across recording sessions to give indications whether a pig is the same pig in another video recording; ii) longer tracking spans of days, weeks, or months, instead of only a few minutes or hours; iii) the pigs in the dataset should not contain additional visible marks that can be exploited by the models. While several existing tracking-oriented datasets have been repurposed for pig ReID [6, 7], neither fully satisfies the requirements for long-term evaluation of pig re-identification.

The PigTrack dataset [6] contains dense labelling of each frame for pigs in diverse pen environments. However, the average monitoring span lasts only 30 seconds, which is useful for short-term tracking but less suitable for long-term ReID. In addition, we observed that some videos in the PigTrack dataset show clearly visible marks painted on the pigs’ backs, which the model can easily exploit instead of learning the pigs’ appearance characteristics. OinkTrack [7] extends the longest single-sequence monitoring time to 1 hour, emphasising the combined difficulties of occlusions and illumination changes during tracking and ReID. However, the dataset was sampled from only 2 pig pens, limiting its diversity, and the paper does not explicitly clarify whether its 574 reported identities are consistent across video sequences or only within each sequence, limiting the dataset’s applicability to ReID across recordings.

Bernotas *et al.* have recently introduced a longitudinal dataset for pig tracking and re-identification, target-

ing longer-term identity continuity [5]. It contains 20 monitoring groups, each corresponding to a fixed set of pigs housed together in a single pen. The dataset meets our key requirements: globally consistent ID labels, a longer monitoring span of 37.2 days on average (± 14.1 days, s.d.), and the absence of visible artificial marks on pigs. Instead of continuously recording, they captured videos on selected days, reducing the data volume for long-term coverage. To provide cross-video-consistent and long-term pig ID labelling, they made parallel recordings in visible and UV light. They then created unique sunscreen patterns on the backs of the pigs, visible only to UV cameras, not to the RGB camera used to capture the images in the dataset. Therefore, the ReID models must rely on the pigs' intrinsic appearance features, mimicking a realistic farm setting. Although the current public release does not contain the full original dataset due to ongoing curation, with only a small number of recordings missing, it remains suitable for our long-term evaluation of ReID performance. We note that the paper describing this dataset evaluates only multi-object tracking methods [5]. Hence, to our knowledge, we are the first to evaluate the performance of re-identification methods on this dataset.

2.2 Re-Identification Loss Functions

For ReID in the domains of pedestrians, wildlife and cattle livestock, the common loss functions used include ID losses, triplet losses, and contrastive losses [8, 9].

ID losses formulate ReID as a classification problem, typically optimised by the Cross-Entropy (CE) or a margin-based loss such as ArcFace [10]. ArcFace imposes an angular margin between the normalised feature embedding and the normalised class weight vector, encouraging intra-ID compactness and inter-ID separability. ID losses require a certain number of samples per identity to achieve optimal training performance, which can be problematic for short video clips. Testing on unseen identities is considered a disadvantage, such as unseen pigs in the test split, since the model is not explicitly optimised to separate them. Building on the ID loss framework, TAUDL extends cross-entropy for unsupervised cross-camera pedestrian ReID [11], enabling learning without clean identity labels or cross-camera annotations. However, it can also be used in supervised single-camera scenarios by seeing each individual pig group as a virtual camera and using ground-truth ID labels.

Triplet losses are metric learning methods that do not directly optimise the ID labels, but guide the appearance embedding to be closer together for positive samples and farther apart for negative samples. They do not rely on a predefined set of labels during training, and are more relaxed for unseen labels. They also directly optimise the image embedding used to measure image similarity. They are often combined with ID loss in person ReID models, such as BoT [12]. The mining strategy significantly affects the model training for triplet losses, namely how to select positive and negative pairs. For instance, hard-mining selects the hardest examples in training batches, i.e., the closest negative and the farthest positive example [13].

Contrastive losses such as the supervised contrastive loss (SupCon) [14] augment each input image twice to manually generate diverse domain differences. At each step, the loss function selects an image as an "anchor" and treats other augmented views with the same ID label as positive views. It pulls the distance between the anchor and the positive views closer and compares it against all other views except the anchor. Compared to ID losses and triplet losses, since it augments each image twice, it provides greater robustness to domain variance, but it doubles the training batch size, requiring more computing power.

3 Long-Term Pig Re-Identification

We define long-term pig re-identification as the problem of recognising an individual pig over long time periods, during which changes may occur in skin/coat markings, weight gain, and growth. We assume that pigs do not have unique artificial identifiers, such as ear tags, so that re-identification must be performed solely on visual appearance (Figure 1). The core task is to match a query image to a gallery captured days or weeks earlier and retrieve all gallery images of the same pig. Both the query and gallery may contain multiple images per individual. In commercial settings, this can be formulated as closed-set matching because the pigs within a pen are typically fixed. Accordingly, each query identity is present in the gallery, while all validation and test identities remain unseen during training. Thus, the models must generalise to unseen pigs.

Split	# Groups	# Pigs	# Pigs Per Group	Recording Interval (Days)	Recording Times
Train	12	102	8.50 ± 4.96	4.02 ± 3.03	8.85 ± 4.35
Validation	2	12	6.00 ± 0.00	7.00 ± 0.78	7.50 ± 0.50
Test	2	12	6.00 ± 0.00	7.00 ± 0.78	7.50 ± 0.50

Table 1: Statistics of our train/validation/test splits of [5].

3.1 Re-Identification Model

We use an ImageNet-pretrained ResNet50 as the feature-extraction backbone in all ReID experiments. Feature maps from the final convolutional layer are globally average-pooled and flattened to produce pig appearance descriptors. Horizontal flipping is used for data augmentation. No explicit head-to-tail orientation normalisation is applied, and the backbone is not assumed to be strictly rotation-invariant. We fine-tune the backbone with Cross-entropy (CE), ArcFace [10], Triplet loss [13], Triplet loss + CE combined (CE+Triplet) [12], SupCon [14], TAUDL and TAUDL with global header (TAUDL_G) [11].

CE, ArcFace, Triplet, CE+Triplet and SupCon Training batches are constructed by selecting P pigs with K images per pig. The backbone outputs are fed into a BNNeck [12] to normalise embeddings. During training, the cross-entropy and arcface losses are computed on features after BNNeck, while the triplet loss is computed on features before BNNeck. During evaluation, features are extracted after BNNeck. All ID losses use global classification, computed as a softmax over all pig identities in the dataset. For distance calculation, the CE model does not need to calculate any distance; ArcFace uses the product of the normalised output-layer classification weight vector and the normalised output embedding vector to compute the angle; Triplet and the CE+Triplet variant use Euclidean distance, and SupCon uses cosine similarity for the difference between the anchor and samples. We use batch-hard-mining for Triplet and CE+Triplet, and no mining for SupCon.

TAUDL Since pigs in different groups do not overlap, we treat each pig group as a separate camera, disable cross-camera matching, and directly use the dataset’s ground-truth labels during training. We adopt TAUDL’s camera-wise sampler, which in our case samples C groups and I instance images per pig. Backbone output after average pooling and ReLU activation is fed to separate classification heads for each pig group, with cross-entropy computed within each head. As a result, the model does not explicitly learn the discrimination across different pig groups. This is intentional and designed to simulate a setting in which global ID labels are unavailable across video recordings. We also evaluate a TAUDL variant with a global classification head, TAUDL_G, which quantifies the value of cross-session global ID labels.

4 Experiments

4.1 Dataset

We use a subset of the recently released Large-Scale Longitudinal Pig ReID dataset (PigReID) for our experiments [5]. In the subset, the statistics for pig groups 1 to 13, 17, 18, and 20 match those of the full dataset; groups 14, 16, 19, and 21–22 are missing; group 18 is missing one pig identity. Full details are reported in Appendix A. We manually split the dataset into train, validation, and test sets at the group level, ensuring no overlap between pigs and pens across the splits. Both the validation and test sets are selected to include 2 pig groups of 6 pigs each to ensure balanced evaluation. The time interval between each pig’s recording sessions in both sets is approximately 7 days (one week). The remaining 12 groups are used for model training. Statistics of the splits are shown in Table 1 and Appendix A.

4.2 Evaluation Protocol

To evaluate ReID performance across temporal separations between recordings, we organise evaluation pairs into *session-gap buckets*. Each pig group c is recorded n_c times during the monitoring period, producing

non-overlapping, discontinuous video sessions $v_{c,1}, v_{c,2}, \dots, v_{c,n_c}$ sorted by date in ascending order, where each session corresponds to a single video that contains images of all pigs in group c . For each group, a pair of sessions $(v_{c,i}, v_{c,j})$ with $i \leq j$ is assigned to the session-gap bucket $b = j - i$, where $v_{c,i}$ provides the query images and $v_{c,j}$ provides the gallery images. In the test and validation setting described in Section 4.1, each session is recorded 7.00 ± 0.78 days apart for each pig group, bucket 0 corresponds to a 0-day gap (same session), bucket 1 to an approximately 7-day gap, and bucket 2 to an approximately 14-day gap, etc. For bucket 0 (i.e., $i = j$), where query and gallery come from the same session, to avoid overlaps, the gallery images are sampled from the first 30% of the session frames, the query images are sampled from the last 30% of the session frames, and the middle 40% of the session frames are discarded.

For one evaluation, we randomly sample one session pair from a session-gap bucket b per pig, and draw N_q query images from $v_{c,j}$ and N_g gallery images from $v_{c,i}$. When multiple images are used, the embeddings are averaged for matching. We evaluate re-identification performance across session-gaps using the Cumulative Matching Characteristic (CMC) Rank-1 Score, defined as the proportion of queries whose highest-ranked gallery descriptor matches the query pig’s identity, with larger values indicate better performance.

4.3 Experimental Settings

For all experiments we use input images of size 384x384 pixels and a batch size of 64. Due to the specificities of contrastive learning, the batch size for SupCon changes to 128 after augmentation. For non-TAUDL models, we sample 4 instances per pig per batch. For TAUDL models, we sample 8 pig groups and 4 instances per pig per batch. Hyperparameters, including training steps, are searched based on the average CMC Rank-1 across all session-gaps in the validation set, and the final performance is reported per session-gap in the test set, averaged over 10 repetitions. Optimal hyper-parameter settings can be found at Section B. All models were trained on the AMD MI300X GPU using the Adam optimiser and a cosine learning rate scheduler.

5 Results

Results in Figure 2 show the CMC Rank-1 performance and standard deviation of all re-identification models across different session-gaps ($g_{_}$) with the corresponding range of calendar days ($d_{_}$), and increasing numbers of query and gallery images per pig. Darker colours indicate better performance.

Long-term Re-Identification Across all models, the strongest performance occurs when query and gallery images are drawn from the same day, i.e., session-gap 0. With six pigs per pen in the validation and testing sets, the at-chance Rank-1 CMC is $\approx 16\%$. Hence, all models demonstrate performance well above chance at session-gap 0, with the best-performing models exceeding 85% across all query and gallery settings. For all models, performance generally decreases as the session-gap increases, and several are near chance level by around 6 weeks (42 days). These results underscore the challenge of long-term pig re-identification, where factors such as growth, appearance variation, and environmental changes may adversely affect model performance over long time periods. Averaged across all session-gaps, the best-performing models are Pure CE and CE+Triplet. This demonstrates that standard ID loss functions, such as cross-entropy, are effective for long-term pig ReID tasks. In contrast, Pure Triplet loss performs poorly across all settings on this dataset. Apart from the case of 1 query and 1 gallery image per pig, which may, however, be more strongly affected by chance factors. The TAUDL and TAUDL_G methods differ by around 6–7% in average performance, suggesting the value of global constant ID across videos, although maintaining such identity information across independent recording sessions may be challenging in commercial deployments.

Varying Query and Gallery Images Increasing the number of query and gallery images improves the performance of all methods, including over longer periods. We hypothesise that increasing the number of images per pig reduces visual noise by diminishing the influence of chance factors, such as occlusion or illumination, which may degrade individual-specific information in randomly selected images. However, simply increasing the number of images per pig is not enough to enable reliable long-term re-identification alone, as even with 5 query and 5 gallery images per pig, performance drops to near chance for many models by around 42 days.

CMC Rank-1 mean and std by session and calendar days (%)

	1 Query 1 Gallery										3 Query 3 Gallery										5 Query 5 Gallery									
ArcFace mean	85.0	36.7	26.7	27.5	18.3	24.2	15.8	33.5			98.3	41.7	33.3	30.0	24.2	21.7	20.8	38.6			97.5	39.2	33.3	27.5	29.2	30.8	25.0	40.4		
std	6.2	12.5	11.7	12.4	11.7	10.8	6.9	10.3			3.3	15.8	11.2	13.5	11.5	10.0	11.3	11.0			3.8	11.2	11.2	9.9	6.7	11.8	10.5	9.3		
CE+Triplet mean	76.7	41.7	30.0	30.8	23.3	20.0	15.0	33.9			96.7	50.0	45.0	38.3	28.3	25.0	23.3	43.8			95.0	51.7	45.0	33.3	26.7	30.0	19.2	43.0		
std	12.2	11.2	15.9	11.2	9.7	10.0	9.7	11.4			4.1	9.9	13.5	13.0	15.5	11.2	11.7	11.3			5.5	15.3	10.7	7.5	7.3	9.3	8.4	9.1		
Pure CE mean	86.7	45.0	30.8	34.2	27.5	15.8	19.2	37.0			96.7	50.0	37.5	29.2	28.3	31.7	22.5	42.3			96.7	46.7	40.0	33.3	25.8	31.7	26.7	43.0		
std	9.3	11.9	14.9	16.9	15.4	9.5	7.5	12.2			4.1	12.9	11.3	10.7	11.9	10.4	10.6	10.3			5.5	8.5	11.1	5.3	7.9	13.8	12.8	9.3		
Pure Triplet mean	38.3	27.5	22.5	22.5	21.7	20.0	15.8	24.0			64.2	30.0	26.7	22.5	28.3	22.5	22.5	31.0			79.2	33.3	33.3	31.7	22.5	19.2	26.7	35.1		
std	7.6	13.5	10.6	7.5	17.2	10.0	12.0	11.2			11.2	11.3	11.1	9.2	7.6	9.2	8.4	9.7			11.9	7.5	15.4	9.0	11.8	10.6	7.3	10.5		
SupCon mean	54.2	41.7	37.5	26.7	32.5	22.5	21.7	33.8			84.2	45.0	40.0	30.0	29.2	30.0	21.7	40.0			85.8	44.2	49.2	36.7	31.7	30.8	22.5	43.0		
std	13.6	9.9	18.0	13.8	13.7	9.2	9.3	12.5			7.9	10.0	7.3	7.6	11.3	6.7	10.0	8.7			9.9	15.8	10.8	8.5	11.1	9.9	8.4	10.6		
TAUDL mean	66.7	39.2	28.3	26.7	26.7	23.3	18.3	32.7			86.7	42.5	25.0	35.0	28.3	17.5	20.8	36.5			92.5	45.8	33.3	30.8	25.8	15.8	15.8	37.1		
std	11.8	12.9	8.5	8.2	11.7	12.8	10.4	10.9			9.3	9.5	11.2	8.2	7.6	7.9	6.7	8.6			7.9	11.3	6.5	7.5	10.8	6.9	4.5	7.9		
TAUDL_G mean	65.0	30.0	35.8	30.8	32.5	24.2	26.7	35.0			87.5	45.8	38.3	35.8	30.0	27.5	37.5	43.2			94.2	50.0	45.8	38.3	24.2	25.8	27.5	43.7		
std	15.7	14.5	17.5	11.8	9.5	14.6	10.4	13.4			8.5	10.7	12.5	10.6	7.6	9.9	10.7	10.1			7.5	13.9	11.3	10.0	9.5	6.9	7.5	9.5		
	g0	g1	g2	g3	g4	g5	g6	AVG			g0	g1	g2	g3	g4	g5	g6	AVG			g0	g1	g2	g3	g4	g5	g6	AVG		
	0d	6-8d	12-16d	20-22d	28-29d	34-36d	41-43d	AVG			0d	6-8d	12-16d	20-22d	28-29d	34-36d	41-43d	AVG			0d	6-8d	12-16d	20-22d	28-29d	34-36d	41-43d	AVG		
	Session gap / calendar days																													

Figure 2: CMC Rank-1 Re-Identification performance across long time periods, using different numbers of images per pig for query and gallery. Darker colours represent better performance. For each method, the mean and standard deviation (std) over 10 runs are reported. Columns are labelled by session-gap $g_$ and corresponding calendar days $_d$.

6 Conclusion

In this paper, we have, for the first time, compared the long-term performance of different ReID methods in pigs, using a new public dataset. Most methods perform well on the same-day re-identification task, achieving at least 75% CMC Rank-1 except for Pure Triplet loss, demonstrating that human ReID methods are capable of performing short-term pig ReID tasks. At the same time, the performance of all evaluated methods degrades over longer time periods. Many methods approach chance-level performance at a gap of around 42 days between query and gallery images. The use of multiple query and gallery images per pig generally improves the performance of all the re-identification methods tried, but alone, this is not enough to enable practical long-term re-identification. The best-performing long-term re-identification methods make use of standard cross-entropy loss, with Pure CE and CE + Triplet losses achieving the best performance. We hope that our analysis will serve as a starting point for future research on long-term pig re-identification, helping to enable individual-level interventions for improved health and welfare outcomes.

7 Limitations and Future Work

Although we experimented with various ReID methods, we used ResNet backbone to extract embeddings. Future work may extend to a backbone based on the Vision Transformer (ViT) and orientation normalisation using a pig body keypoint estimation model to align images along the head-to-tail axis. Limited by the currently available dataset size, our evaluation set contains only 12 pigs, resulting in relatively large variance and limiting fine-grained statistical comparisons between methods. We expect this to be addressed once the full dataset becomes available.

Acknowledgments

This work is funded by the UKRI AI Centre for Doctoral Training in Sustainable Understandable agri-food Systems Transformed by Artificial Intelligence (SUSTAIN) and Zoetis.

References

- [1] Elanchezhian Arulmozhi et al. “The Application of Cameras in Precision Pig Farming: An Overview for Swine-Keeping Professionals”. en. In: *Animals* 11.8 (Aug. 2021), p. 2343. ISSN: 2076-2615. DOI: 10.3390/ani11082343. URL: <https://www.mdpi.com/2076-2615/11/8/2343>.
- [2] Anicetus Odo, Niall McLaughlin, and Ilias Kyriazakis. “Automated Monitoring of Ear Biting in Pigs by Tracking Individuals and Events”. In: *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Jan. 2024, pp. 7080–7088. DOI: 10.1109/WACV57701.2024.00694. URL: <https://ieeexplore.ieee.org/document/10484109>.
- [3] M. H. Nidhi, K. Liu, and K. J. Flay. “Review: Multiobject tracking in livestock – from farm animal management to state-of-the-art methods”. en-US. In: *animal* 19.5 (May 2025), p. 101503. ISSN: 1751-7311. DOI: 10.1016/j.animal.2025.101503. URL: <https://www.sciencedirect.com/science/article/pii/S1751731125000862>.
- [4] Anicetus Odo, Niall McLaughlin, and Ilias Kyriazakis. “Re-identification for long-term tracking and management of health and welfare challenges in pigs”. In: *Biosystems Engineering* 251 (Mar. 2025), pp. 89–100. ISSN: 1537-5110. DOI: 10.1016/j.biosystemseng.2025.02.001. URL: <https://www.sciencedirect.com/science/article/pii/S153751102500025X>.
- [5] Gytis Bernotas et al. “A Large-Scale Longitudinal Dataset for Pig Tracking and Re-Identification”. In: *Smart Agricultural Technology* (Mar. 2026), p. 101967. ISSN: 2772-3755. DOI: 10.1016/j.atech.2026.101967. URL: <https://www.sciencedirect.com/science/article/pii/S2772375526001917>.
- [6] Jonathan Henrich et al. *Benchmarking pig detection and tracking under diverse and challenging conditions*. arXiv:2507.16639 [cs]. July 2025. DOI: 10.48550/arXiv.2507.16639. URL: <http://arxiv.org/abs/2507.16639>.
- [7] Feng-Kai Huang et al. “OinkTrack: An Ultra-Long-Term Dataset for Multi-Object Tracking and Re-Identification of Group-Housed Pigs”. en. In: *Proceedings of the 33rd ACM International Conference on Multimedia*. Dublin, Ireland: ACM, Oct. 2025, pp. 12556–12563. ISBN: 979-8-4007-2035-2. DOI: 10.1145/3746027.3758189. URL: <https://dl.acm.org/doi/10.1145/3746027.3758189>.
- [8] Saihui Hou et al. *OpenAnimals: Revisiting Person Re-Identification for Animals Towards Better Generalization*. arXiv:2410.00204 [cs]. Sept. 2024. DOI: 10.48550/arXiv.2410.00204. URL: <http://arxiv.org/abs/2410.00204>.
- [9] Phoenix Yu et al. *Automated Re-Identification of Holstein-Friesian Cattle in Dense Crowds*. arXiv:2602.15962 [cs]. Feb. 2026. DOI: 10.48550/arXiv.2602.15962. URL: <http://arxiv.org/abs/2602.15962>.
- [10] Jiankang Deng et al. “ArcFace: Additive Angular Margin Loss for Deep Face Recognition”. In: 2019, pp. 4690–4699. URL: https://openaccess.thecvf.com/content_CVPR_2019/html/Deng_ArcFace_Additive_Angular_Margin_Loss_for_Deep_Face_Recognition_CVPR_2019_paper.html.
- [11] Minxian Li, Xiatian Zhu, and Shaogang Gong. “Unsupervised Person Re-identification by Deep Learning Tracklet Association”. en. In: *Computer Vision – ECCV 2018*. Ed. by Vittorio Ferrari et al. Vol. 11208. Cham: Springer International Publishing, 2018, pp. 772–788. ISBN: 978-3-030-01224-3 978-3-030-01225-0. DOI: 10.1007/978-3-030-01225-0_45. URL: https://link.springer.com/10.1007/978-3-030-01225-0_45.
- [12] Hao Luo et al. *Bag of Tricks and A Strong Baseline for Deep Person Re-identification*. arXiv:1903.07071 [cs]. Apr. 2019. DOI: 10.48550/arXiv.1903.07071. URL: <http://arxiv.org/abs/1903.07071>.

- [13] Alexander Hermans, Lucas Beyer, and Bastian Leibe. *In Defense of the Triplet Loss for Person Re-Identification*. arXiv:1703.07737 [cs]. Nov. 2017. DOI: 10.48550/arXiv.1703.07737. URL: <http://arxiv.org/abs/1703.07737>.
- [14] Prannay Khosla et al. *Supervised Contrastive Learning*. arXiv:2004.11362 [cs]. Mar. 2021. DOI: 10.48550/arXiv.2004.11362. URL: <http://arxiv.org/abs/2004.11362>.

A Appendix: Data subset statistics

Split	Group	# Pigs	Recoding Interval (Days)	Max Interval (Days)	Min Interval (Days)	Recording Times
Train	G3	6	7.14 ± 0.64	8	6	8.00 ± 0.00
Train	G4	6	7.14 ± 0.64	8	6	8.00 ± 0.00
Train	G5	6	7.14 ± 0.64	8	6	8.00 ± 0.00
Train	G6	6	7.14 ± 0.64	8	6	8.00 ± 0.00
Train	G7	7	7.25 ± 0.83	8	6	5.00 ± 0.00
Train	G8	7	7.25 ± 0.83	8	6	5.00 ± 0.00
Train	G9	7	6.83 ± 0.90	8	6	7.00 ± 0.00
Train	G12	5	6.83 ± 0.90	8	6	7.00 ± 0.00
Train	G13	7	3.64 ± 1.15	6	2	12.00 ± 0.00
Train	G17	20	2.44 ± 2.38	7	1	4.50 ± 0.92
Train	G18	19	2.21 ± 2.77	11	1	14.47 ± 2.23
Train	G20	6	1.00 ± 0.00	1	1	18.00 ± 0.00
Validation	G1	6	7.14 ± 0.64	8	6	8.00 ± 0.00
Validation	G10	6	6.83 ± 0.90	8	6	7.00 ± 0.00
Test	G2	6	7.14 ± 0.64	8	6	8.00 ± 0.00
Test	G11	6	6.83 ± 0.90	8	6	7.00 ± 0.00

Table 2: Statistics of the defined PigReID train/validation/test split. The observe interval is computed from adjacent recording dates of the same pig, and sampling times denote the number of recording dates per pig.

B Appendix: Loss Function Hyper-parameters

Loss name	Hyper-parameters
CE+Triplet	$id_loss_weight = 0.3$; $tri_loss_weight = 0.7$; $margin = 0.2$; $LR = 6e-6$; $steps = 16025$
ArcFace	$id_loss_scale = 48$; $id_loss_margin = 0.35$; $LR = 5e-6$; $steps = 25640$
SupCon	$temperature = 0.03$; $LR = 1e-5$; $steps = 6410$
TAUDL	$head = multi_head$; $LR = 6e-5$; $steps = 1000$
TAUDL_G	$head = single_head$; $LR = 6e-5$; $steps = 1000$
Pure Triplet	$id_loss_weight = 0.0$; $tri_loss_weight = 1.0$; $margin = 0.2$; $LR = 6e-6$; $steps = 16025$
Pure CE	$id_loss_weight = 1.0$; $tri_loss_weight = 0.0$; $LR = 6e-6$; $steps = 16025$

Table 3: Loss hyper-parameters. id_loss_weight and tri_loss_weight are the factors for ID loss and triplet loss, respectively, that contribute to the final loss; the two factors sum to 1. LR represents the learning rate.

Model Compression for Very Short-Term Solar Irradiance Forecasting Using Infrared Sky Images

Ifran Rahman Nijhum^{1,2,†}, Arnav Jaitly^{2,†}, David Lillis², and Soumyabrata Dev^{1,2,3}

¹*The ADAPT Research Centre, Dublin, Ireland*

²*School of Computer Science, University College Dublin, Dublin, Ireland*

³*School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland*

[†]*These authors contributed equally to this work.*

Abstract

Accurate short-term solar irradiance forecasting is essential for reliable photovoltaic power integration. While convolutional neural networks (CNNs) have demonstrated strong performance for sky-image-based forecasting, deploying deep learning models on resource-constrained edge devices remains challenging due to their computational and memory requirements. However, the impact of model compression on solar forecasting accuracy and deployment efficiency has not been systematically investigated. In this work, we present a comprehensive benchmark of structured and unstructured pruning for infrared sky image-based solar irradiance forecasting using infrared sky images from the Girasol dataset. We evaluate four widely used CNN architectures ResNet-18, EfficientNet-B0, MobileNetV2, and MobileNetV3-Small along with a lightweight custom baseline model (TinyCNN) across multiple pruning levels ranging from 20% to 80%. Models are assessed using forecasting accuracy, model size, parameter reduction, and inference latency. Experimental results show that MobileNetV2 achieves the strongest baseline forecasting performance, with an RMSE of 37.11 W/m². Structured pruning yields substantial parameter and latency reductions but is architecture-dependent, with some models degrading sharply even at the lowest pruning level tested. Unstructured pruning preserves accuracy more consistently across architectures and substantially reduces compressed checkpoint size, but provides little to no inference speed improvement on conventional hardware. TinyCNN tolerates the highest compression of any architecture under both strategies, though from a much higher baseline error. Overall, the two strategies offer complementary trade-offs structured pruning favoring latency and parameter reduction, unstructured pruning favoring accuracy preservation demonstrating that meaningful compression is achievable for CNN-based solar irradiance forecasting, with implications for edge deployment.

Keywords: Solar irradiance forecasting, infrared sky imaging, convolutional neural networks, model pruning, edge deployment

1 Introduction

The increasing adoption of photovoltaic energy systems has made accurate short-term solar irradiance forecasting essential for grid stability, energy management, and power dispatch [Barhmi et al., 2026, Alnajadah et al., 2025]. Since rapid cloud movement can cause large intra-hour fluctuations in solar power generation, forecasting Global Solar Irradiance (GSI) remains an important research problem [Nie et al., 2024]. Ground-based sky imaging systems are particularly effective for short-term irradiance forecasting because they directly capture local cloud dynamics at high temporal resolution [Diagne et al., 2013, Paletta et al., 2023]. The Girasol dataset introduced by Terrén-Serrano et al. [Terrén-Serrano et al., 2021] provides synchronized infrared sky images and pyranometer measurements for irradiance estimation. Infrared imaging is useful in this context because it captures cloud structure while being less affected by strong solar glare than visible-light imaging.

Recent deep learning approaches have improved image-based irradiance forecasting by learning cloud-related features directly from sky images [Klotz and Nayar, 2026]. CNN-based models, including lightweight architectures such as MobileNetV2, have shown promising performance on the Girasol dataset [Nijhum et al., 2024]. Complementary research has also leveraged RGB and near-infrared (NIR) sky images for solar forecasting, highlighting the value of multi-spectral imaging for capturing cloud dynamics and irradiance variability [Nijhum et al., 2026]. However, practical deployment remains challenging. Solar monitoring stations and embedded IoT devices often operate under limited memory, computation, and energy budgets, while many CNN-based forecasting models are still too costly for efficient edge inference [Menghani, 2023, Paletta et al., 2023]. This creates a trade-off between forecasting accuracy and deployment efficiency. Model compression provides a practical way to reduce this cost. Pruning can remove redundant parameters or channels from neural networks, while quantization reduces numerical precision to lower storage and computational requirements [He and Xiao, 2023, Fang et al., 2023, Wu et al., 2020, Guo, 2018]. Although these techniques are widely studied in general computer vision, their use for sky-image-based solar irradiance forecasting remains limited.

To the best of our knowledge, no prior work has systematically evaluated pruning techniques across multiple CNN architectures for infrared sky image-based solar irradiance forecasting. Existing studies mainly focus on forecasting accuracy, with limited attention to deployment-oriented metrics such as model size and inference latency. In this work¹, we present a systematic compression benchmark for CNN-based solar irradiance forecasting using infrared sky imagery from the Girasol dataset. Five architectures, ResNet18, MobileNetV2, MobileNetV3-Small, EfficientNet-B0, and a custom baseline (Tiny CNN), are evaluated under multiple structured and unstructured pruning levels. The models are compared using validation RMSE, model size, and inference latency.

The main contributions of this work are summarized as follows:

- We evaluate the performance of several popular lightweight CNN architectures for short-term solar irradiance forecasting using infrared sky imagery.
- We perform both structured and unstructured pruning across multiple compression levels to analyze their impact on forecasting performance.
- We provide a comparative evaluation of forecasting accuracy, model size, and inference latency to study the trade-off between compression and deployment efficiency.

2 Methodology

This study investigates the impact of structured and unstructured pruning on convolutional neural networks (CNNs) for infrared sky image-based solar irradiance forecasting. The overall pipeline consists of four stages: dataset preparation, baseline model training, model compression through pruning, and post-pruning fine-tuning and evaluation. Infrared sky images are used as model inputs, while future global horizontal irradiance (GHI) values measured by a pyranometer serve as prediction targets. Multiple lightweight CNN architectures are evaluated, including MobileNetV2, MobileNetV3-small, EfficientNet-B0, ResNet-18, and a custom Tiny CNN baseline. Baseline models are first trained using a unified training configuration. Structured and unstructured pruning are then independently applied at multiple pruning ratios, followed by fine-tuning and evaluation.

2.1 Problem Formulation

The forecasting task is formulated as a supervised image-to-regression problem. Given an infrared sky image captured at time step t , the objective is to predict the future solar irradiance value at time step $t + L$, where L denotes the forecasting horizon.

$$\hat{y}_{t+L} = f(I_t; \theta), \quad (1)$$

¹<https://github.com/ifran-rahman/solar-forecasting-pruning>

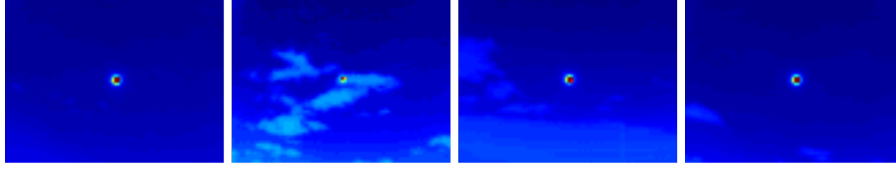


Figure 1: Sample infrared sky images from the Girasol dataset used for solar irradiance forecasting. The images capture a range of sky conditions, including clear, partially cloudy, and overcast scenarios.

where I_t denotes the infrared sky image at time t , $\hat{y}_{t+L}$ denotes the predicted future irradiance value, $f(\cdot)$ represents the CNN forecasting model, and θ denotes the trainable model parameters. The forecasting horizon was fixed at 5 minutes (20 future time steps).

2.2 Dataset Preparation

The Girasol Sky Imaging and Global Solar Irradiance Dataset [Terrén-Serrano et al., 2021] contains synchronized infrared sky images and pyranometer-based irradiance measurements collected at the University of New Mexico for short-term solar forecasting. Figure 1 demonstrate some sample IR images from the dataset. This study uses 59,778 infrared sky images acquired between December 2017 and January 2018. To preserve temporal consistency and prevent data leakage, the dataset was split chronologically, with training samples drawn from 8 December 2017 to 8 January 2018 and validation samples taken from the subsequent 7-day period, consisting of 48,164 raw training images and 11,614 raw validation images. The forecasting horizon was fixed at $L = 20$ future time steps. After removing samples without valid future targets, the final dataset contained 47,564 training samples and 11,474 validation samples. For each sample, the input image at time t was paired with the irradiance measurement at time $t + L$. Images with a resolution of 60×80 pixels were normalized using min-max normalization.

2.3 CNN Architectures

Five CNN architectures were evaluated to analyze the effectiveness of pruning across lightweight and heavy-weight models. These widely adopted architectures are commonly used benchmarks for studying accuracy-efficiency trade-offs. All architectures were adapted for scalar irradiance regression by replacing their original classification heads with a single fully connected regression layer producing one irradiance prediction.

MobileNetV2 is a lightweight CNN architecture based on inverted residual blocks and depthwise separable convolutions, designed for efficient mobile and edge inference. MobileNetV3 Small extends this design by incorporating squeeze-and-excitation modules and hardware-aware neural architecture search for improved computational efficiency. EfficientNet-B0 employs compound scaling of network depth, width, and input resolution to achieve a balanced trade-off between accuracy and computational cost. ResNet18 is a residual CNN architecture that uses skip connections to facilitate stable optimization of deeper networks and serves as the heavyweight reference model in this study. In addition, a custom lightweight architecture, referred to as TinyCNN, was implemented using two convolutional layers with ReLU activations and max-pooling operations, followed by a fully connected regression head. The TinyCNN architecture consists of two convolutional blocks followed by a fully connected regression layer, as summarized in Eq. (2). TinyCNN was designed as an extremely lightweight edge-deployment baseline.

$$\text{Input} \rightarrow \text{Conv}(3 \times 3, 16) \rightarrow \text{ReLU} \rightarrow \text{MaxPool} \rightarrow \text{Conv}(3 \times 3, 32) \rightarrow \text{ReLU} \rightarrow \text{MaxPool} \rightarrow \text{Flatten} \rightarrow \text{FC}(9600 \rightarrow 1) \quad (2)$$

All models were trained from scratch without pretrained weights. For MobileNetV2, MobileNetV3-Small, EfficientNet-B0, and ResNet18, the original classification layers were removed and replaced with regression heads. Adaptive average pooling was additionally applied after feature extraction for the MobileNet-based architectures.

2.4 Structured Pruning

Structured pruning removes entire convolutional filters or channels, reducing parameter count and computational complexity while preserving dense tensor operations compatible with standard hardware [Zhu et al., 2025]. Magnitude-based structured pruning was implemented using the `torch-pruning` framework with L2-norm filter importance:

$$S_i = \|W_i\|_2 = \sqrt{\sum_j w_{ij}^2}, \quad (3)$$

where W_i denotes the weights of filter i . Filters with lower importance scores were pruned first.

Pruning ratios of 20%, 40%, 60%, and 80% were evaluated across all CNN architectures. Following pruning, models were fine-tuned for 3 epochs using Adam with a reduced learning rate of 10^{-4} . Performance was evaluated using RMSE, MAE, parameter reduction, model size, and inference latency.

$$\text{Parameter Reduction (\%)} = \left(1 - \frac{P_{\text{pruned}}}{P_{\text{baseline}}}\right) \times 100, \quad (4)$$

where P_{baseline} and P_{pruned} denote the number of parameters before and after pruning, respectively.

2.5 Unstructured Pruning

Unstructured pruning removes individual weights while preserving the original tensor dimensions, producing sparse weight matrices. L1 unstructured pruning was applied to convolutional and linear layers using the PyTorch pruning utilities. Weight importance was determined using absolute magnitude:

$$S_{ij} = |w_{ij}|, \quad (5)$$

where w_{ij} denotes an individual weight parameter.

Pruning ratios of 20%, 40%, 60%, and 80% were evaluated. After pruning, models were fine-tuned for 3 epochs using Adam with a learning rate of 10^{-4} . Final evaluation included RMSE, MAE, sparsity percentage, model size, and inference latency.

2.6 Training Configuration

All models were trained using the Adam optimizer with a batch size of 256 and a mean squared error (MSE) loss function. To ensure reproducibility, a fixed random seed of 42 was used for NumPy, Python, and PyTorch, with deterministic CUDA operations enabled. The baseline models were trained from random initialization for 50 epochs using an initial learning rate of 0.001. A StepLR scheduler was employed to reduce the learning rate by a factor of 0.5 every 10 epochs, and the checkpoint corresponding to the lowest validation loss was retained for evaluation. For both structured and unstructured pruning experiments, the trained baseline models served as the starting point. Pruning was applied at four pruning levels (20%, 40%, 60%, and 80%), followed by three epochs of fine-tuning using a reduced learning rate of (1×10^{-4}) to recover performance. Structured pruning was implemented using the Torch-Pruning MagnitudePruner with L2-norm importance, while unstructured pruning employed L1-based weight pruning on convolutional and linear layers. In all cases, the best fine-tuned model was selected based on validation loss and used for the final evaluation.

3 Experiments and Results

3.1 Evaluation Metrics

All models were optimized using mean squared error (MSE) loss and evaluated using root mean squared error (RMSE) and mean absolute error (MAE), reported in W/m²:

Table 1: Baseline forecasting performance, parameter counts, and checkpoint sizes for 60×80 infrared sky images. Lower values indicate better forecasting performance.

Model	Params (M)	RMSE (W/m ²)	MAE (W/m ²)	MSE (W ² /m ⁴)	Size (MB)	Latency (ms/img)
TinyCNN	0.015	82.14	61.87	6747.77	0.06	0.0200
MobileNetV2	2.225	37.11	27.14	1376.78	8.72	0.0997
MobileNetV3	0.928	39.72	28.22	1577.53	3.67	0.0475
EfficientNet-B0	4.009	38.97	25.20	1518.69	15.58	0.1180
ResNet-18	11.177	45.49	33.76	2068.93	42.71	0.0791

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2, \quad (6)$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}, \quad (7)$$

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|. \quad (8)$$

Here y_i and $\hat{y}_i$ denote the ground-truth and predicted irradiance values, respectively. In addition to forecasting accuracy, deployment-oriented metrics including parameter reduction, sparsity percentage, checkpoint size, compressed checkpoint size, and inference latency per image are also measured to evaluate model efficiency after compression.

3.2 Baseline Performance

The forecasting performance of the evaluated CNN architectures is presented in Table 1. All models were trained using infrared sky images with a spatial resolution of 60×80 pixels. MobileNetV2 achieved the lowest RMSE of 37.11 and the lowest MSE of 1376.78, indicating the strongest overall forecasting performance among the evaluated models. EfficientNet-B0 obtained the lowest MAE of 25.20, while its RMSE remained close to MobileNetV2. MobileNetV3 also achieved competitive performance, with an RMSE of 39.72 and a smaller checkpoint size of 3.67 MB. ResNet-18, despite having the largest checkpoint size of 42.71 MB, did not provide the best forecasting accuracy. On the contrary, EfficientNet-B0 is smaller than ResNet-18 (15.58 MB vs. 42.71 MB) yet slower (0.1180 ms/img vs. 0.0791 ms/img), while MobileNetV3 is the fastest of the non-TinyCNN models at 0.0475 ms/img. TinyCNN required only 0.06 MB of storage and 0.0200 ms per image, making it the smallest and fastest model, but its prediction error was noticeably higher than the other architectures. Overall, these results show that larger model size does not necessarily lead to better forecasting performance for low-resolution infrared sky images. MobileNetV2 provides the best accuracy, while MobileNetV3 offers a strong balance between accuracy, storage, and latency efficiency.

3.3 Structured Pruning

Structured pruning was applied to all baseline models using the torch-pruning framework at pruning levels of 20%, 40%, 60%, and 80%. Following pruning, each model was fine-tuned for three epochs and evaluated using RMSE, parameter reduction, model size, and inference latency. For comparison, the most compressed configuration below the 10% RMSE-increase threshold is reported; when no configuration met the threshold, the one with the smallest RMSE increase is shown instead. Table 2 shows that structured pruning removes a substantial share of parameters while largely preserving accuracy, with parameter counts falling by roughly a third to a half and RMSE increasing between 2.0% (ResNet-18) and 22.2% (MobileNetV3); MobileNetV2 and MobileNetV3 exceed the 10% threshold even at the lowest pruning level tested. TinyCNN attains the largest

Table 2: Selected structured-pruning configurations. For architectures with no configuration satisfying the 10% relative RMSE-increase threshold, the least-degrading tested configuration is reported.

Model	Prune (%)	Param. Red. (%)	Base RMSE (W/m ²)	Final RMSE (W/m ²)	Size (MB)	Latency (ms/img)
EfficientNet	20	35.37	38.97	40.97	10.15	0.1010
MobileNetV2	20	35.46	37.10	42.62	5.69	0.0772
MobileNetV3	20	35.19	39.72	48.52	2.42	0.0818
TinyCNN	40	48.87	82.14	86.12	0.03	0.0274
ResNet-18	20	36.26	45.48	46.40	27.25	0.1003

parameter reduction (48.87%) at a modest accuracy cost. Among the full-size architectures, EfficientNet is most accurate but slowest, ResNet-18 is most robust in RMSE despite being the largest model and among the slowest, and MobileNetV3 yields the smallest model at the largest RMSE increase, making it size-aggressive rather than best-balanced.

Figure 2a shows that 80% pruning degrades all architectures sharply, pushing validation RMSE well above 130 (and above 290 for EfficientNet, MobileNetV2/V3, and ResNet-18), indicating this ratio exceeds the viable compression limit for the task. TinyCNN dominates latency, running roughly 3-10× faster than the other architectures, but its best configuration (20% pruning, RMSE ≈82) cannot match the accuracy of larger models at low pruning ratios, reflecting a capacity-versus-speed trade-off. At 20% pruning, MobileNetV2 and ResNet-18 form the accuracy-efficiency Pareto frontier: MobileNetV2 attains the lowest RMSE (≈45) at ≈0.075 ms, while ResNet-18 reaches comparable RMSE (≈50) at ≈0.10 ms but with a substantially larger parameter footprint, making MobileNetV2 the more parameter-efficient choice for high-accuracy deployment.

3.4 Unstructured Pruning

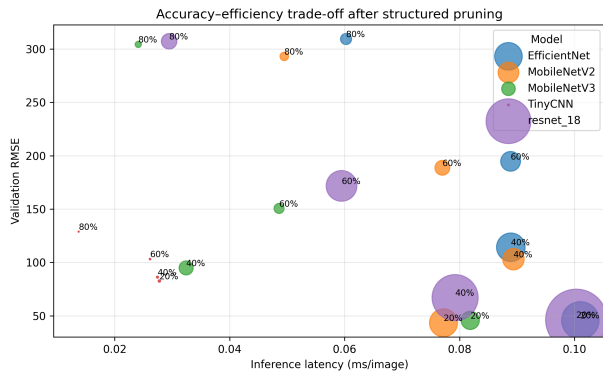
To evaluate the compressibility of the forecasting models, unstructured pruning was applied using L1-norm weight pruning on all convolutional and fully connected layers. Each model was pruned at 20%, 40%, 60%, and 80% sparsity levels and subsequently fine-tuned for 3 epochs. Since unstructured pruning preserves the original network architecture and does not reduce the number of stored parameters, compression effectiveness was evaluated using the size of gzip-compressed model checkpoints, in contrast to structured pruning, which reports the actual checkpoint size since parameters are physically removed. For each architecture, the best configuration was selected as the most compressed model that maintained an RMSE increase below 10% relative to the baseline model. Where no configuration met this threshold, the configuration with the smallest RMSE increase is reported instead.

Table 3 summarizes the selected configurations. EfficientNet, MobileNetV2, and ResNet-18 met the 10% RMSE threshold at approximately 20% sparsity, with RMSE increases of 3.6%, 3.0%, and 7.8%, respectively. TinyCNN tolerated the highest sparsity (60%) with only a 1.7% RMSE increase, but from a much higher baseline (82.14 W/m²). MobileNetV3 was the exception, showing a 31.7% RMSE increase (39.72 to 52.30 W/m²) at comparable sparsity, indicating higher sensitivity to unstructured weight removal. Excluding TinyCNN, MobileNetV2 anchors the accurate end with the lowest RMSE (38.23 W/m²). ResNet-18 offers the lowest latency among the full-size models (0.0887 ms/img) but the highest RMSE among them (49.02 W/m²), while MobileNetV3 combines the smallest footprint (2.90 MB) with the highest RMSE (52.30 W/m²) among the accurate models. Overall, unstructured pruning substantially reduces storage while largely preserving performance, but benefits are architecture-dependent and do not reliably translate into latency gains on dense hardware.

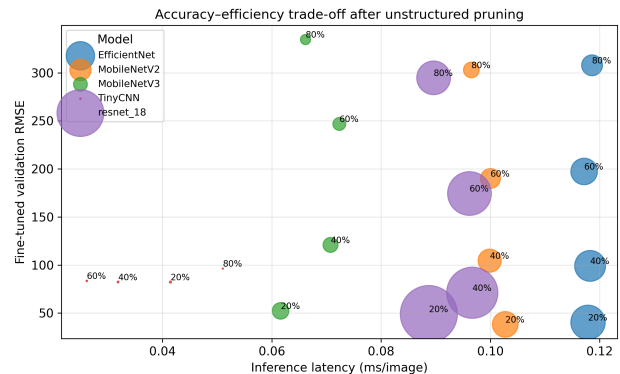
Figure 2b illustrates the resulting accuracy-efficiency trade-off. Across EfficientNet, MobileNetV2, and ResNet-18, latency remains essentially flat from 20% to 80% sparsity, confirming that unstructured sparsity yields no measurable hardware speedup. RMSE, in contrast, rises monotonically with pruning ratio across all architectures, reaching a 4-6× increase over the 20% baseline by 80% sparsity. MobileNetV3 shows the sharpest trade-off: lowest latency at 20% pruning but the steepest accuracy collapse at 80%. ResNet-18 matches EfficientNet’s accuracy at every pruning level while sustaining roughly 25% lower latency, indicating a more favorable accuracy-latency profile despite its larger parameter count.

Table 3: Summary of unstructured pruning configurations selected using a 10% relative RMSE-increase comparison threshold. When no configuration met the threshold, the configuration with the smallest RMSE increase is reported.

Model	Prune (%)	Sparsity (%)	Base RMSE (W/m ²)	Final RMSE (W/m ²)	Size (MB)	Latency (ms/img)
EfficientNet	20	19.74	38.97	40.37	12.41	0.1178
MobileNetV2	20	19.69	37.10	38.23	6.94	0.1027
MobileNetV3	20	19.68	39.72	52.30	2.90	0.0616
TinyCNN	60	59.80	82.14	83.50	0.03	0.0261
ResNet-18	20	19.98	45.48	49.02	34.18	0.0887



(a) Accuracy–efficiency trade-off after structured pruning. Bubble size is proportional to model size.



(b) Accuracy–efficiency trade-off after unstructured pruning. Bubble size is proportional to the compressed checkpoint size.

Figure 2: Accuracy–efficiency trade-offs obtained using structured and unstructured pruning. Lower RMSE and lower inference latency indicate better forecasting efficiency, while bubble size represents the storage footprint of each pruned model.

4 Discussion and Conclusion

These results show that effective solar irradiance forecasting from low-resolution infrared sky images can be achieved using relatively lightweight CNN architectures: MobileNetV2 achieved the lowest baseline error despite being smaller than ResNet-18, and this relationship inverts under compression, where ResNet-18 tolerates structured pruning best while MobileNetV2 tolerates unstructured pruning best among the full-size architectures. MobileNetV3 is the consistent exception, degrading sharply under both strategies despite its compact baseline footprint. Under structured pruning at low pruning levels, MobileNetV2 reaches lower error than ResNet-18 at lower latency, despite ResNet-18’s greater robustness at higher pruning levels. Under unstructured pruning, latency stayed flat across all sparsity levels for every architecture even as error rose steadily, showing the compression gains were confined to storage rather than inference speed. Both strategies collapsed at the highest pruning ratio tested, with error degrading sharply across most architectures. Future work will explore knowledge distillation and mixed pruning schedules alongside quantization to test whether this compression ceiling can be extended.

Acknowledgments

This research was conducted with the financial support of Research Ireland under Grant Agreement No. 13/RC/2106_P2 at the ADAPT Centre at Trinity College Dublin. ADAPT, the Research Ireland Centre for AI-Driven Digital Content Technology, is funded through the Research Ireland Centres Programme.

References

- [Alnajadah et al., 2025] Alnajadah, A., Bremer, C., Butler, S., Li, F., and TamizhMani, G. (2025). Intra-hour solar irradiance forecasting using sky cameras for pv plants. *Sustainable Energy Technologies and Assessments*, 84:104743.
- [Barhmi et al., 2026] Barhmi, K., Golroodbari, S. M., Knap, W., and Van Sark, W. (2026). Real-time solar irradiance forecasting for grid integration using all-sky imagery and multi-stage ai with kalman filter optimization. *Renewable Energy*, 259:125117.
- [Diagne et al., 2013] Diagne, M., David, M., Lauret, P., Boland, J., and Schmutz, N. (2013). Review of solar irradiance forecasting methods and a proposition for small-scale insular grids. *Renewable and Sustainable Energy Reviews*, 27:65–76.
- [Fang et al., 2023] Fang, G., Ma, X., Song, M., Mi, M. B., and Wang, X. (2023). Depgraph: Towards any structural pruning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16091–16101.
- [Guo, 2018] Guo, Y. (2018). A survey on methods and theories of quantized neural networks. *arXiv preprint arXiv:1808.04752*.
- [He and Xiao, 2023] He, Y. and Xiao, L. (2023). Structured pruning for deep convolutional neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 46(5):2900–2919.
- [Klotz and Nayar, 2026] Klotz, J. and Nayar, S. K. (2026). Forecasting solar energy using a single image. *Solar Energy*, 313:114540.
- [Menghani, 2023] Menghani, G. (2023). Efficient deep learning: A survey on making deep learning models smaller, faster, and better. *ACM Computing Surveys*, 55(12):1–37.
- [Nie et al., 2024] Nie, Y., Li, X., Paletta, Q., Aragon, M., Scott, A., and Brandt, A. (2024). Open-source sky image datasets for solar forecasting with deep learning: A comprehensive survey. *Renewable and Sustainable Energy Reviews*, 189:113977.
- [Nijhum et al., 2024] Nijhum, I. R., Kaloni, D., Kenny, P., and Dev, S. (2024). Enhancing intra-hour solar irradiance estimation through knowledge distillation and infrared sky images. In *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*, pages 7207–7211.
- [Nijhum et al., 2026] Nijhum, I. R., Scanlon, C., and Dev, S. (2026). Design and evaluation of a dual-spectrum low-cost sky imager for enhanced solar prediction. *Measurement: Energy*, page 100103.
- [Paletta et al., 2023] Paletta, Q., Terrén-Serrano, G., Nie, Y., Li, B., Bieker, J., Zhang, W., Dubus, L., Dev, S., and Feng, C. (2023). Advances in solar forecasting: Computer vision with deep learning. *Advances in Applied Energy*, 11:100150.
- [Terrén-Serrano et al., 2021] Terrén-Serrano, G., Bashir, A., Estrada, T., and Martínez-Ramón, M. (2021). Girasol, a sky imaging and global solar irradiance dataset. *Data in Brief*, 35:106914.
- [Wu et al., 2020] Wu, H., Judd, P., Zhang, X., Isaev, M., and Micikevicius, P. (2020). Integer quantization for deep learning inference: Principles and empirical evaluation. *arXiv preprint arXiv:2004.09602*.
- [Zhu et al., 2025] Zhu, K., Hu, F., Ding, Y., Zhou, W., and Wang, R. (2025). A comprehensive review of network pruning based on pruning granularity and pruning time perspectives. *Neurocomputing*, 626:129382.

Comparative Study of Parameter-Efficient Fine-Tuning on Low-Resource Vision-Language Modelling: Burmese-CLIP

Htike Thu Kaung¹ and Salaheddin Alakkari²

¹*Department of Electronic and Computer Engineering, University of Limerick*

²*Department of Computer Science and Information Systems, University of Limerick*

Abstract

Contrastive Language-Image Pre-Training (CLIP) is a powerful vision-language architecture, but fine-tuning it requires substantial computational and data resources. We present the Burmese-CLIP, trained on 20,000 machine-translated image-caption pairs using Low Rank Adaptation (LoRA). Our Burmese-CLIP achieves approximately a seven-fold improvement in Mean Reciprocal Rank (MRR) for text-to-image retrieval over zero-shot NLLB-CLIP. We further conduct a systematic comparison of three PEFT methods, LoRA, DoRA, and AdaLoRA, under identical training conditions to understand their behaviours in a low-resource cross-modal fine-tuning environment. LoRA at rank 32, the model achieves the strongest retrieval quality (MRR@1 of 0.2214), while DoRA at rank 16 retains approximately 86% of this performance with nearly half the trainable parameters, suggesting a favourable efficiency trade-off. AdaLoRA under-performs across all metrics, which we hypothesise that it might be due to the aggressive pruning of attention projection layers, which are critical for cross-modal semantic alignment.

Keywords: Low-Resource Language, Burmese-CLIP, Parameter-Efficient Fine-Tuning(PEFT), Contrastive Learning

1 Introduction

Contrastive Language-Image Pre-Training (CLIP) is a powerful architecture that uses text and visual transformers to encode information into a shared representation space using contrastive learning (Radford et al., 2021; Dosovitskiy et al., 2020; Vaswani et al., 2017). Trained on roughly 400 million image-text pairs from the web (Radford et al., 2021), CLIP achieves outstanding results on various downstream tasks such as zero-shot classification (Thengane et al., 2022; Li et al., 2022), text-image and image-text extraction and more complex tasks such as semantic segmentation (Lin et al., 2023).

CLIP has been adapted to other languages and domains. The Italian CLIP (Bianchi et al., 2021) was adapted by using 1.4 million Italian image-text pairs, while the Chinese CLIP (Yang et al., 2023) scales the approach to 200 million native pairs. Multilingual CLIP (mCLIP) (Zhang et al., 2023b) pursues a broader strategy, mapping dozens of languages into a single shared visual space, and AltCLIP (Chen et al., 2022) uses a teacher-student framework over 2 million image-caption pairs to support cross-lingual retrieval. For low-resource languages, NLLB-CLIP (Visheratin, 2023b) has been developed by replacing the CLIP text encoder with ‘No Language Left Behind’ (NLLB) text encoder achieving excellent results in low-resource languages. Domain-specific variants follow a similar pattern - FashionCLIP (Cartella et al., 2023) fine-tuned CLIP for the retail sector using around 700,000 curated image-text pairs.

In this paper, we explore how different Parameter Efficient Fine Tuning (PEFT) methods perform in fine-tuning CLIP in the low-resource language environment using a small amount of machine-translated data. Therefore, for this experiment, we chose Burmese as our target language. It is due to its unique computational and linguistic challenges such as lack of robust support in ASCII-based systems online for Burmese scripts (Htet and

Dras, 2025). We introduce Burmese-CLIP which is trained on 20,000 machine-translated image-caption pairs from LAION-COCO-NLLB dataset using various Low-Rank Adaptation methodologies (Vishneratin, 2023a). We replaced the text encoder with multilingual XLM-Roberta-Base (XLM-R) (Conneau et al., 2020), which was chosen after a token fertility analysis. Our results indicate that Burmese-CLIP performs better than zero-shot NLLB-CLIP in terms of Mean Reciprocal Rank (MRR) for text-to-image retrieval task, while updating only 4.3% of trainable parameters.

We also investigate the performance of different LoRA methods in developing Burmese-CLIP. We utilised Decomposed Low-Rank Adaptation (DoRA) (Liu et al., 2024) with two different ranks and Adaptive Low-Rank Adaptation (AdaLoRA) (Zhang et al., 2023a), alongside LoRA with 32 rank. The evaluation result reveals LoRA provides the highest with MRR@1 of 0.2214, while DoRA ($r=16$) retains 86% of the retrieval accuracy while training nearly half of the trainable parameters of LoRA. Meanwhile, AdaLoRA with dynamic target rank of 16 under-performs across the board. Our analysis reveals the importance-scoring mechanism prematurely eliminated key attention layers, confirming the AdaLoRA’s bias toward FFN layers, which works against CLIP due to its reliance upon query and key projections (Zhang et al., 2023a; Chung et al., 2025; Wang et al., 2024). To our knowledge, this is the first systematic PEFT study for Burmese-specific CLIP adaptation.

2 Related Work

2.1 Low-Rank Adaptation (LoRA)

Fine-tuning Large Language Models is computationally expensive and data-intensive (Han et al., 2024; Szep et al., 2025; Hu et al., 2022). Low-Rank Adaptation (LoRA) addresses this by freezing pre-trained weights and injecting trainable low-rank matrices as weight updates (Hu et al., 2022; Han et al., 2024; Ding et al., 2023a,b):

$$\Delta W = BA, \quad W_{\text{new}} = W_0 + BA, \quad (1)$$

with $A \in \mathbb{R}^{d \times r}$, $B \in \mathbb{R}^{r \times d}$, and $r \ll d$, reducing trainable parameters to 0.1–3% of the original model while preserving performance relative to full fine-tuning (Han et al., 2024; Hu et al., 2022).

Several LoRA variants have refined this further (Han et al., 2024). DoRA decomposes pre-trained weights into magnitude and direction components, optimising each separately (Liu et al., 2024). AdaLoRA treats low-rank matrices as an SVD and dynamically allocates the parameter budget via an importance-scoring metric (Zhang et al., 2023a), concentrating learning on the layers most critical to the downstream task.

2.2 Low-Resource Language - Burmese Language

Although Myanmar (Burma) has a population of roughly 54-55 million (World Bank, 2023; United Nations Department of Economic and Social Affairs, 2024), Burmese has a limited machine-readable digital footprint (Htet and Dras, 2025), shaped by political, socio-economic, and infrastructural constraints that leave text resources sparse, fragmented, and domain-narrow (Htet and Dras, 2025). This is compounded by encoding fragmentation. The Burmese script lacked robust ASCII support (Htet and Dras, 2025), and although Unicode is now the standard, much online Burmese texts remain in the legacy Zawgyi font system, introducing substantial corpus noise (Oo et al., 2020; Htet and Dras, 2025; Liao, 2017).

Burmese orthography poses further challenges. It does not mark word boundaries with whitespace, and meaning derives from syllable combinations, so naive segmentation can alter semantic meaning (Thet et al., 2008; Maung and Mikami, 2008; Ding et al., 2019). Thus, the chosen tokeniser must minimise out-of-vocabulary fragmentation to avoid such semantic hallucinations (Ding et al., 2019; Thet et al., 2008; Htet and Dras, 2025).

Recent Burmese NLP works include MyanmarGPT (Si Thu, 2024), trained on a curated multi-domain corpus with syllable-aware tokenisation, and BURMESE-SAN (Aung et al., 2026), a benchmark for evaluating LLMs on Burmese-specific tasks. SEA-LION (Ng et al., 2025) similarly argues for targeted regional pre-training on underrepresented Southeast Asian languages. Our work extends this landscape from text-only modelling toward multi-modal.

3 Pre-trained Multilingual Model Selection

Due to unique challenges of the Burmese language, we conducted a fertility analysis to select the most suitable pre-trained multilingual model for Burmese-CLIP. We defined fertility as the ratio of produced tokens to input words for a given text sequence (Rust et al., 2021; Nayeem et al., 2025). Lower fertility indicates that the tokeniser encodes more information per token, where high fertility corresponds to fragmentation into byte-level or encoding-like artefacts that carry minimal semantic content (Nayeem et al., 2025). A total of 47 Burmese sentences were used for this analysis. These sentences can be categorised into machine-translated sentences, similar to the training data, simple sentences such as a common greeting, weather description, and more complex sentences curated by the authors from Burmese news websites such as BBC Burmese, VOA Burmese with topics ranging from politics, economics, education and sport. Results are summarised in Table 1, and XLM-Roberta-Base and mT5-Base outperform the rest. They both produce a clean and coherent tokenisation of Burmese language due to inclusion of Burmese languages as well as employment of SentencePiece tokeniser (Conneau et al., 2020; Xue et al., 2021). XLM-Roberta-Base was selected as the text encoder due to its architecture which aligns with the CLIP model (Radford et al., 2021) and wide applications in many other clips models (Zhang et al., 2023b; Carlsson et al., 2022).

Model Name	Average Fertility Score
Mistral 7B v0.1	1.546
BLOOMZ-560M	1.682
ByT5-small	2.875
mGPT	0.730
mBERT	0.616
mT5-base	0.328
XLM-Roberta-Base	0.325

Table 1: Fertility Analysis of Different Multilingual Models

4 Methodology

4.1 Training & Evaluation Dataset

The required dataset was sourced from `laion-coco-nllb` dataset (Visheratin, 2023a). English captions have been machine-translated into over 200 languages using the NLLB-200 model (Visheratin, 2023a). We filtered the dataset for entries carrying Burmese captions with the aesthetic score threshold ($\text{score} \geq 5.5$) to retain visually informative images to select images with stronger compositional and visual quality. The final dataset comprises approximately 20,000 image-caption pairs. 2,000 image-caption pairs were held out for evaluation. Manual inspection of a random sample of 20 pairs found 18 correctly aligned Burmese captions. The dataset uses Burmese Unicode and therefore, it was not needed to use Unicode converter. No other normalisation practices were performed on the dataset. It is noted that due to a machine translation, some semantic drift and translation artefacts are noted.

4.2 Training

Burmese-CLIP was trained using `VisionTextDualEncoderModel`, which supports independent visual and textual encoders that produce embeddings into a shared projection space. The visual encoder is OpenAI’s CLIP ViT-B/32 that produces a 512-dimensional image embedding, and is frozen throughout all training phases. This is because the visual representation transfers well across different languages (Bianchi et al., 2021; Visheratin, 2023b). The text encoder is XLM-Roberta-Base with 12-layer bidirectional transformer producing a 768-dimensional [CLS] representation projected into the shared 512-dimensional embedding space via a learned linear projection head.

The model is trained using the symmetric contrastive loss from the original CLIP formulation. For a batch of N image-caption pairs, the losses are cross-entropy losses over the rows and columns of the $N \times N$ similarity matrix respectively, with ground-truth labels on the diagonal (Radford et al., 2021; Vishnath, 2023b; Bianchi et al., 2021).

Training follows a two-phase training motivated by prior work of Italian-CLIP adaptation. In phase 1 (3 epochs), both encoders are frozen and only the projection layers are updated. In Phase 2 (7 epochs), LoRA adaptors are applied to only the text encoder. The rationale is that the premature fine-tuning of the text encoder risks destabilising the shared embedding space before a coherent alignment has been established (Bianchi et al., 2021). We chose the batch size of 64 with maximum token length of 64, and use AdamW Optimiser (Bianchi et al., 2021), with cosine annealing learning rate. The best checkpoint was selected by minimum evaluation loss. All shared parameters are in the table below.

Hyperparameter	LoRA $r = 32$	DoRA $r = 32$	DoRA $r = 16$	AdaLoRA
Rank r	32	32	16	init=32, target=16
α	64	64	32	32
t_{init}	—	—	—	$0.10 \cdot T$
t_{final}	—	—	—	$0.50 \cdot T$
ΔT	—	—	—	10
β_1, β_2	—	—	—	0.85, 0.85
Orth. reg. weight	—	—	—	0.5
Trainable params	5.30M	5.44M	2.76M	~2.65M

Table 2: Method-specific hyperparameters.

5 Results

5.1 Retrieval Performance and Baseline Comparison

Methods	Tuning Strategy	Trainable Params	MRR@1	MRR@5	MRR@10
NLLB-CLIP Base	Zero-Shot	0.00M	0.0218	0.0413	0.0462
NLLB-CLIP Large	Zero-Shot	0.00M	0.0248	0.0418	0.0480
AdaLoRA	target $r = 16$	2.65M	0.1337	0.2192	0.2374
DoRA	$r = 16$	2.76M	0.1912	0.2913	0.3093
DoRA	$r = 32$	5.44M	0.2115	0.3136	0.3332
LoRA	$r = 32$	5.30M	0.2214	0.3247	0.3441

Table 3: Text-to-Image retrieval performance across zero-shot baselines, and LoRA variants on the Burmese evaluation dataset.

Table 3 reports Mean Reciprocal Rank (MRR) at $k \in \{1, 5, 10\}$ on the held-out evaluation set of approximately 2,000 image-caption pairs on different methodologies. MRR is computed for text-to-image retrieval task meaning each Burmese caption is used to query and rank the 2,000-image evaluation dataset. LoRA at rank 32 achieves the strongest retrieval quality across all values of k , with MRR@10 exceeding 0.34, indicating that the correct image is retrieved within the top-10 candidates for more than one-third of Burmese text queries. DoRA at the same rank($r=32$) performs similarly, which indicates that DoRA does not provide a meaningful advantage when adaptor capacity is sufficient (Liu et al., 2024). This is consistent with prior observations in literature, where DoRA’s gains are more observable at low rank (Han et al., 2024; Liu et al., 2024). DoRA with rank 16 retains approximately 86% of the LoRA ($r=32$) performance with a nearly 50% reduction in trainable parameters. This indicates that DoRA with lower rank is justifiable where storage, cost, and memory bandwidth constraints exist.

The worst-performing was AdaLoRA, achieving only 0.2374 MRR@10 despite using a similar number of parameters as DoRA ($r=16$). A deep dive reveals that query and key layers are heavily pruned in early, and middle layers (5,7, and 8) received near-zero rank allocations. This aggressive pruning may be detrimental in cross-modal settings where precise query-key interactions are essential for the alignment of text with visual embedding. This behaviour has been noted in the literature (Zhang et al., 2023a; Su et al., 2025). AdaLoRA consistently prefers to allocate more budget to FFN modules and top layers (Zhang et al., 2023a). Since query and key projections play a critical role in aligning representations across modalities, the aggressive pruning of attention layers suffer from sharp accuracy degradation (Su et al., 2025).

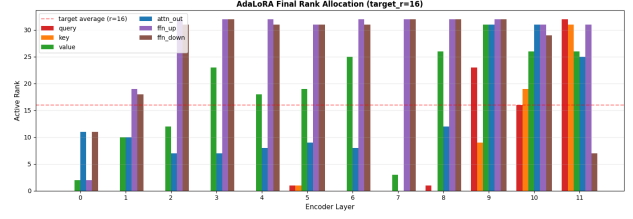


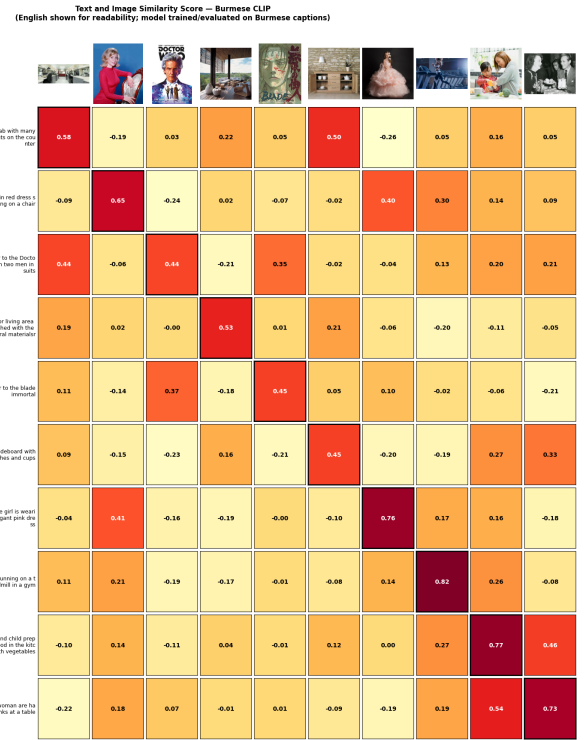
Figure 1: AdaLoRA Final Rank Allocation

5.2 Performance Comparison with NLLB-CLIP

The MRR for text-to-image retrieval was calculated for NLLB-CLIP Base and NLLB-CLIP Large under zero-shot condition (Visheratin, 2023b). Burmese-CLIP (LoRA, $r=32$) outperforms across all values of k , with MRR@10 improving from 0.0480 to 0.3441, approximately seven-fold increase. Whilst NLLB-CLIP’s text encoder possess Burmese language understanding, this result indicates that fine-tuning with a task-specific objective yields a better result, which is often the case in multilingual models that require explicit cross-modal supervision to achieve competitive retrieval performance (Wang et al., 2024). It is important to note that this comparison is against zero-shot NLLB-CLIP and therefore, the improvement could be attributed to task-specific fine-tuning itself rather than this specific methodology. A Burmese fine-tuned NLLB-CLIP would provide a stricter comparison.

5.3 Cross-Modal Alignment: Similarity Matrix

The Figure 2 presents a 10×10 cross-modal similarity matrix of best-performing model (LoRA $r = 32$) on a random sample of 10 image-caption pairs. It reports the cosine similarity between the text and image pairs. The diagonal entries achieve cosine similarities ranging from 0.44 to 0.82, with a mean diagonal score of approximately 0.62. The result demonstrates that meaningful cross-modal alignment has been established. Row labels in the figure are replaced with English captions for readability.


 Figure 2: Similarity matrix (LoRA $r = 32$ model)

6 Limitations

First, the data used to train the model are machine-translated, which could cause semantic drift and translation artefacts. Second, the reported improvement over NLLB-CLIP is measured against a zero-shot baseline, and therefore, part of this gain may be attributable to task-specific fine-tuning itself rather than to the specific PEFT methodology evaluated here. A fine-tuned NLLB-CLIP baseline would offer a stricter point of comparison.

and is left for future work. Third, the results presented here represent a lower bound on what is achievable for Burmese vision–language modelling. A natural next step would be to curate a native Burmese image-text pair dataset to determine whether a better result can be achieved with full fine-tuning or PEFT-adapted models trained on non-machine-translated data. Fourth, a comparison of alternative text encoders should be studied to understand the impact of tokenisation choice on downstream performance, especially given the unique orthographic and segmentation challenges posed by the Burmese language. Fifth, the explanation of the poor performance by the AdaLoRA remains hypothetical since no ablation study was performed. Finally, the evaluation is restricted to a single contrastive retrieval task on a single language; the generalisation of these findings to other low-resource languages, or to non-contrastive objectives, remains an open question.

7 Conclusion

In this paper, we present the Burmese-CLIP trained on 20,000 machine-translated image-text pairs using various PEFT methods. The best Burmese-CLIP model, LoRA ($r=32$) outperformed Zero-Shot NLLB-CLIP Base and NLLB-CLIP Large on Mean Reciprocal Rank for text-to-image retrieval at $k \in \{1, 5, 10\}$ using the same held-out evaluation set of 2,000 image-text pairs. We also explored how other LoRA variants methodologies perform in the low-resource language setting. Across four LoRA variants evaluated under identical conditions, LoRA at rank 32 produced the strongest retrieval quality, suggesting that the LoRA formulation is often sufficient and does not require more complex adaptive variants to be effective in this setting. But the comparison also showed that alternative approaches are worth exploring. DoRA at rank 16 retained 86% of LoRA’s retrieval performance with roughly half the trainable parameters, offering a favourable efficiency trade-off for resource-constrained deployments, while AdaLoRA’s under-performance highlighted that adaptive methods can fail due to misallocation of ranks.

References

- Aung, T., Montalan, J. R., Ngui, J. G., and Limkonchotiawat, P. (2026). Burmese-san: Burmese nlp benchmark for evaluating large language models.
- Bianchi, F., Attanasio, G., Pisoni, R., Terragni, S., Sarti, G., and Lakshmi, S. (2021). Contrastive language-image pre-training for the italian language.
- Carlsson, F., Eisen, P., Rekathati, F., and Sahlgren, M. (2022). Cross-lingual and multilingual CLIP. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Odijk, J., and Piperidis, S., editors, *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6848–6854, Marseille, France. European Language Resources Association.
- Cartella, G., Baldrati, A., Morelli, D., Cornia, M., Bertini, M., and Cucchiara, R. (2023). Openfashionclip: Vision-and-language contrastive learning with open-source fashion data.
- Chen, Z., Liu, G., Zhang, B.-W., Ye, F., Yang, Q., and Wu, L. (2022). Altclip: Altering the language encoder in clip for extended language capabilities.
- Chung, D., Shin, D., Sung, Y., Moon, S., Jeon, J., and Lee, B.-J. (2025). uclip: Parameter-efficient multilingual extension of vision-language models with unpaired data.
- Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2020). Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

- Ding, C., Aye, H. T. Z., Pa, W. P., Nwet, K. T., Soe, K. M., Utiyama, M., and Sumita, E. (2019). Towards burmese (myanmar) morphological analysis: Syllable-based tokenization and part-of-speech tagging. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 19(1).
- Ding, N., Qin, Y., Yang, G., Wei, F., Yang, Z., Su, Y., Hu, S., Chen, Y., Chan, C.-M., Chen, W., et al. (2023a). Delta tuning: A comprehensive study of parameter efficient methods for pre-trained language models. *Nature Machine Intelligence*.
- Ding, N., Qin, Y., Yang, G., Wei, F., Yang, Z., Su, Y., Hu, S., Chen, Y., Chan, C.-M., Chen, W., et al. (2023b). Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5:220–235.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.
- Han, Z. et al. (2024). Parameter-efficient fine-tuning in large models: A survey of methodologies. *arXiv preprint arXiv:2410.19878*.
- Htet, A. K. and Dras, M. (2025). Myanmar xnli: Building a dataset and exploring low-resource approaches to natural language inference with myanmar.
- Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2022). LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.
- Li, J., Shakhnarovich, G., and Yeh, R. A. (2022). Adapting clip for phrase localization without further training. *arXiv preprint arXiv:2204.03647*.
- Liao, H.-T. (2017). Encoding for access: how zawgyi success impedes full participation in digital myanmar. *SIGCAS Comput. Soc.*, 46(4):18–24.
- Lin, Y., Chen, M., Wang, W., Wu, B., Li, K., Lin, B., Liu, H., and He, X. (2023). Clip is also an efficient segmenter: A text-driven approach for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15305–15314.
- Liu, S.-Y., Wang, C.-Y., Yin, H., Molchanov, P., Wang, Y.-C. F., Cheng, K.-T., and Chen, M.-H. (2024). DoRA: Weight-decomposed low-rank adaptation. In *International Conference on Machine Learning*.
- Maung, Z. M. and Mikami, Y. (2008). A rule-based syllable segmentation of myanmar text. In *International Joint Conference on Natural Language Processing*.
- Nayeem, M. T., Alqahtani, S., Laskar, M. T. R., Mohiuddin, T., and Bari, M. S. (2025). Beyond fertility: Analyzing str as a metric for multilingual tokenization evaluation.
- Ng, R., Nguyen, T. N., Huang, Y., Tai, N. C., Leong, W. Y., Leong, W. Q., Yong, X., Ngui, J. G., Susanto, Y., Cheng, N., Rengarajan, H., Limkonchotiwat, P., Hulagadri, A. V., Teng, K. W., Tong, Y. Y., Siow, B., Teo, W. Y., Lau, W., Tan, C. M., Ong, B., Ong, Z. H., Montalan, J. R., Chan, A., Antonyrex, S., Lee, R., Choa, E., Tat-Wee, D. O., Liu, B. J. D., Tjhi, W. C., Cambria, E., and Teo, L. (2025). Sea-lion: Southeast asian languages in one network.
- Oo, Y. M., Wattanavekin, T., Li, C., De Silva, P., Sarin, S., Pipatsrisawat, K., Jansche, M., Kjartansson, O., and Gutkin, A. (2020). Burmese speech corpus, finite-state text normalization and pronunciation grammars with an application to text-to-speech. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6328–6339, Marseille, France. European Language Resources Association.

- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.
- Rust, P., Pfeiffer, J., Vulić, I., Ruder, S., and Gurevych, I. (2021). How good is your tokenizer? on the monolingual performance of multilingual language models.
- Si Thu, P. P. S. (2024). Development and evaluation of myanmar gpt: A language model for myanmar natural language processing. *SSRN*.
- Su, Z., Zhang, Z., Zhang, W., Liu, Z., Xu, G., and Zhang, W. (2025). Shrp: Specialized head routing and pruning for efficient encoder compression.
- Szep, M., Rueckert, D., von Eisenhart-Rothe, R., and Hinterwimmer, F. (2025). Fine-tuning large language models with limited data: A survey and practical guide. *Transactions of the Association for Computational Linguistics*.
- Thengane, V., Khan, S., Hayat, M., and Khan, F. (2022). Clip model is an efficient continual learner. *arXiv preprint arXiv:2210.03114*.
- Thet, T. T., Na, J.-C., and Ko, W. K. (2008). Word segmentation for the myanmar language. *J. Inf. Sci.*, 34(5):688–704.
- United Nations Department of Economic and Social Affairs (2024). World population prospects 2024. <https://population.un.org/wpp/>.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- Visheratin, A. (2023a). Laion-coco-nllb dataset.
- Visheratin, A. (2023b). Nllb-clip – train performant multilingual image retrieval model on a budget. *3rd Workshop on Efficient Natural Language and Speech Processing at NeurIPS 2023*.
- Wang, T., Li, F., Zhu, L., Li, J., Zhang, Z., and Shen, H. T. (2024). Cross-modal retrieval: A systematic review of methods and future directions.
- World Bank (2023). Myanmar population. <https://data.worldbank.org/indicator/SP.POP.TOTL?locations=MM>.
- Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., and Raffel, C. (2021). mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498.
- Yang, A., Pan, J., Lin, J., Men, R., Zhang, Y., Zhou, J., and Zhou, C. (2023). Chinese clip: Contrastive vision-language pretraining in chinese.
- Zhang, Q., Chen, M., Bukharin, A., Karampatziakis, N., He, P., Cheng, Y., Chen, W., and Zhao, T. (2023a). AdaLoRA: Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations*.
- Zhang, Z., Wang, J., and Wang, X. E. (2023b). Parameter-efficient cross-lingual transfer of vision and language models via translation-based alignment.

Perspective-Aware Billboard Corner Detection with a Web-Based End-to-End System

Sukriti Dhang^{*1,2}, Tingyu Zhang^{*1}, Mimi Zhang³, Soumyabrata Dev^{1,2,3}

¹*School of Computer Science, University College Dublin, Ireland*

²*The ADAPT Research Centre, Dublin, Ireland*

³*School of Computer Science and Statistics, Trinity College Dublin, Ireland*

** These authors contributed equally to this work.*

Abstract

Billboard placement technologies have become increasingly important in digital advertising, multimedia production, and augmented reality applications. Accurate computation of the four corner coordinates of a billboard are essential for the seamless integration of new billboard content. We formulate a loss function to minimise the error across the computed four corner coordinates while accounting for varying perspective alignments. We evaluate the accuracy of the corner coordinate predictions using the Root Mean Square Error (RMSE), obtaining scores of 4.0682 for x-coordinates and 3.8553 for y-coordinates. These results indicate a high level of accuracy in corner coordinate computation. Next, this paper presents a web-based automated billboard system (ABS) that integrates video upload, billboard detection, localisation, and placement within a unified end-to-end framework. The method's practical applications extend to media companies and digital content creators, offering valuable insights to video editors and broadening its impact.

Keywords: Digital advertising, Corner coordinates detection, Billboard detection, Web application

1 Introduction

Advertisements in multimedia videos play a crucial role in capturing the attention of a broad audience. Recent interest focuses on integrating advertising boards with modern techniques [Krishna and Aizawa, 2018], enhancing effectiveness in the evolving digital advertising landscape. For effective billboard integration, accurate detection and corner coordinate computation are crucial. Traditionally, video editors manually inspect image frames and mark the opposite corners to insert new advertisements, making it a time-consuming and labor-intensive process. This technological gap significantly affects industries such as advertising and sports broadcasting, as most existing research approaches billboard detection and analysis as independent tasks, thereby constraining the development of fully integrated intelligent advertising systems that can jointly detect, analyse, and optimise billboard placement in real-world environments.

The primary contribution of this paper is the development of a novel corner detection algorithm. We have also extended this to design and implement an end-to-end automated billboard system that supports practical user interaction, video processing, and advertisement management workflows. This work focuses on the system-level integration and deployment of existing billboard detection [Dhang et al., 2024a] models within a scalable web-based application framework. This is particularly crucial in applying the billboard system, as end users are often media companies, video editors or content creators. The main contributions to this work ¹ are: *i)* Extended traditional corner detection with proposed Corner Detection for Advert (CDANet), enabling precise corner computation under different perspective alignment. *ii)* Defined a custom loss function that minimises

¹To facilitate the reproducibility of this research, the code of this paper is made available: <https://github.com/sukritidhang/CEDANet-cornercoord-detection>

error across all four corners, improving localisation accuracy to support multiple billboards in a video frame.

iii) Design and implementation of a user-friendly web-based automatic billboard system interface, Provide real-time preview and interactive editing functionalities for billboard analysis and visualisation.

The rest of the paper is organized as follows: Section 2 provides the background and related work. Section 3, outline the proposed methodology. Section 4 presents evaluation results. Finally, Section 5 concludes the paper by summarising the results.

2 State of the Art

Earlier edge detection methods relied on first-order [Sobel et al., 1968] and second-order [Torre and Poggio, 1986] operators, which were limited to predicting axis-aligned bounding boxes (x_1, y_1, x_2, y_2) along with the width (w) and height (h) of the object. Difficulties arise when the perspective alignment of the object deviates from the conventional axis $(x_1, y_1, x'_2, y'_2, x_3, y_3, x'_4, y'_4)$. These perspectives aid in comprehensively understanding an object's dimensions and spatial relationships by learning its bounding box, making accurate computation of corner points for effective image registration [Dhang et al., 2025]. Recent deep learning methods have improved object detection and edge prediction [Liu, 2023]. Wibisono *et al.* [Wibisono and Hang, 2020] combined traditional edge detectors with CNNs, while RCNN [Girshick et al., 2014] and Faster RCNN [Ren et al., 2015] introduced regression-based bounding box refinement. RetinaNet [Lin et al., 2017] further enhanced general object detection, but such methods primarily focus on standard objects and rectangular bounding boxes. Dhang *et al.* [Dhang et al., 2024b] proposed a billboard segmentation framework that localises billboards and extracts the corner points. The method employs Otsu's thresholding, followed by erosion and dilation to reduce noise and reconnect fragmented areas, with contour detection used to identify the final billboard boundaries. This model computes billboard corner coordinates rather than estimating perspective-aligned billboard corner coordinates.

Despite advances in complex model architectures, scalability and deployment remain major challenges. Smart advertisement insertion has gained attention, though most studies address detection, classification, segmentation, and inpainting as independent tasks. For instance, Bulkan *et al.* [Bulkan et al., 2023] proposed a smart advertisement insertion framework with duration constraints. Cloud-based systems such as AdNext [Akgul et al., 2020] depend heavily on CNNs and remote infrastructure, limiting suitability for latency-sensitive environments, whereas lightweight systems like LiveSense [Chen et al., 2019] improve scalability using pre-trained multimodal models and reduced cloud dependency. Despite these advances, robust deployment across diverse devices and network conditions remains difficult. Collectively, these works indicate a shift toward intelligent and context-aware advert insertion systems.

3 Methodology

3.1 Dataset

In this work, we utilised the ALOS dataset [Dev et al., 2019], to train the CDA model, we chose 1,750 images as the positive class (images containing advert) and 1,750 images as the negative class (images without advert) [Heonho, 2022][Bansal, 2019]. For evaluation, four publicly available non-copyrighted advert videos were randomly selected from YouTube, covering varying viewpoints and angles. Due to the limited availability of advert-focused video datasets with ground-truth corner annotations, the evaluation was conducted on this small custom sample. The description of the dataset is summarised in Table 1.

3.2 Proposed Corner Detection for Advert (CDA) model

The perspective alignment of the object deviates from the conventional axis $(x_1, y_1, x'_2, y'_2, x_3, y_3, x'_4, y'_4)$, introducing complexities in corner coordinate computation. Understanding these types of perspective viewpoints is crucial, as they offer different angles for observing and representing an object. These viewpoints include:

Table 1: Dataset description

Name	ALOS Dataset	Video Dataset			
		Video 1	Video 2	Video 3	Video4
Total number of images	9315	2874	1231	1573	4022
Dimension	512 x 512	1280 x720	1280 x720	1070 x 720	1280 x720
Color grading	RGB	RGB	RGB	RGB	RGB
Billboard images	1750	1333	-	1573	2710
No-billboard images	1750	1514	1231	-	1312

i) Front View ii) Side View iii) Perspective View. The Figure 1 depicts the different viewpoints of the billboard class.

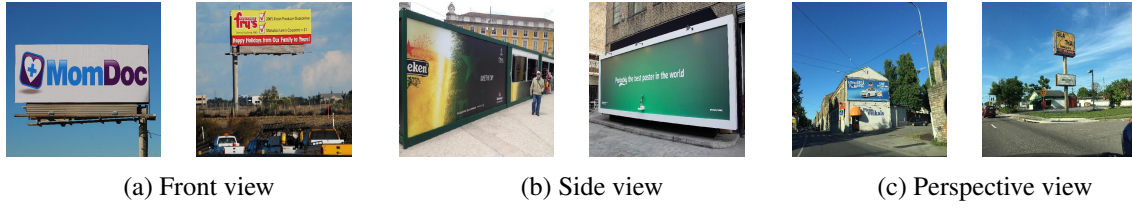


Figure 1: Represents different types of perspective viewpoint.

We adopt pretrained AdSegNet model [Dhang et al., 2024b] to generate the probabilistic mask. The model employs an encoder-decoder architecture, where the encoder comprises three convolution layers with filters [64, 128, 256] and three max-pooling layers sized (3×3) , and the decoder includes four deconvolution layers and three up-sampling layers. The output of the encoder block is fed into the decoder block with deconvolution layers consisting of filters [256, 128, 64]. The final deconvolution layer comprises a filter with dimensions (1×1) for the localization output. To enable billboard corner prediction, the decoder output was flattened and connected to an additional fully connected regression module. This module consists of two FC layers with 1024 neurons and ReLU activation, with a dropout layer (rate = 0.5) applied after the first FC layer. The final FC layer contains eight neurons with sigmoid activation, representing the normalised coordinates of the four billboard corners. These modifications allow the network to directly regress corner points. To accurately predict the coordinates, we formulate the loss function as follows: Let $g = [x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4]$ denote a ground-truth coordinates, represented by left top coordinate (x_1, y_1) , right top coordinate (x_2, y_2) , left bottom coordinate (x_3, y_3) and right bottom coordinate (x_4, y_4) . The predicted coordinates as $\hat{p} = [\hat{x}_1, \hat{y}_1, \hat{x}_2, \hat{y}_2, \hat{x}_3, \hat{y}_3, \hat{x}_4, \hat{y}_4]$. The mean squared error (MSE) for each individual coordinate (x and y) is calculated separately, and then these individual losses are summed to accurately predict each of the four coordinates independently.

The loss function used for computing corner coordinates provides valuable insight, as it is designed to minimize the error across all four corner coordinates. Unlike standard regression losses that focus on individual coordinates (such as the top-left or bottom-right corners), this loss function considers all four corners of the bounding box, which is crucial for accurate and complex perspectives and varying positions. The total loss is the sum of the squared errors between the predicted and ground truth coordinates for each corner. For each corner, the squared difference between the predicted and actual x -coordinates and y -coordinates is computed, and these errors are averaged over the dataset. The formula for corner's loss is defined as $loss_{x_1} = \frac{1}{N} \sum_{i=1}^N (\hat{x}_{1i} - x_{1i})^2$, $loss_{y_1} = \frac{1}{N} \sum_{i=1}^N (\hat{y}_{1i} - y_{1i})^2$. Similarly, we calculated the $loss_{x_2}, loss_{y_2}, loss_{x_3}, loss_{y_3}, loss_{x_4}, loss_{y_4}$. We define the total loss function of the CDA model by the following equation:

$$loss_coordinates = (loss_{x_1} + loss_{y_1} + loss_{x_2} + loss_{y_2} + loss_{x_3} + loss_{y_3} + loss_{x_4} + loss_{y_4}).$$

By considering all four corners, the model can better account for non-axis-aligned bounding boxes, ensuring accurate predictions even when billboards are in varying visibility levels (minority billboard area, and majority billboard area) and different perspective alignment.

3.3 Automatic Billboard System (ABS) architecture

Next, a automated billboard system has been designed. This section outlines the development of a responsive, efficient system design that leverages React for the front-end and Python for the back-end services. Figure 2 demonstrates the flow diagram of the ABS. This end-to-end system integrates all necessary components, such as data processing, detection, localisation and integration, into one cohesive pipeline. In this web applica-

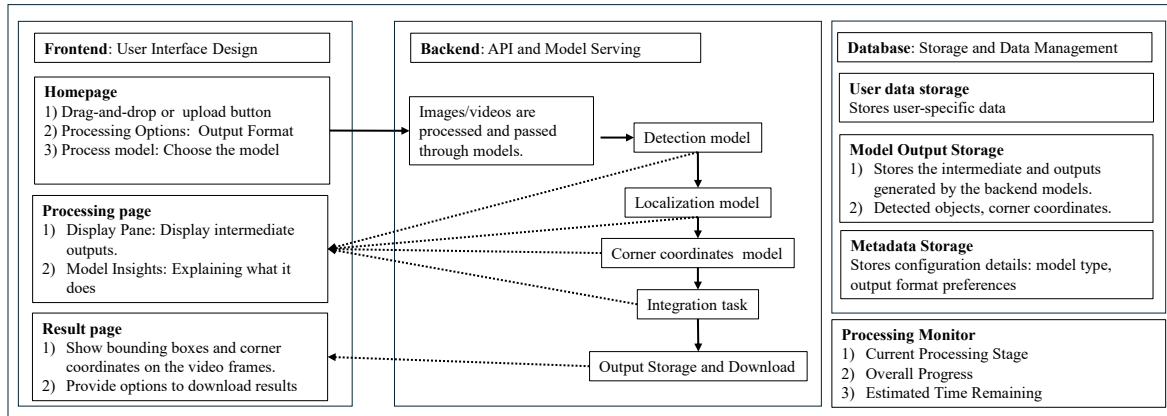


Figure 2: Automated billboard placement system architecture

tion, the Billboard Video Upload (BVU) component was designed for uploading videos. Uploaded videos are temporarily stored within the web application, allowing users to manage video content directly through the interface. In addition, the system provides video preview capabilities and displays real-time video metadata such as file format, resolution, duration, and file size.

3.3.1 System Architecture

The system architecture adopts a frontend-backend model to support efficient user interaction, real-time processing, and seamless integration with deep learning-based billboard detection models.

Front-end Technology: The frontend of the application was developed using React and Ant Design component library. React was selected due to its component-based architecture, which is highly suitable for building complex and interactive user interfaces. In addition, React’s virtual Document Object Model (DOM) allows efficient rendering and updating of interface components, which is particularly beneficial for real-time previewing and interactive billboard manipulation tasks within the application. The Ant Design framework was integrated to provide a comprehensive collection of high-quality user interface components with consistent styling and responsive layouts. Furthermore, the framework improves the overall user experience through modern design principles and extensive documentation support. Figure 3 illustrates the front-end interface used for BVU and processing.

Back-end Technology: The backend system was implemented using Python for computer vision, and deep learning applications. Furthermore, Python offers seamless implementation with deep learning frameworks enabling efficient deployment of the billboard detection models proposed in [Dhang et al., 2024a, Dhang et al., 2024b]. A RESTful API architecture was adopted to establish standardised communication between the front-end and back-end systems. The API-based design improves modularity, maintainability, and scalability while allowing the front-end interface to communicate efficiently with the detection and processing modules.

4 Experimental results

Our evaluation includes both static images and video frames, demonstrating the model’s ability to compute corner-coordinates of the billboard under various visibility levels and different perspective alignments. We then evaluated the performance of the ABS.

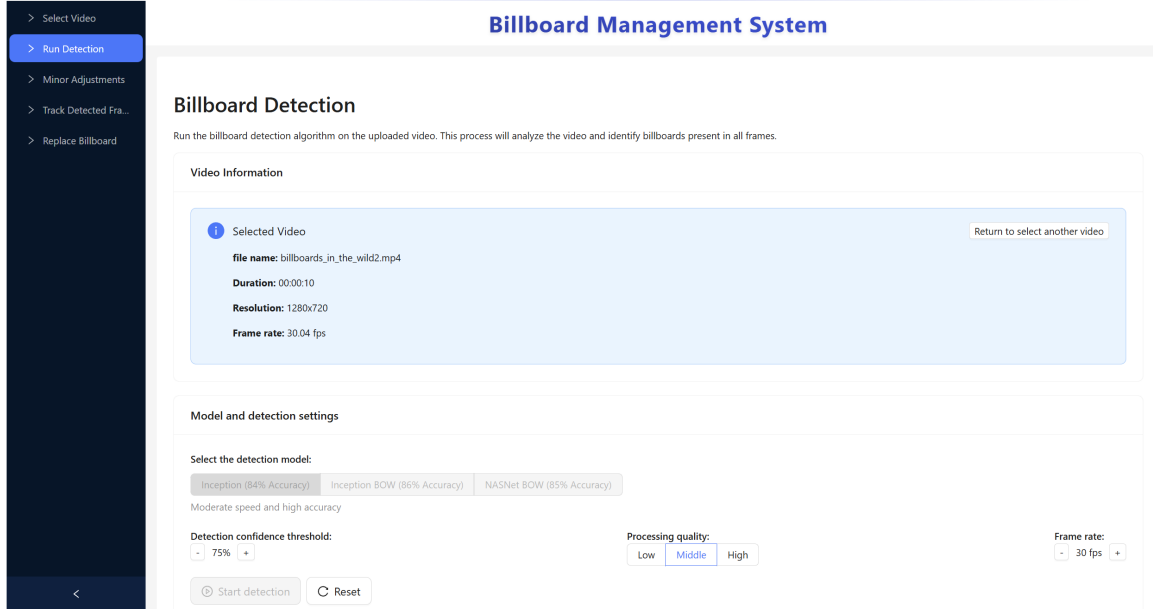


Figure 3: Interface for billboard video upload

4.1 CDA model performance

We assess the accuracy of the predicted (pred) four corner coordinates against the ground truth (gt) corner coordinates, by calculating the root mean square error (RMSE) metric as expressed $RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2}$, the square root of the mean of the squared differences between the pred and gt coordinate values. where, x_i is the average of all the four corner coordinates along x-axis of the ground truth and $\hat{x}_i$ is the average of all the four corner coordinates along x-axis of the pred coordinates, n is the number of testing images. Similarly, RMSE is calculated for y-axis. We also determine the correlation coefficient between the pred coordinates and gt coordinates using the equation $r = \frac{\sum_{i=1}^n (x_i - \bar{x})(\hat{x}_i - \bar{\hat{x}})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (\hat{x}_i - \bar{\hat{x}})^2}}$ where, $\bar{x}$ is the mean of the four corner coordinates of the gt and $\bar{\hat{x}}$ is the mean of the pred four corner coordinates.

Table 2: Summarisation of RMSE and correlation coefficient of models, with custom loss function and without custom loss function on ALOS dataset (All results are average across the dataset).

Model	RMSE ↓		Correlation coefficient ↑	
	X coord	Y coord	X coord	Y coord
With custom loss function				
Proposed CDA	4.0682	3.8553	0.9875	0.9647
UNet	6.0244	7.1055	0.9696	0.8553
Without custom loss function [Dhang et al., 2024b]				
AdSegNet	17.7350	15.3519	0.8257	0.6416
UNet	22.9443	18.6021	0.6988	0.4845
Sobel edge detection method	17.7350	15.3519	0.8257	0.6416

Table 2 presents the results obtained with and without the proposed custom loss function. Compared with AdSegNet, the proposed CDA model demonstrates superior performance in corner coordinate prediction. The proposed CDA reduces the RMSE by 77.06% and 74.88% for the x- and y-coordinates, respectively. Additionally, the correlation coefficients improve by 19.5% for the x-coordinate and 50.35% for the y-coordinate. These improvements indicate that the proposed CDA model provides more accurate corner coordinates that are closer to the gt coordinates. Figure 4a compares the average gt x-coords (in x-axis) with the pred x-coords from the proposed CDA model (in y-axis). Figure 4b and 4c show the comparison between the average gt x-coords and the pred x-coords using the U-Net and Sobel methods, respectively. Similarly, we plot the average pred y-coords obtained using the proposed CDA model, U-Net and Sobel methods in Figure 4d, 4e and 4f, respec-

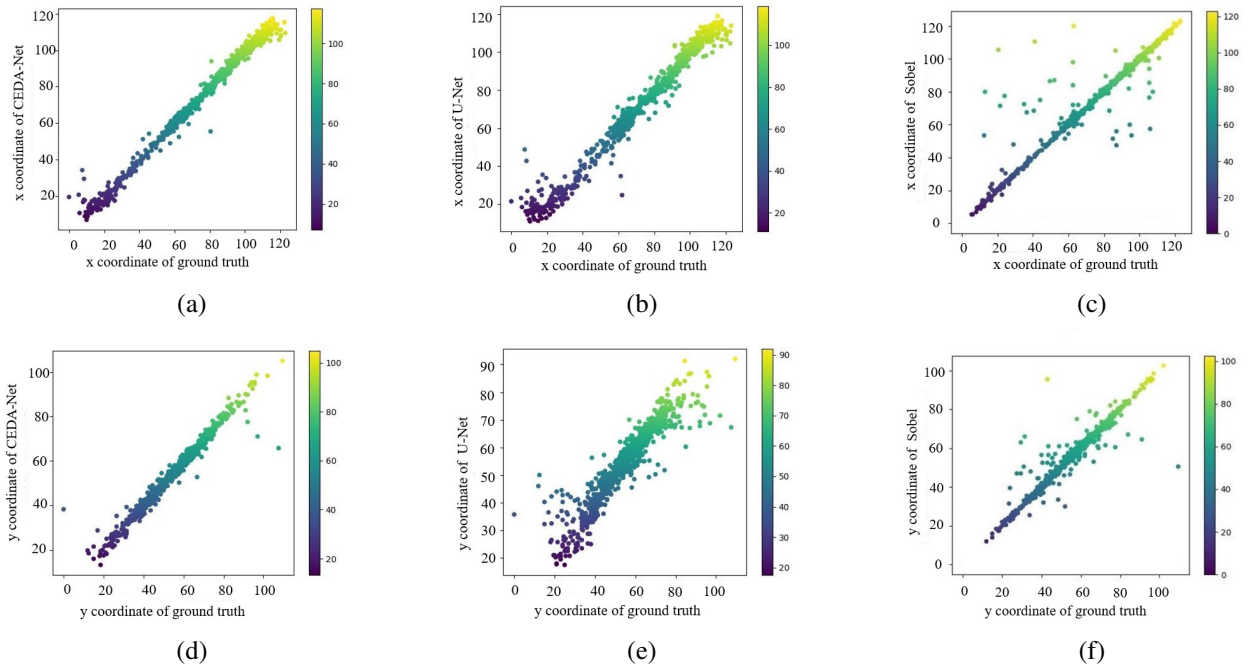


Figure 4: Representation of correlation coefficient plot. We represent (a) x-coords of proposed CDA vs ground truth coords, (b) x-coords of U-Net vs ground truth coords, (c) x-coords of sobel detection method vs ground truth coords, (d) y-coords of proposed CDA vs ground truth coords, (e) y-coords of U-Net vs ground truth coords, (f) y-coords of sobel detection method vs ground truth coords.

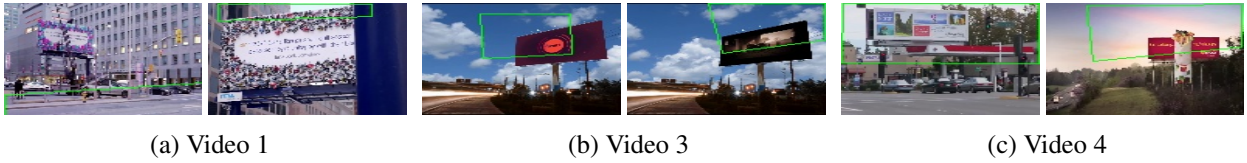


Figure 5: Frame-level extraction of billboard corner coordinate by UNet model.

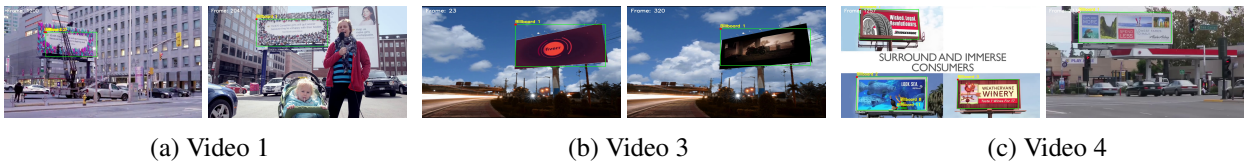


Figure 6: Frame-level extraction of billboard corner coordinates using the proposed CDA model.

tively. The selected representative cases in Figure 5 and 6 reflect billboards in different perspective alignment and visibility levels to ensure that evaluation covers a wide range of conditions for videos.

4.2 ABS Performance

This section focuses on system integration, operational functionality, interface usability, and deployment performance within the proposed web-based framework. The system successfully processes uploaded videos by extracting frames at configurable intervals and performing billboard localisation on the extracted frames. Figure 7 depicts the billboard detection interface displaying test results, confidence score, and whether the billboard exists.

The backend API communication operated reliably through testing, enabling efficient data exchange between the front-end and deep learning models. Resource utilisation monitoring tracked CPU, memory, and

Test results

Table View Gallery View Export Options

Frame	Original Image	Billboards exist	Test results	Confidence	Operate
1		✓	Detected	0.934	Check Save
21		✓	Detected	0.932	Check Save

Figure 7: Billboard detection interface with results display

Table 3: ABS: Resource utilisation summary

Operation	CPU	Memory	Network
Video Upload	15–25%	120–180 MB	10–50 MB/s
Billboard analysis	70–90%	1.2–1.8 GB	5–10 MB/s
UI Rendering	5–15%	150–250 MB	0.1–1 MB/s
Idle Application	1–3%	90–120 MB	< 0.1 MB/s

The color intensity indicates the relative resource demand of each operation, where:

Red denotes high utilisation.

Yellow denotes moderate utilisation.

Green denotes low utilisation.

network usage of ABS during operation of the implemented components as tabulated in Table 3, confirming that most operations maintain reasonable resource usage. Thus, ABS represents the most computationally intensive process due to the high processing demands of the deep learning model.

5 Conclusion

Billboard analysis has become increasingly important in digital advertising and multimedia production. We formulate a loss function to minimise the error across the computed four corner coordinates while accounting for varying perspective alignments. Our findings indicate predicting corner coordinates in the ALOS dataset with RMSE scores of 4.0682 for x-coordinates and 3.8553 for y-coordinates, a high level of accuracy in corner coordinate computation. We also presented a web-based automated system that integrates video upload, frame extraction, billboard detection, interactive editing, and result presentation into a unified pipeline.

Overall, the proposed system demonstrates the feasibility of integrating automated billboard placement and management into a scalable web-based platform, offering practical benefits for digital advertising and multimedia editing. Although large-scale annotation of billboard corner coordinates is labour-intensive, future work will focus on expanding the evaluation using more comprehensive advertisement video datasets with ground-truth annotations.

Acknowledgments

This work was conducted with the financial support of the Research Ireland Centre for Research Training in Digitally-Enhanced Reality (d-real) under Grant No. 18/CRT/6224. This research was conducted with the financial support of Research Ireland under Grant Agreement No. 13/RC/2106_P2 at the ADAPT Research Centre at University College Dublin. ADAPT, the Research Centre for AI-Driven Digital Content Technology, is funded by Science Foundation Ireland through the Research Centres Programme.

References

[Akgul et al., 2020] Akgul, T., Ozcan, S., and Iplik, A. (2020). A cloud-based end-to-end server-side dynamic ad insertion platform for live content. In *Proceedings of the 11th ACM Multimedia Systems Conference*, pages 361–364.

- [Bansal, 2019] Bansal, P. (2019). Intel Image Classification. <https://www.kaggle.com/datasets/puneet6060/intel-image-classification>.
- [Bulkan et al., 2023] Bulkan, U., Dagiuklas, T., and Iqbal, M. (2023). Supereye: smart advertisement insertion for online video streaming. *Multimedia Tools and Applications*, 82(6):9361–9379.
- [Chen et al., 2019] Chen, X., Nguyen, T. V., Shen, Z., and Kankanhalli, M. (2019). Livesense: Contextual advertising in live streaming videos. In *Proceedings of the 27th ACM international conference on multimedia*, pages 392–400.
- [Dev et al., 2019] Dev, S., Hossari, M., Nicholson, M., McCabe, K., Conran, A. N. C., Tang, J., Xu, W., and Pitié, F. (2019). The ALOS dataset for advert localization in outdoor scenes. In *2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–3. IEEE.
- [Dhang et al., 2024a] Dhang, S., Idogho, A., Zhang, M., and Dev, S. (2024a). Towards accurate billboard detection: an ablation and benchmarking study of deep learning models. In *IET Conference Proceedings CP887*, volume 2024, pages 178–185. IET.
- [Dhang et al., 2024b] Dhang, S., Zhang, M., and Dev, S. (2024b). Adsegnet: a deep network to localize billboard in outdoor scenes. *Signal, Image and Video Processing*, 18(10):7221–7235.
- [Dhang et al., 2025] Dhang, S., Zhang, M., and Dev, S. (2025). Labinet-an approach for seamlessly integrating new advertisement into an existing scene. *IEEE Transactions on Artificial Intelligence*.
- [Girshick et al., 2014] Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587.
- [Heonho, 2022] Heonho (2022). Unpaired Day and Night cityview images. <https://www.kaggle.com/datasets/heonh0/daynight-cityview>.
- [Krishna and Aizawa, 2018] Krishna, O. and Aizawa, K. (2018). Billboard saliency detection in street videos for adults and elderly. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 2326–2330.
- [Lin et al., 2017] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.
- [Liu, 2023] Liu, Z. (2023). A deep neural framework to detect individual advertisement (ad) from videos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3578–3587.
- [Ren et al., 2015] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.
- [Sobel et al., 1968] Sobel, I., Feldman, G., et al. (1968). A 3x3 isotropic gradient operator for image processing. *a talk at the Stanford Artificial Project in*, pages 271–272.
- [Torre and Poggio, 1986] Torre, V. and Poggio, T. A. (1986). On edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2):147–163.
- [Wibisono and Hang, 2020] Wibisono, J. K. and Hang, H.-M. (2020). Traditional method inspired deep neural network for edge detection. In *2020 IEEE international conference on image processing (ICIP)*, pages 678–682. IEEE.

Automask: Promptable Concept Segmentation with SAM3 for Glacier 3D Reconstruction from UAV Imagery

SiCheng Zhao^{1,2,3,†}, Matthew Byrne^{2,†}, and Soumyabrata Dev^{1,2,3}

¹*The ADAPT Research Centre, Dublin, Ireland*

²*School of Computer Science, University College Dublin, Dublin, Ireland*

³*School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland*

[†]*These authors contributed equally to this work.*

Abstract

Photogrammetric surveys are the standard way to measure glacier retreat and surface change, but the sky, survey aircraft, and ground crew that appear in the imagery destabilise Structure-from-Motion (SfM) pose estimation and leave halos and ghosting in the reconstructed radiance field. Masking every frame by hand, the standard remedy, costs an analyst one to two hours per 500-image survey and scales linearly with dataset size. We show that this manual stage can be replaced by a single short text prompt, and quantify what that substitution is worth. *Automask* applies one Promptable Concept Segmentation (PCS) prompt from Segment Anything Model 3 (SAM3) across a whole survey, with no fine-tuning and no per-image intervention, ahead of an unmodified COLMAP and Gaussian Splatting pipeline. On a 589-image uncrewed aerial vehicle (UAV) glacier survey, prompted masking improves every metric over an unmasked baseline across ten end-to-end runs: peak signal-to-noise ratio (PSNR) by ~ 1.2 dB, structural similarity index (SSIM) by 0.065, Learned Perceptual Image Patch Similarity (LPIPS) by 0.096, and mean SfM reprojection error from 0.556 px to 0.506 px. We also characterise the glacier-specific failure modes, chiefly ice read as sky under overcast lighting, that bound what a fully automatic single-prompt run can achieve. The implementation is publicly available at <https://github.com/AppleJack9050/automask>.

1 Introduction

Repeated photogrammetric surveys of glaciers yield metre-scale estimates of front retreat, surface elevation change, and ice-flow velocity, but each survey produces hundreds to thousands of overlapping images that must be reconstructed in 3D [Westoby et al., 2012, Geissler et al., 2021]. The standard pipeline is two-stage: Structure-from-Motion (SfM) [Schönberger and Frahm, 2016] recovers camera poses and a sparse point cloud, after which a dense stage produces a continuous radiance field, increasingly with Gaussian Splatting [Kerbl et al., 2023] rather than Multi-View Stereo [Schönberger et al., 2016]; both stages are typically run through COLMAP [Schönberger and Frahm, 2016, Schönberger et al., 2016].

Glacier imagery stresses both stages. Textureless sky destabilises SIFT correspondences at cloud edges and horizons [Lowe, 2004, Gindraux et al., 2017], inflating bundle-adjustment residuals; moving objects (helicopters, ground crew) introduce location-dependent supervision that survives SfM outlier rejection and reappears as ghosting in the splat optimisation; specular and dynamic surfaces violate photometric consistency; and shaded firm or smooth ice under-fires the SIFT detector, leaving the dense stage poorly anchored. Per-image manual masking, the standard remedy, costs an analyst one to two hours per 500-image survey and scales linearly with dataset size.

Promptable foundation models reshape this trade-off. Earlier pipelines chaining a text-conditioned detector [Liu et al., 2024] with a class-agnostic segmenter are brittle on out-of-distribution imagery, whereas Segment Anything Model 3 (SAM3) [Carion et al., 2025] integrates the text-to-region step via its Promptable Concept

Segmentation (PCS) head, which accepts short noun-phrase prompts such as `sky. vehicles. directly.` With image and video propagation, a single prompt can yield coherent masks across an entire survey. The relevant question is therefore not whether such masks appear plausible, but whether unmodified PCS prompting materially improves downstream 3D reconstruction.

We make three contributions: (i) *Automask*, a containerised semi-automatic masking service combining SAM3 with a queue-driven batch processor and a Vue.js editor, designed to integrate with existing pipelines (camera $\rightarrow$ Automask $\rightarrow$ COLMAP $\rightarrow$ Gaussian Splatting) and scale via Kubernetes/KEDA; (ii) an end-to-end evaluation on a 589-image uncrewed aerial vehicle (UAV) glacier survey, reporting peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), Learned Perceptual Image Patch Similarity (LPIPS), and reprojection error over ten runs; and (iii) an analysis of the residual failure modes that bound the reliability of a fully automatic single-prompt run, chiefly ice misclassified as sky under overcast lighting and thin over-bright structures being missed.

2 Design

2.1 Pipeline Overview

An input dataset $\mathcal{I} = \{I_1, \dots, I_N\}$ flows through three stages: (1) **Masking**, in which a SAM3 PCS prompt p and action $a \in \{\text{REMOVE}, \text{KEEP}, \text{HIGHLIGHT}\}$ yield I'_i ; (2) **SfM**, in which COLMAP recovers cameras and a sparse cloud; (3) **Gaussian Splatting**, in which Gaussians initialised from the sparse cloud are optimised against the masked images. The user interacts only with stage 1; SfM and splatting run unchanged from their public implementations (COLMAP 3.13 and `gsplat`).

2.2 SAM3 Masking

For each prompt p , the PCS head returns binary masks $\{m_{i,k}\}_k$ per matched concept instance. In batch mode, p is fixed and SAM3’s internal features enforce coarse spatial and temporal consistency across views, which is what makes a single-prompt, full-dataset workflow tractable. The action a composes masks into I'_i : REMOVE zeroes masked pixels (used when the prompt names unwanted content); KEEP is its dual (occasionally useful in interactive refinement); HIGHLIGHT alpha-blends a random colour over each region, biasing the SIFT detector toward those regions by raising local image entropy.

2.3 Prompt and Action

The configuration evaluated in this paper is background removal: $p = \text{sky. vehicles.}$ with $a = \text{REMOVE}$. It targets the dominant noise sources identified in Section 1, on the hypothesis that SfM is misled by spurious matches in these regions. The KEEP and HIGHLIGHT actions are available in the system and are used during interactive refinement, but are not evaluated here; this configuration is referred to below as “Masked”.

2.4 Reconstruction Backend

COLMAP uses default SIFT settings, untuned per dataset. The calibrated cameras and sparse cloud initialise Gaussian Splatting (PyTorch, `gsplat`, default optimiser settings). When REMOVE has been applied, masked pixels do not contribute to the photometric loss, so the splat optimisation benefits from cleaner supervision as well as cleaner poses.

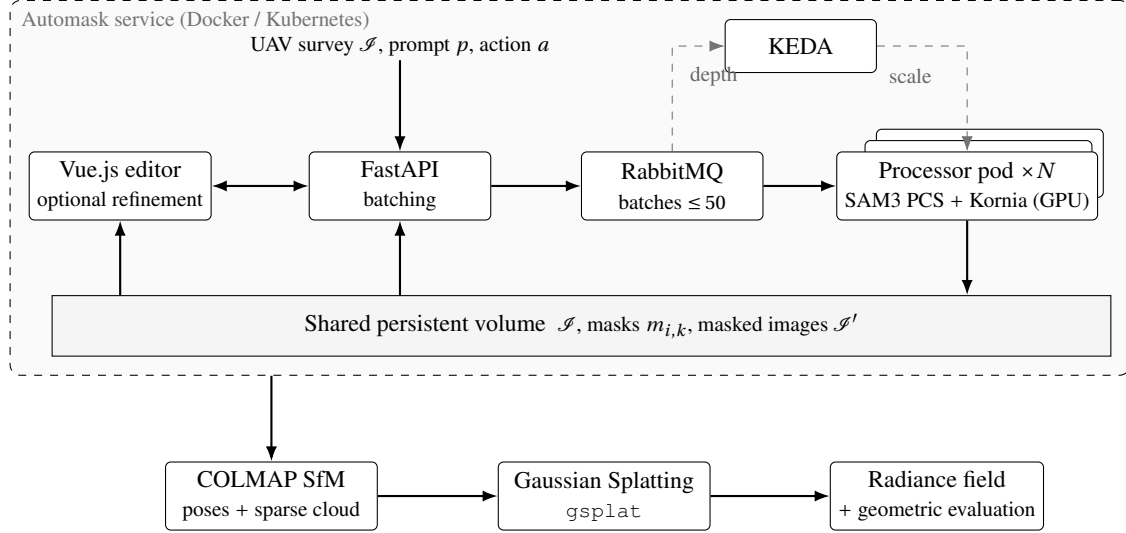


Figure 1: Automask architecture. Solid arrows carry image data, dashed arrows carry control signals. One prompt and action pair is applied to a whole survey: the API chunks the dataset into batches, RabbitMQ distributes them across processor pods, and KEDA adds pods when the backlog exceeds five batches. Masked images are written to the shared volume and consumed unmodified by the reconstruction backend; the editor is optional and operates on individual images.

2.5 Evaluation Metrics

We use PSNR (pixel fidelity, dB), SSIM (local luminance/contrast/structure), LPIPS (AlexNet-backbone perceptual similarity, sensitive to blur and ringing that PSNR/SSIM miss), and SfM reprojection error:

$$\text{RMSE} = \frac{1}{|\mathcal{O}|} \sum_{(i,j) \in \mathcal{O}} \|x_{ij} - \bar{x}_{ij}\|, \quad (1)$$

where $\mathcal{O}$ is the set of valid observations, x_{ij} the observed position of point i in image j , and $\bar{x}_{ij}$ its prediction.

3 Implementation

Automask separates three concerns (a low-latency interactive front-end, a stateful API, and a GPU-backed image processor) into Docker services orchestrated on Kubernetes, communicating through a RabbitMQ queue and a shared persistent volume (Figure 1). Only the masking processor is specific to the vision task; the remaining components exist so that one prompt can be applied to an entire survey without analyst involvement, which is the property the evaluation in Section 4 depends on.

3.1 Masking Processor

Each processor pod runs a consumer that hands batches to a `FileProcessor`, which processes files serially within the pod; parallelism is provided across pods rather than within them. An empty prompt invokes SAM3’s automatic mask generator; a non-empty prompt goes through PCS directly. Masks are composed with the action by a `FileEditor` built on Kornia [Riba et al., 2019], keeping the pipeline on the GPU and avoiding a host round trip per image. The three actions reduce to tensor operations: `HIGHLIGHT` alpha-blends a per-mask random colour at $\alpha = 0.5$; `KEEP` multiplies by the mask union; `REMOVE` multiplies by its complement. Per-image cost is dominated by SAM3 inference, so the processor is compute-bound and scales close to linearly in the number of pods.

3.2 Batch Execution and Scaling

The API chunks each request into batches of up to 50 files. Batching bounds per-message GPU memory, lets multiple pods work in parallel on one user’s request, and gives RabbitMQ enough granularity to redistribute work after pod failure. KEDA monitors queue depth and adds processor pods when the backlog exceeds five queued batches (~250 images). A persistent volume mounted in every pod holds input, intermediate, and output images, so newly scheduled pods resume without per-pod data migration. Processors are stateless except for the GPU-resident SAM3 checkpoint, which is baked into the image at build time to keep cold start deterministic; horizontal scaling therefore requires no coordination between pods.

3.3 Interactive Refinement

The Vue.js editor exposes four file states (Uploaded, Processing, Processed, Saved). From Processed, the user opens an editor displaying the source image, the SAM3 masks, and refinement tools; per-mask remove and keep operations execute client-side, and a brush tool handles cases the prompt misses, such as the thin structures discussed in Section 5. Edits are recorded in an undo/redo store that persists across page navigation. The editor is used for per-image correction only and plays no part in the batch results reported below, which use a single unedited prompt.

4 Evaluation

4.1 Setup

We use a 589-image UAV glacier survey (DJI FC330, pinhole with two radial distortion coefficients) from the UNAVCO TLS repository, downsampled to 2000×1500 pixels. Approximately one-third of frames include sky, a smaller fraction the survey helicopter, and the glacier surface ranges from crevassed sections to smooth shaded firn. All experiments run on a single workstation (NVIDIA RTX 5090, AMD Ryzen 7 9700X, 48 GB RAM, Ubuntu 24.04.3) using COLMAP 3.13 (default SIFT), `gsplat` on PyTorch 2.8/CUDA 12.9 (default optimiser), and the public SAM3 checkpoint with no fine-tuning.

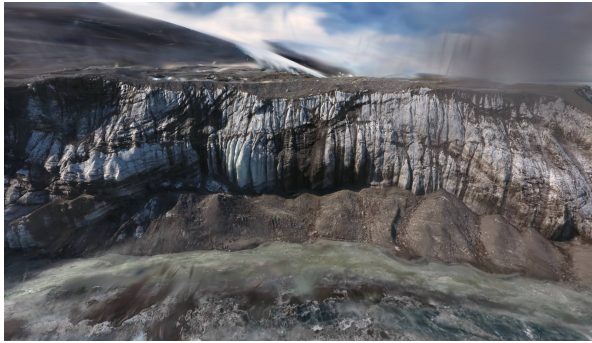
Both configurations are run through the full pipeline ten times to control for splat stochasticity. PSNR, SSIM, and LPIPS are computed on the original unmasked held-out views, so both configurations are scored over an identical pixel set; masking affects training supervision only.

4.2 Full-Pipeline Quantitative Results

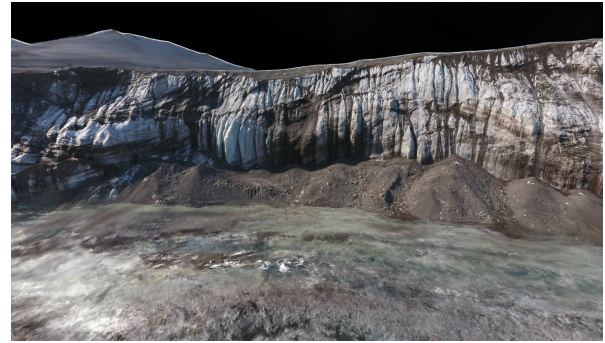
Table 1: Reconstruction quality on the 589-image UAV glacier dataset. Mean $\pm$ standard deviation over ten independent runs; best in bold.

Dataset	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	Reproj. $\downarrow$
Original	22.30 $\pm$ 0.18	0.723 $\pm$ 0.006	0.408 $\pm$ 0.014	0.556 $\pm$ 0.012
Masked	23.52 $\pm$ 0.21	0.788 $\pm$ 0.004	0.312 $\pm$ 0.015	0.506 $\pm$ 0.010

Masking improves every metric (Table 1): PSNR by ~ 1.2 dB, SSIM by 0.065, LPIPS by 0.096, reprojection error from 0.556 to 0.506 px. Masked variance is comparable to or smaller than Original’s on every metric, so the gain is not driven by a few favourable runs. Expressed as standardised effect sizes, the four improvements span $d \approx 4.5$ (reprojection error) to $d \approx 12.7$ (SSIM): modest in absolute terms, but large relative to the run-to-run variation of the splat optimisation. SSIM shows the smallest variance (consistent with its smoothness as a metric) and LPIPS the largest (deep-feature metrics are more sensitive to optimiser-trajectory changes).



(a) Unmasked reconstruction.



(b) Masked reconstruction.

Figure 2: Renderings from comparable viewpoints. Unmasked (left) shows blurred halos and ghosting from sky pixels and dynamic objects entering the splat optimisation; these are absent on the right.

4.3 Qualitative Results

Figure 2 shows two patterns the per-pixel metrics only partially capture: surviving sky pixels produce diffuse blue halos along the glacier silhouette (low-opacity Gaussians instantiated at indeterminate depth), and dynamic content produces ghosts where the same 3D location receives inconsistent supervision across frames. Both are largely absent from the masked reconstruction.

5 Discussion

Magnitude of the gains. The improvements are consistent but modest in absolute terms. Viewpoint diversity and inter-image overlap dominate reconstruction quality, and preprocessing cannot manufacture them; masking contributes at the margin by removing avoidable noise. What remains is bounded by the failure modes below, which suggests that further gains lie in improving model behaviour where it fails rather than in further prompt engineering.

Failure modes. Two glacier-specific failures recur. *Ice misclassified as sky*: foreground ice is occasionally classified as sky under overcast lighting, where the statistics of grey cloud overlap with shaded firn or wet ice; PCS does not distinguish visually similar concepts when one is absent from its training distribution. This motivates domain-specific fine-tuning of the kind exemplified by MedSAM [Ma et al., 2024]. *Thin over-bright structures missed*: helicopter rotors, guide wires, and antenna edges are at the segmentation model’s resolution threshold and are consistently missed by the `vehicles` prompt even when the helicopter body is segmented. Both are easily corrected with a brush stroke in the editor but constrain what a fully automatic single-prompt run can achieve.

Limitations and future work. The evaluation uses a single dataset and a single prompt and action configuration; alternative prompts, alternative actions, and their composition are not evaluated here, and the result may not transfer to substantially different content distributions. SfM and splat backends use default settings. The evaluation measures novel-view rendering quality and SfM residuals; the geometric accuracy of the reconstructed surface is not assessed against an independent reference. The most direct extension is glacier-specific fine-tuning of the PCS head, in the manner of MedSAM3 [Liu et al., 2025], for which a curated corpus with hand-verified ice/sky boundaries should suffice. Beyond preprocessing, the SAM3D Objects family suggests an end-to-end pipeline that segments and produces 3D structure with one model, subsuming COLMAP. A natural front-end extension is collaborative editing via a WebSocket that broadcasts edit events through the existing undo/redo store.

6 Related Work

Promptable segmentation. Segment Anything (SAM) [Kirillov et al., 2023] introduced promptable image segmentation as a foundation-model task, separating a heavy ViT image encoder (run once per image) from a lightweight prompt encoder and mask decoder so image embeddings can be reused across prompts. SAM3 [Carion et al., 2025] shifts the centre of gravity to Promptable Concept Segmentation, which accepts noun-phrase prompts and/or image exemplars and returns one mask per matching instance, using a multimodal LLM [Yin et al., 2024] for ambiguous prompts and a DETR-style detector [Carion et al., 2020] with a learned “presence token”. The PCS head subsumes the role of a separate open-vocabulary detector, the property Automask exploits to drop a stage relative to a chained text-detector + segmenter design.

Domain-adapted SAM variants. General-purpose SAM checkpoints fail on data far from natural images. MedSAM [Ma et al., 2024] adapts SAM to medical imaging via encoder downsampling, an FPN for multi-scale lesion segmentation, and CT/MRI fine-tuning. SAM-Adapter [Chen et al., 2023] freezes foundation weights and injects task-specific knowledge through small MLP adapters. We considered SAM-Adapter for ice/water boundaries but found no clean integration with PCS-style prompting; the results in Section 4 suggest fine-tuning is the appropriate next step.

Applications and SfM preprocessing. Existing SAM-based applications cluster into browser-based single-image demos (often using SlimSAM [Chen et al., 2024] or ONNX runtime, unsuitable for batch processing hundreds of images) and vertical clinical tools such as MedSAM-Lite. Automask occupies the middle: batch input with per-image interactive refinement for remote-sensing researchers. SfM preprocessing pipelines typically use fixed-class semantic networks or heuristic colour models for sky removal, both of which generalise poorly to glacial imagery in which overcast sky is hard to separate from snow. Foundation models accept the problem class as a prompt at inference time; the contribution here is therefore not a new architecture but a controlled empirical study of what such a model contributes inside an existing SfM + Gaussian Splatting pipeline.

Glacier remote sensing. Photogrammetric monitoring of glaciers has expanded with UAV adoption [Bhardwaj et al., 2016, Zhao et al., , Zhao et al., 2025]: consumer-grade platforms enable repeated surveys at lower cost than crewed aircraft or terrestrial laser scanning while resolving sub-metre features. Such imagery is dense, high-overlap, and rich in precisely the non-glacier content this work targets.

7 Conclusion

Automask uses SAM3’s PCS head to convert a single short text prompt into per-image masks across an entire glacier survey, improving downstream COLMAP + Gaussian Splatting reconstruction on all four metrics simultaneously across ten runs. The magnitude of the gains is constrained by source-survey viewpoint coverage and by glacier-specific failure modes the unmodified PCS head inherits from its training distribution, which makes fine-tuning rather than further prompt engineering the natural next question. The SAM3 checkpoint is the only stateful component of the image processor, so replacing it is a single configuration change.

Acknowledgments

This research was conducted with the financial support of Research Ireland under Grant Agreement No. 13/RC/2106_P2 at the ADAPT Centre at Trinity College Dublin. ADAPT, the Research Ireland Centre for AI-Driven Digital Content Technology, is funded through the Research Ireland Centres Programme. For the purpose of Open Access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission. This work is partly supported by China Scholarship Council (CSC) (Grant No. 202408320050).

References

- [Bhardwaj et al., 2016] Bhardwaj, A., Sam, L., Martín-Torres, F. J., Kumar, R., et al. (2016). Uavs as remote sensing platform in glaciology: Present applications and future prospects. *Remote sensing of environment*, 175:196–204.
- [Carion et al., 2025] Carion, N., Gustafson, L., Hu, Y.-T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K. V., Khedr, H., Huang, A., Lei, J., Ma, T., Guo, B., Kalla, A., Marks, M., Greer, J., Wang, M., Sun, P., Rädle, R., Afouras, T., Mavroudi, E., Xu, K., Wu, T.-H., Zhou, Y., Momeni, L., Hazra, R., Ding, S., Vaze, S., Porcher, F., Li, F., Li, S., Kamath, A., Cheng, H. K., Dollár, P., Ravi, N., Saenko, K., Zhang, P., and Feichtenhofer, C. (2025). Sam 3: Segment anything with concepts.
- [Carion et al., 2020] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers.
- [Chen et al., 2023] Chen, T., Zhu, L., Deng, C., Cao, R., Wang, Y., Zhang, S., Li, Z., Sun, L., Zang, Y., and Mao, P. (2023). Sam-adapter: Adapting segment anything in underperformed scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3367–3375.
- [Chen et al., 2024] Chen, Z., Fang, G., Ma, X., and Wang, X. (2024). Slimsam: 0.1
- [Geissler et al., 2021] Geissler, J., Mayer, C., Jubanski, J., Münzer, U., and Siegert, F. (2021). Analyzing glacier retreat and mass balances using aerial and uav photogrammetry in the ötztal alps, austria. *The Cryosphere*, 15(8):3699–3717.
- [Gindraux et al., 2017] Gindraux, S., Boesch, R., and Farinotti, D. (2017). Accuracy assessment of digital surface models from unmanned aerial vehicles’ imagery on glaciers. *Remote Sensing*, 9(2):186.
- [Kerbl et al., 2023] Kerbl, B., Kopanas, G., Leimkühler, T., and Drettakis, G. (2023). 3D gaussian splatting for real-time radiance field rendering.
- [Kirillov et al., 2023] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. (2023). Segment anything. *arXiv:2304.02643*.
- [Liu et al., 2025] Liu, A., Xue, R., Cao, X. R., Shen, Y., Lu, Y., Li, X., Chen, Q., and Chen, J. (2025). Medsam3: Delving into segment anything with medical concepts.
- [Liu et al., 2024] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., et al. (2024). Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pages 38–55. Springer.
- [Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.
- [Ma et al., 2024] Ma, J., He, Y., Li, F., Han, L., You, C., and Wang, B. (2024). Segment anything in medical images. *Nature Communications*, 15(1).
- [Riba et al., 2019] Riba, E., Mishkin, D., Ponsa, D., Rublee, E., and Bradski, G. (2019). Kornia: an open source differentiable computer vision library for pytorch.
- [Schönberger and Frahm, 2016] Schönberger, J. L. and Frahm, J.-M. (2016). Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Schönberger et al., 2016] Schönberger, J. L., Zheng, E., Pollefeys, M., and Frahm, J.-M. (2016). Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*.

- [Westoby et al., 2012] Westoby, M. J., Brasington, J., Glasser, N. F., Hambrey, M. J., and Reynolds, J. M. (2012). ‘structure-from-motion’ photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, 179:300–314.
- [Yin et al., 2024] Yin, S., Fu, C., Zhao, S., Li, K., Sun, X., Xu, T., and Chen, E. (2024). A survey on multimodal large language models. *National Science Review*, 11(12).
- [Zhao et al., 2025] Zhao, S., Liang, Y., Campbell, A., Pakrashi, A., and Dev, S. (2025). 3D glacier reconstruction in challenging environments: A comparative study of traditional and deep learning approaches. In *2025 18th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–6. IEEE.
- [Zhao et al.,] Zhao, S., Pakrashi, A., and Dev, S. Towards efficient glacier 3D gaussian splatting reconstruction with pairvpr-based image selection. *Available at SSRN 6481998*.

Low-Data RGB-to-NIR Translation with Augmentation and Pseudo-Label Learning

Md Yearat Hossain^{1,2,†}, Arpit Mirdha^{2,†}, Riju Das^{1,2}, Gavin McArdle², and Soumyabrata Dev^{1,2,3}

¹*The ADAPT Research Centre, Dublin, Ireland*

²*School of Computer Science, University College Dublin, Dublin, Ireland*

³*School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland*

[†]*These authors contributed equally to this work.*

Abstract

Near-infrared (NIR) imagery plays an important role in agricultural and remote sensing applications such as vegetation monitoring and environmental analysis. The multispectral sensors required to acquire it carry higher cost and integration overhead than standard RGB cameras, which motivates methods that infer NIR information directly from RGB imagery. Existing RGB-to-NIR translation approaches focus primarily on architecture and methodology, but their performance is constrained by the scarcity of paired training data, and the question of how to achieve reliable results when paired data is limited has received little direct attention. This work explores two complementary strategies for addressing this constraint: data augmentation and pseudo-label semi-supervised learning. Using the Canola UAV dataset and the SEN12MS satellite dataset, two architectures of substantially different capacity, U-Lite and MIRNet-v2, were trained under five augmentation configurations and a semi-supervised pipeline that treats model predictions on unlabelled RGB images as pseudo-labels. Augmentation consistently improved performance across both datasets and architectures, with the best supervised result reaching 29.74 dB PSNR on Canola. Semi-supervised learning produced smaller gains, improving performance on Canola when combined with augmentation but failing to improve over the augmented baseline on SEN12MS. The findings indicate that augmentation is the more dependable of the two strategies in the low-data regime, while the effectiveness of pseudo-labelling depends on whether the supervised baseline has learned the target dataset well enough for its predictions to act as useful training signal.

Keywords: Infrared Imaging, RGB-to-NIR, Image Translation, Image Augmentation

1 Introduction

Near-infrared (NIR) imagery plays an important role in remote sensing applications such as vegetation monitoring and environmental analysis [Gutman et al., 2021, Vescovo et al., 2012, Badgley et al., 2017]. Unlike visible red-green-blue (RGB) imagery, NIR captures material and structural properties that are not directly perceived in the visible spectrum, including indicators of plant health and surface reflectance [Gutman et al., 2021, Badgley et al., 2017]. However, dedicated NIR and multispectral sensors carry higher cost and integration overhead than standard RGB cameras, and are often unavailable on consumer-grade or resource-constrained platforms. This motivates methods that infer NIR information directly from widely available RGB imagery, enabling NIR-like benefits without additional hardware [Aslahishahri et al., 2021, Picon et al., 2022, Jin et al., 2025].

RGB-to-NIR translation can be framed as an image-to-image regression problem, in which a model learns a mapping from visible bands to a continuous NIR channel [Aslahishahri et al., 2021, Isola et al., 2017]. Recent work in deep learning, particularly using convolutional and encoder-decoder architectures, has shown strong performance on related image translation tasks. Learning an accurate RGB-to-NIR mapping nonetheless remains challenging. The relationship between visible and near-infrared reflectance is highly nonlinear and scene-dependent, and NIR labels typically require specialised sensors and careful calibration [Aslahishahri et al., 2021].

Available RGB-NIR datasets are often limited in size, geographic coverage, or diversity of land cover conditions [Aslahishahri et al., 2021].

Two limitations follow from these constraints. Models trained on small, homogeneous datasets tend to overfit and fail to generalise to new locations, seasons, or sensor configurations [Aslahishahri et al., 2021, Bansal et al., 2022]. Higher-capacity architectures that could exploit richer context are difficult to apply effectively in this regime, since they typically require large quantities of annotated data to avoid poor generalisation [Aslahishahri et al., 2021, Jin et al., 2025].

Image augmentation is a widely adopted strategy for mitigating data scarcity in deep learning. It increases the effective size and diversity of a dataset by applying label-preserving geometric and photometric transformations to existing samples, enabling models to generalise better from limited data [Kumar et al., 2024]. Complementarily, pseudo-labelling provides a semi-supervised learning mechanism that leverages a small set of annotated data to generate approximate labels for unlabelled samples [van Engelen and Hoos, 2020]. In this approach, a model trained on labelled data is used to infer targets for unseen inputs, which are then incorporated into the training process to improve performance and generalisation [Su et al., 2024]. While both techniques have demonstrated effectiveness across various vision tasks, their impact on RGB-to-NIR translation remains underexplored. This is particularly relevant given the scarcity of paired RGB-NIR datasets, where such methods could play a crucial role in improving model robustness and scalability.

Motivated by these challenges, this work investigates two complementary strategies for RGB-to-NIR translation under data-scarce conditions: data augmentation and pseudo-label-based semi-supervised learning. Augmentation expands the effective diversity of the labelled training set through geometric and photometric transformations applied to RGB-NIR pairs. Semi-supervised learning extends the training signal to unlabelled RGB images by treating predictions from a supervised baseline as approximate ground truth labels. Both strategies are evaluated on two architectures of substantially different capacity, U-Lite and MIRNet-v2, and across two remote sensing datasets, the canola subset of the Aslahishahri et al. [Aslahishahri et al., 2021] aerial dataset and the SEN12MS satellite dataset [Schmitt et al., 2019]. The combination allows the experiments to test whether observed gains hold across architectures and across remote sensing domains. All codes and datasets are available at: <https://github.com/yearat/RGB2NIR-Low-Data-Augmentation>

2 Related Works

Image-to-image (I2I) translation describes the class of problems in which a model learns to convert one representation of a scene into another, for instance from an edge map to a photograph, or from a visible-spectrum image to a synthetic near-infrared output, using a unified learning framework rather than task-specific engineering [Isola et al., 2017]. Rather than generating entirely new content, I2I methods condition their output on an observed input, enabling applications such as style transfer, image colourisation, modality conversion, and semantic segmentation [Isola et al., 2017, Minaee et al., 2021].

In remote sensing, I2I translation is relevant because aerial and satellite imagery is multi-spectral, with different spectral bands capturing complementary physical properties of the observed scene [Aslahishahri et al., 2021, Gutman et al., 2021]. Near-infrared (NIR) imagery encodes information related to vegetation health and surface reflectance that is not directly observable in the visible spectrum [Gutman et al., 2021, Badgley et al., 2017, Vescovo et al., 2012]. Translating between spectral domains therefore has practical value in precision agriculture, environmental monitoring, and land-use analysis [Gutman et al., 2021, Badgley et al., 2017] and synthesising NIR from RGB imagery reduces the instrumentation burden of multispectral data collection [Aslahishahri et al., 2021].

Due to the cost and complexity of dedicated NIR imaging systems, alongside the widespread availability of RGB cameras, recent research has increasingly focused on deep learning-based RGB-to-NIR translation. These approaches leverage image-to-image (I2I) models to synthesise NIR imagery directly from RGB inputs [Illarionova et al., 2021, Aslahishahri et al., 2021, Hossain et al., 2025]. Given that vegetation health indicators are primarily derived from NIR data, many studies utilise agricultural datasets and evaluate the effectiveness of generated NIR images through downstream metrics such as the Normalized Difference Vegetation Index (NDVI) and the Normalized Difference Red Edge (NDRE) [Zhao et al., 2024, Davidson et al., 2022,

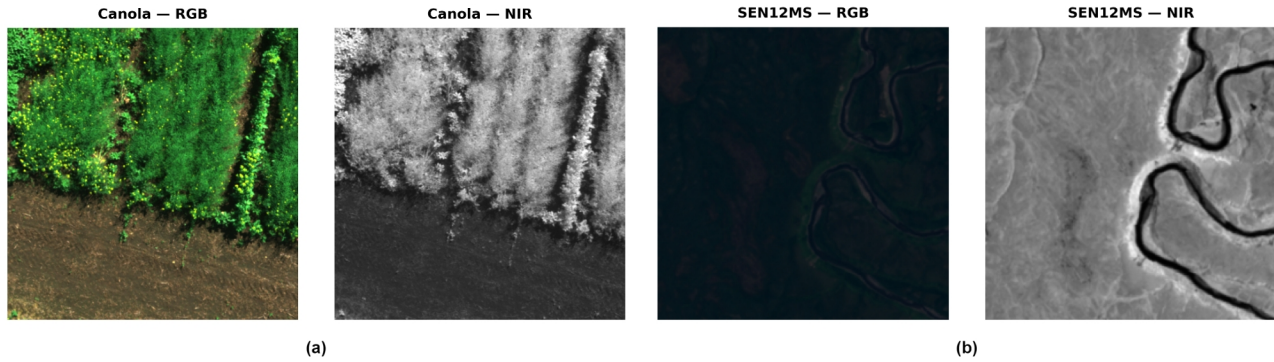


Figure 1: Representative paired RGB–NIR samples from the datasets used in this study. (a) Canola dataset showing UAV aerial imagery of crop canopy. (b) SEN12MS dataset showing Sentinel-2 satellite imagery. In both cases, the left image represents the RGB input and the right image represents the corresponding NIR target.

Aslahishahri et al., 2021]. Despite their effectiveness, these methods typically rely on pixel-wise paired RGB–NIR datasets, the collection of which is both technically demanding and resource-intensive. Such datasets require specialised imaging systems, careful sensor alignment, and radiometric calibration to ensure consistency and usability [Aslahishahri et al., 2021, Hossain and Dev, 2026].

Deep learning models generally achieve improved accuracy and generalisation when trained on large-scale datasets [Krizhevsky et al., 2012]. However, obtaining sufficient labelled data remains a significant challenge in many domains, including RGB-to-NIR translation. Addressing data scarcity is therefore an active area of research in machine learning [Alzubaidi et al., 2023]. One widely adopted approach is data augmentation, which involves generating additional training samples through label-preserving transformations such as geometric and photometric variations [Kumar et al., 2024]. Image augmentation has proven effective in improving model robustness and generalisation, particularly in low-data regimes [Rebuffi et al., 2021, Xu et al., 2023].

Another promising strategy is pseudo-label-based semi-supervised learning. In this approach, a model is first trained on a limited set of labelled data and then used to generate approximate labels for unlabelled samples [van Engelen and Hoos, 2020, Alzubaidi et al., 2023]. These pseudo-labelled data are subsequently incorporated into the training process, enabling the model to learn from a larger effective dataset and achieve improved performance and generalisation [Su et al., 2024].

Despite the data-intensive nature of RGB-to-NIR translation and the challenges associated with dataset collection, existing work has largely overlooked the problem of improving performance under limited data conditions. To address this gap, this work investigates the effectiveness of image augmentation and pseudo-label-based semi-supervised learning for enhancing RGB-to-NIR translation in data-scarce scenarios.

3 Methodology

3.1 Datasets

Two datasets are used in this work: a UAV-based agricultural dataset and a satellite remote sensing dataset, enabling evaluation of the proposed methods across different domains and testing their robustness and generalisation capabilities under varying imaging conditions and scene characteristics. Representative RGB–NIR image pairs from both datasets are shown in Figure 1.

The first dataset is the canola subset of the paired RGB–NIR aerial image dataset introduced by Aslahishahri et al. [Aslahishahri et al., 2021], captured using a UAV-mounted multispectral sensor over agricultural fields in Canada. The second dataset is derived from the SEN12MS dataset proposed by Schmitt et al. [Schmitt et al., 2019], using Sentinel-2 satellite imagery. For both datasets, 360 paired RGB–NIR samples are used in this work, with 300 samples used for training and 60 reserved for testing. All images are cropped to 256×256 patches prior to training. While the Canola dataset contains UAV-based crop imagery, SEN12MS provides large-scale satellite

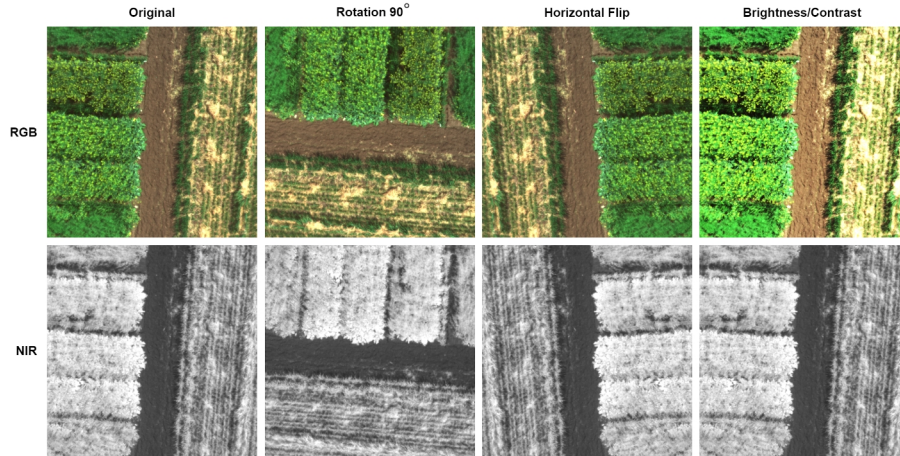


Figure 2: Examples of augmentation methods used in this study. From left to right: original image, 90° rotation, horizontal flip, brightness and contrast adjustment. Geometric augmentations are applied pairwise to both the RGB input and corresponding NIR target, while photometric augmentations are applied only to the RGB input and the NIR target remains unchanged.

imagery across diverse land-cover conditions, enabling evaluation across substantially different remote sensing domains.

3.2 Image Augmentation Pipeline

Five augmentation configurations are evaluated to analyse the effect of different augmentation strategies under limited-data conditions. The applied transformations include horizontal flip, vertical flip, rotations of 90°, 180°, and 270°, brightness jitter, and contrast jitter, with representative examples shown in Figure 2. Brightness and contrast augmentations are applied only to the input RGB images while the target NIR images remain unchanged, whereas geometric transformations are applied consistently to both RGB and NIR pairs to preserve spatial alignment. Each configuration is trained and evaluated independently using the same 300 training images, and all augmented samples are generated prior to training and kept fixed across epochs to ensure reproducibility.

3.3 Semi-Supervised Learning Pipeline

For the semi-supervised learning (SSL) experiments, the original 300 training pairs are divided into 200 labelled RGB–NIR pairs and 100 unlabelled RGB images, while the remaining 60 paired samples are reserved for evaluation and checkpoint selection. The split is generated once using a fixed random seed and kept consistent across all SSL experiments. A supervised baseline model is first trained on the 200 labelled pairs and then used to generate pseudo-labelled NIR outputs for the 100 unlabelled RGB images. Pseudo-labels are generated from the original unaugmented RGB inputs without confidence filtering, and all pseudo-labelled samples are retained for training. Two SSL settings are evaluated: without augmentation and with augmentation. In the no-augmentation setting, the mixed dataset contains 200 real labelled pairs and 100 pseudo-labelled pairs, giving 300 training samples in total. In the augmentation setting, all seven augmentation transforms are applied, expanding the 200 labelled pairs to 1,600 samples and the 100 pseudo-labelled pairs to 800 samples, resulting in 2,400 training samples overall. For each setting, two incorporation strategies are investigated: retraining a new model from scratch on the mixed dataset for 150 epochs using a learning rate of 2×10^{-4} , and fine-tuning from the baseline weights for 50 epochs using a reduced learning rate of 2×10^{-5} .

3.4 Models

Two architectures of substantially different capacity are evaluated. The first is a modified U-Lite [Dinh et al., 2023], a lightweight U-Net-based architecture using axial depthwise convolutions for computational efficiency. U-Net is chosen as it is one of the most widely used image-to-image (I2I) models and forms the basis of many subsequent architectures. The channel configuration is widened from [16, 32, 64, 128, 256, 512] to [20, 40, 80, 160, 320, 640], resulting in approximately 1.36 million trainable parameters. The second model is MIRNet-v2 [Zamir et al., 2023], a higher-capacity multi-scale restoration architecture that has achieved state-of-the-art performance across various image restoration tasks. Both models are trained using a reconstruction-only objective without adversarial learning, which is often unstable under limited paired data conditions [Goodfellow et al., 2014].

3.5 Implementation Details

All experiments are conducted on an NVIDIA GeForce RTX 3060 12GB GPU using mixed precision training to reduce memory usage and accelerate training. A global random seed of 42 is used across all experiments to ensure reproducibility. Models are trained using the Adam optimiser with a learning rate of 2×10^{-4} , a batch size of 4, and a CosineAnnealingLR scheduler over 150 epochs. Gradient clipping with a maximum norm of 1.0 is applied to improve training stability. Model performance is evaluated using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM), two widely used reference-based image quality assessment metrics for evaluating image reconstruction and restoration quality.

4 Results

4.1 Supervised Learning with Augmentation

Table 1 presents PSNR and SSIM results for both models across five augmentation strategies on Canola and SEN12MS datasets.

Model	Augmentation	Canola		SEN12MS	
		PSNR	SSIM	PSNR	SSIM
MIRNet-v2	Baseline	28.62	0.9280	28.14	0.8245
	Flips	29.49	0.9394	29.83	0.8495
	Rotations	29.74	0.9414	29.28	0.8435
	Brightness	29.44	0.9372	28.99	0.8380
	Contrast	29.44	0.9372	28.99	0.8380
U-Lite	Baseline	27.95	0.9103	27.10	0.7728
	Flips	28.34	0.9222	27.92	0.8048
	Rotations	28.36	0.9230	27.96	0.8093
	Brightness	28.30	0.9200	27.73	0.7977
	Contrast	28.30	0.9200	27.73	0.7977

Table 1: Supervised learning results across augmentation strategies and datasets.

All augmentation strategies improve upon the no-augmentation baseline for both models and datasets. Geometric augmentations (flips and rotations) yield larger gains than photometric jitter: on Canola, rotations achieve 29.74 dB (MIRNet-v2, +1.12 dB), while brightness and contrast both reach 29.44 dB (+0.82 dB). A similar pattern appears on SEN12MS: flips yield 29.83 dB (+1.69 dB) while photometric augmentation yields 28.99 dB (+0.85 dB). MIRNet-v2 outperforms U-Lite across all conditions, with the gap widening on SEN12MS (1.87 dB at best) compared to Canola (1.38 dB).

4.2 Semi-Supervised Learning Results

Table 2 reports PSNR and SSIM for both models across all SSL conditions on both datasets.

Model	Aug.	SSL	Canola		SEN12MS	
			PSNR	SSIM	PSNR	SSIM
MIRNet-v2	None	Baseline	28.23	0.9235	27.39	0.8099
	None	Retrain	28.15	0.9244	27.62	0.8151
	None	Finetune	28.26	0.9239	27.41	0.8106
	Aug all	Baseline	29.42	0.9392	29.71	0.8492
	Aug all	Retrain	29.54	0.9418	28.65	0.8431
	Aug all	Finetune	29.53	0.9397	29.42	0.8493
U-Lite	None	Baseline	27.46	0.9013	26.50	0.7616
	None	Retrain	27.65	0.9056	26.80	0.7682
	None	Finetune	27.45	0.9016	26.51	0.7616
	Aug all	Baseline	27.87	0.9228	27.49	0.8043
	Aug all	Retrain	28.27	0.9181	26.95	0.8109
	Aug all	Finetune	27.97	0.9129	27.58	0.8067

Table 2: Semi-supervised learning results. PSNR is reported in dB. Best PSNR and SSIM values for each model-dataset combination are highlighted in **bold**.

In the no-augmentation SSL condition, pseudo-labelling produces only marginal changes, with small gains in some settings and negligible decreases in others. MIRNet-v2 on Canola improves from 28.23 dB (baseline) to 28.26 dB (finetune, +0.03 dB); on SEN12MS, retrain reaches 27.62 dB (+0.23 dB). U-Lite shows a similar pattern, with retrain gains of +0.19 dB on Canola and +0.30 dB on SEN12MS. Augmented SSL yields larger gains on Canola: MIRNet-v2 achieves 29.54 dB (retrain, +1.31 dB from no-augmentation baseline of 28.23 dB), and U-Lite achieves 28.27 dB (retrain, +0.81 dB). However, on SEN12MS, augmented SSL does not consistently improve over the augmented baseline. MIRNet-v2 reaches 29.71 dB at the augmented baseline but drops to 28.65 dB under retrain and 29.42 dB under finetune. U-Lite exhibits the same pattern, with retrain (26.95 dB) falling below the augmented baseline (27.49 dB), while finetune (27.58 dB) slightly exceeds it.

5 Discussion

The results demonstrate that data augmentation consistently improves RGB-to-NIR translation performance across both architectures and datasets, with geometric augmentations providing the strongest gains. MIRNet-v2 consistently outperforms the lightweight U-Lite model, particularly on the more complex SEN12MS dataset. Semi-supervised learning without augmentation provides only marginal improvements, suggesting that pseudo-label quality is limited when trained on small labelled datasets. Since pseudo-labels are generated by the baseline model itself, prediction errors can be propagated back into training, limiting the amount of new information introduced by the additional data. When combined with augmentation, SSL improves performance on the more homogeneous Canola dataset, but produces mixed results on the more diverse SEN12MS dataset, where the greater variability of the data likely increases pseudo-label noise, reducing the effectiveness of retraining. Overall, the findings indicate that augmentation is a reliable strategy for improving RGB-to-NIR translation under limited-data conditions, while the effectiveness of pseudo-label-based SSL depends strongly on dataset complexity and pseudo-label quality.

6 Conclusion

This work investigated data augmentation and pseudo-label-based semi-supervised learning for RGB-to-NIR translation under limited-data conditions across two architectures and two remote sensing datasets. The results show that data augmentation consistently improves performance, particularly with geometric transformations, while the effectiveness of pseudo-label-based SSL depends strongly on dataset complexity and pseudo-label

quality. Overall, augmentation proved to be a reliable strategy for low-data RGB-to-NIR translation, whereas SSL produced mixed results on more diverse datasets. Future work will explore more advanced SSL methods and evaluate the generated NIR imagery on downstream remote sensing tasks such as NDVI analysis.

Acknowledgments

This research was conducted with the financial support of Research Ireland under Grant Agreement No. 13/RC/2106_P2 at the ADAPT Centre at Trinity College Dublin. ADAPT, the Research Ireland Centre for AI-Driven Digital Content Technology, is funded through the Research Ireland Centres Programme.

References

- [Alzubaidi et al., 2023] Alzubaidi, L., Bai, J., Al-Sabaawi, A., Santamaría, J., Albahri, A. S., Al-Dabbagh, B. S. N., Fadhel, M. A., Manoufali, M., Zhang, J., Al-Timemy, A. H., et al. (2023). A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications. *Journal of Big Data*, 10(1):46.
- [Aslahishahri et al., 2021] Aslahishahri, M., Stanley, K. G., Duddu, H., Shirliffe, S., Vail, S., Bett, K., Pozniak, C., and Stavness, I. (2021). From RGB to NIR: Predicting of near infrared reflectance from visible spectrum aerial images of crops. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 1312–1322.
- [Badgley et al., 2017] Badgley, G., Field, C. B., and Berry, J. A. (2017). Canopy near-infrared reflectance and terrestrial photosynthesis. *Science Advances*, 3(3):e1602244.
- [Bansal et al., 2022] Bansal, A., Sharma, R., and Kathuria, M. (2022). A systematic review on data scarcity problem in deep learning: Solution and applications. *ACM Computing Surveys*, 54(10s):1–29.
- [Davidson et al., 2022] Davidson, C., Jaganathan, V., Sivakumar, A. N., Czarnecki, J. M. P., and Chowdhary, G. (2022). NDVI/NDRE prediction from standard RGB aerial imagery using deep learning. *Computers and Electronics in Agriculture*, 203:107396.
- [Dinh et al., 2023] Dinh, B.-D., Nguyen, T.-T., Tran, T.-T., and Pham, V.-T. (2023). 1M parameters are enough? a lightweight CNN-based model for medical image segmentation. In *2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1279–1284.
- [Goodfellow et al., 2014] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27.
- [Gutman et al., 2021] Gutman, G., Skakun, S., and Gitelson, A. (2021). Revisiting the use of red and near-infrared reflectances in vegetation studies and numerical climate models. *Science of Remote Sensing*, 4:100025.
- [Hossain and Dev, 2026] Hossain, M. Y. and Dev, S. (2026). End-to-end fusion of rgb and low-resolution nir for enhanced crop-specific nir generation. *Computers and Electronics in Agriculture*, 248:111582.
- [Hossain et al., 2025] Hossain, M. Y., Rakib, M. M. H., Pakrashi, A., and Dev, S. (2025). Towards efficient spectral translation: A lightweight image-to-image model for rgb-to-nir conversion. In *2025 18th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–6.

- [Illarionova et al., 2021] Illarionova, S., Shadrin, D., Trekin, A., Ignatiev, V., and Oseledets, I. (2021). Generation of the nir spectral band for satellite images with convolutional neural networks. *Sensors*, 21(16):5646.
- [Isola et al., 2017] Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976.
- [Jin et al., 2025] Jin, Y., Park, I., Song, H., Ju, H., Nalcakan, Y., and Kim, S. (2025). Pix2Next: Leveraging vision foundation models for RGB to NIR image translation. *Technologies*, 13(4):154.
- [Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.
- [Kumar et al., 2024] Kumar, T., Brennan, R., Mileo, A., and Bendeche, M. (2024). Image data augmentation approaches: A comprehensive survey and future directions. *IEEE Access*, 12:187536–187571.
- [Minaee et al., 2021] Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., and Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(7):1940–1965.
- [Picon et al., 2022] Picon, A., Bereciartua-Perez, A., Eguskiza, I., Romero-Rodriguez, J., Jimenez-Ruiz, C. J., Eggers, T., Klukas, C., and Navarra-Mestre, R. (2022). Deep convolutional neural network for damaged vegetation segmentation from RGB images based on virtual NIR-channel estimation. *Artificial Intelligence in Agriculture*, 6:199–210.
- [Rebuffi et al., 2021] Rebuffi, S.-A., Goyal, S., Calian, D., Stimberg, F., Wiles, O., and Mann, T. (2021). Data augmentation can improve robustness. In *Proceedings of the 35th International Conference on Neural Information Processing Systems, NIPS '21*, Red Hook, NY, USA. Curran Associates Inc.
- [Schmitt et al., 2019] Schmitt, M., Hughes, L. H., Qiu, C., and Zhu, X. X. (2019). SEN12MS – a curated dataset of georeferenced multi-spectral Sentinel-1/2 imagery for deep learning and data fusion. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume IV-2/W7, pages 153–160.
- [Su et al., 2024] Su, J., Luo, Z., Lian, S., Lin, D., and Li, S. (2024). Mutual learning with reliable pseudo label for semi-supervised medical image segmentation. *Medical Image Analysis*, 94:103111.
- [van Engelen and Hoos, 2020] van Engelen, J. E. and Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440.
- [Vescovo et al., 2012] Vescovo, L., Wohlfahrt, G., Balzarolo, M., Pilloni, S., Sottocornola, M., Rodeghiero, M., and Gianelle, D. (2012). New spectral vegetation indices based on the near-infrared shoulder wavelengths for remote detection of grassland phytomass. *International Journal of Remote Sensing*, 33(7):2179–2195.
- [Xu et al., 2023] Xu, M., Yoon, S., Fuentes, A., and Park, D. S. (2023). A comprehensive survey of image augmentation techniques for deep learning. *Pattern Recognition*, 137:109347.
- [Zamir et al., 2023] Zamir, S. W., Arora, A., Khan, S., Hayat, M., Khan, F. S., Yang, M.-H., and Shao, L. (2023). Learning enriched features for fast image restoration and enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1934–1948.
- [Zhao et al., 2024] Zhao, G., He, Y., Wang, Z., Wu, H., and Cheng, L. (2024). Generation of NIR spectral band from RGB image with wavelet domain spectral extrapolation generative adversarial network. *Computers and Electronics in Agriculture*, 227:109461.

Geo-referenced small object detection in oblique aerial imagery

Paolo Moloney¹, Aidan Magee¹, Daire Walsh¹, Tim McCarthy¹, and John McDonald²

¹*National Centre for Geocomputation, Maynooth University, Co. Kildare Ireland*

²*Department of Computer Science, Maynooth University, Co. Kildare, Ireland*

Abstract

Unmanned Aerial Vehicles (UAVs) are increasingly deployed in emergency response operations, resulting in growing volume and complexity of aerial video which creates a critical situational awareness bottleneck for incident commanders. We present an end-to-end Geospatial-aware ML framework that fuses computer vision with terrain-aware geospatial processing to automatically detect, localise, and visualise objects of interest from drone-borne video streams. The system integrates a scalable ML inference backend, a terrain-aware geo-positioning pipeline, and a suite of operator-focused visualisation methods, transforming continuous aerial video into structured, geo-referenced intelligence that incident commanders can act upon without reviewing raw footage. Additionally, we carry out a preliminary performance evaluation stratified by sensor-to-target distance on a small held-out dataset, to characterise the operational envelope of the system in actionable terms.

Keywords: UAV, aerial object detection, geospatial AI, emergency response, situational awareness

1 Introduction

Time-constrained emergency response operations — wildfires, search-and-rescue, flood assessment — increasingly rely on drone-borne imagery to build situational awareness over terrain that is dynamic, potentially hazardous, and often unfamiliar to first responders. While UAVs provide continuous aerial surveillance at low cost, their intelligence value is largely unrealised in practice: operators must manually scan hours of video to locate personnel, vehicles, or structural hazards, creating a critical information bottleneck at exactly the moment when rapid decisions are most consequential.

Several compounding factors make this a hard technical problem: drone-generated data volumes exceed what operators can parse in real time, and existing computer vision systems typically lack the geospatial integration incident commanders need for map- and GIS-based workflows.

This paper presents an end-to-end framework addressing these challenges through three core pillars - Operations, ML & Geospatial Processing, and Visualisation. The primary contribution of the framework is the integration of established components into a deployable pipeline, together with an evaluation methodology framed in operationally meaningful terms.

2 Related Work

Oblique aerial imagery is preferred over nadir capture for Search and Rescue (SAR) as an oblique frame covers a substantially larger ground area, reducing survey passes required. However it presents distinct detection challenges: altitude leads to reduced pixel footprints for distant objects, and oblique viewing angles introduce extreme scale variation across and within frames, which are not present in ground-level or nadir imagery.

The VisDrone dataset [Zhu et al., 2021] has become a popular benchmark for this domain, with approximately two thirds of objects falling under the COCO small category (bounding boxes $< 32^2$ px).

[Akyon et al., 2022] proposes a generic framework for small object detection based on partitioning images into overlapping patches at both the fine-tuning and inference stages, increasing the relative pixel coverage of small objects within each patch. During Slicing Aided Fine-tuning (SF), training data is augmented with resized crops alongside the original images, preserving sensitivity across object scales. During Slicing Aided Hyper Inference (SAHI), the same slicing strategy is applied to query images, with slice predictions merged back into the original coordinate space via Non-Maximum Merging (NMM). They find that existing models can benefit from SAHI without additional fine-tuning, but gains can be maximised by combining with SF.

Converting a 2D detection in the image into a real-world ground location requires resolving where the camera's line of sight to that pixel intersects the terrain. The dominant approach is to cast a ray from the camera's optical centre using attitude and gimbal telemetry, then intersect with a Digital Elevation Model (DEM) to recover a ground coordinate. This is validated by [Paulin et al., 2024] in real world SAR scenarios, and by [Rossi et al., 2025] in thermal hotspot geolocation in wildfire contexts.

3 Methodology

The framework is structured around three pillars - Operations, ML and Geospatial Processing, and Visualisation. Figure 1 gives a high-level system overview.

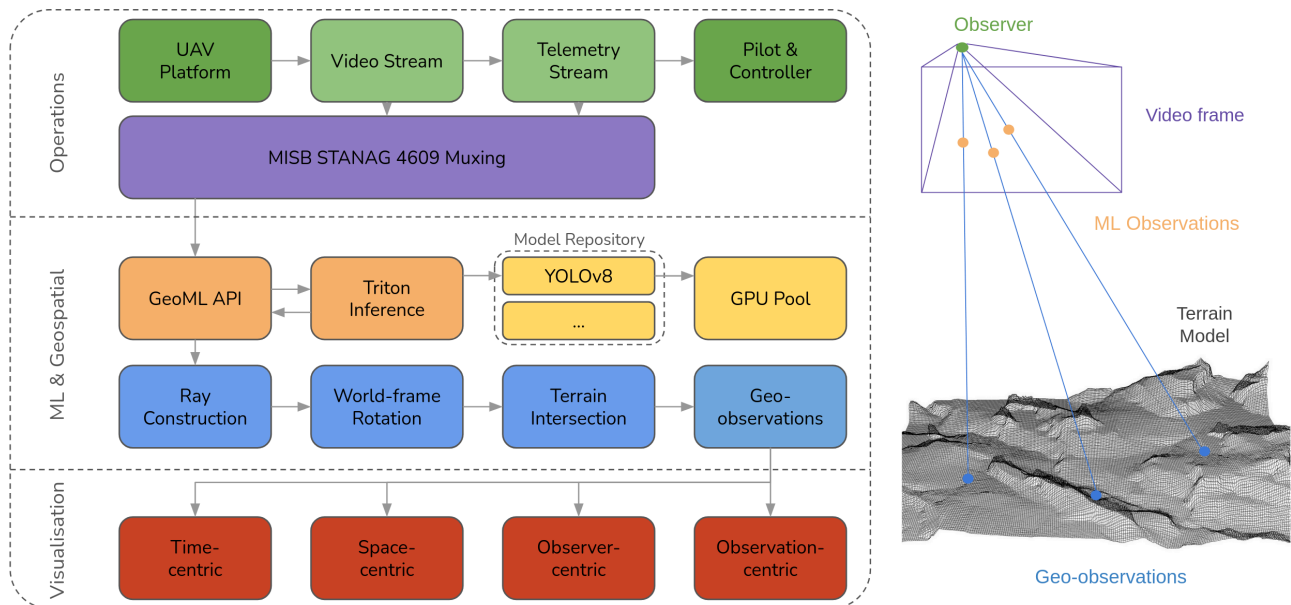


Figure 1: End-to-end GeoML pipeline. Video and telemetry are multiplexed via MISB STANAG 4609 KLV encoding before ingestion by the GeoML FastAPI and NVIDIA Triton for inference. Detections are passed to the geo-positioning pipeline and geo-observations are displayed by four visualisation modes designed with incident response in mind.

3.1 Operations

Tight video-telemetry synchronisation is essential for accurate geolocation. We achieve this by encoding video and telemetry as a single multiplexed stream, following established NATO and defence-imagery standards for full-motion video (FMV), rather than storing them as separate, independently-timestamped files. For this assessment, video and telemetry data were captured using a DJI Matrice 4E with the wide camera lens, then preliminarily multiplexed following NATO FMV Standardisation Agreement (STANAG) 4609, and Motion Imagery Standards Board (MISB) Standard 0601. Video and telemetry synchronisation quality was then as-

essed and updated globally with position and attitude offsets before re-encoding as Key-Length-Value (KLV) metadata within a Video Transport Stream (.ts) format.

The GeoML API is implemented using the FastAPI framework, exposing an interface for inference requests, model management operations, and downstream integration with in-house or third-party visualisation platforms. Inference is managed by an NVIDIA Triton Inference Server, which executes YOLOv8 models concurrently across available GPU resources. Triton’s model repository enables hot-swapping of model versions and independent assignment of model instances to GPUs, enabling multi-mission/-drone deployments. The entire stack is containerised with Docker, enabling horizontal scaling across available resources.

3.2 ML and Geospatial Processing

3.2.1 ML Pipeline

Object detection is performed by YOLOv8-x [Jocher et al., 2023], targeting two operationally relevant classes: *human* and *vehicle*. The extra-large variant was selected to maximise accuracy within the real-time throughput budget afforded by the Triton backend. At operational altitudes, subjects occupy only a small fraction of the image plane, causing standard full-resolution inference to miss detections that fall below the model’s effective receptive field. We apply Slicing Aided Fine-Tuning (SF) and Slicing Aided Hyper Inference (SAHI) [Akyon et al., 2022] to tile each input frame into overlapping sub-windows, run detection independently on each tile, and re-assemble results with class-aware Non-maximum Merging (NMM). Slices are batched before submission to Triton, recovering some GPU utilisation efficiency that would otherwise be lost to the smaller per-tile input size and duplicated processing of overlapping regions.

Both models are trained identically except for the use of sliced data augmentation, isolating the effect of the SF strategy described above; the following details are provided for reproducibility. Two YOLOv8x models are fine-tuned from the same COCO-pretrained checkpoint (Ultralytics v8.4.0) on the VisDrone-DET training set [Zhu et al., 2021]. The baseline model (YOLOv8-x) is trained on full-resolution frames comprising 7,944 training images and 448 validation images, containing 108,794 human and 237,531 vehicle annotations. The SF model’s (YOLOv8-x-SF) training data is augmented with sliced patches: full-resolution training images are partitioned into 640×640 pixel patches with 20% horizontal and vertical overlap, yielding 61,974 training patches and 2,261 validation patches. The increased annotation counts in the slice-augmented dataset (227,238 human; 501,118 vehicle) reflect object duplication across overlapping patch boundaries. Both models are trained for 100 epochs, batch size 32, SGD ($lr = 0.01$, momentum = 0.9). Default Ultralytics YOLOv8 data augmentation is applied throughout, with mosaic, scale, and perspective augmentation disabled.

3.2.2 Geospatial Pipeline

Retrieving the real-world ground location of a detection involves two steps: first, constructing a ray from the drone’s camera through the detected pixel, using the drone’s position and orientation; second, finding where that ray meets the ground, accounting for terrain elevation, Earth’s curvature, and atmospheric refraction. The following paragraphs describe each step in detail.

We first construct a ray from the frame centre, using the UAV’s location and pose together with the sensor’s pose relative to the platform, following the Euler angle order of operations defined in [Motion Imagery Standards Board, 2014]. To account for lens distortion, the camera’s intrinsic parameters were separately estimated using OpenCV and a chessboard calibration pattern [Zhang, 2000]. Using these intrinsics together with the camera’s vertical and horizontal field of view, the 2D pixel coordinates of a detection are converted into an angular offset from the frame centre, giving the direction of the ray corresponding to that detection.

Second, the point at which this ray intersects the ground is then found using a Digital Surface Model (DSM), while also accounting for Earth’s curvature and atmospheric refraction. For simplicity, Earth’s curvature is approximated using the ellipsoidal radius of curvature $R(\varphi, \alpha)$ in a given azimuth α , following Euler’s formula

for normal curvature [Torge and Müller, 2012]:

$$R(\varphi, \alpha) = \frac{M(\varphi) \cdot N(\varphi)}{N(\varphi) \cos^2 \alpha + M(\varphi) \sin^2 \alpha} \quad (1)$$

where $M(\varphi)$ and $N(\varphi)$ are the radius of curvature in the north–south (meridional) and east–west (prime vertical) directions, respectively, at latitude φ .

The impact of atmospheric refraction is approximated with an atmospheric refraction coefficient (k) of 0.13. Combined with the radius of curvature above, the elevation offset (Δh) of the DSM at a given ground distance from the sensor is given by [International Telecommunication Union, Radiocommunication Sector (ITU-R), 2017]:

$$\Delta h(d, \varphi, \alpha) = \frac{d^2}{2R(\varphi, \alpha)}(1 - k) \quad (2)$$

where $R(\varphi, \alpha)$ is the azimuth-dependent radius of curvature from Equation 1, k is the atmospheric refraction coefficient, and d is the ground distance along the ray.

To assess the accuracy of the IFP pipeline, we tracked stationary objects across two flight videos, applying IFP every 10 frames to each tracked object. The resulting position estimate was compared against the object’s true location, identified independently in Esri World Imagery, to compute positioning error. Figure 2 shows the combined results across all 12 tracked objects.

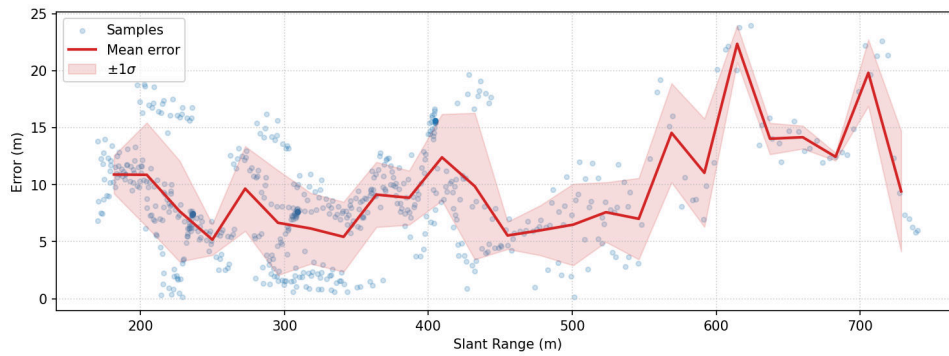


Figure 2: IFP error by distance.

3.3 Visualisation

Even after geo-positioning, the observation stream remains high-dimensional; each detection carries class, confidence, geographic position, drone position, timestamp, and derived attributes. Presenting this raw stream directly to an operator replicates the original information overload problem in a different form. We define four visualisation modes, each reducing the data along a different axis.

The time-centric view (Figure 3a) reduces the video to a one-dimensional activity timeline, allowing operators to identify periods of high activity without reviewing low-content footage, and jump directly to peaks of interest.

The space-centric view (Figure 3b) accumulates all detections across the sequence into a per-class density heatmap overlaid on the base map, making areas of sustained activity immediately legible

The observer-centric view (Figure 3c) displays the drone’s trajectory, colour-coded by detections observed at each position, with the union of all camera footprints outlined to distinguish surveyed from unsurveyed area.

The observation-centric view (Figure 3d) renders each detection as a selectable, colour-coded marker exposing its class, confidence, and geo-location metadata.

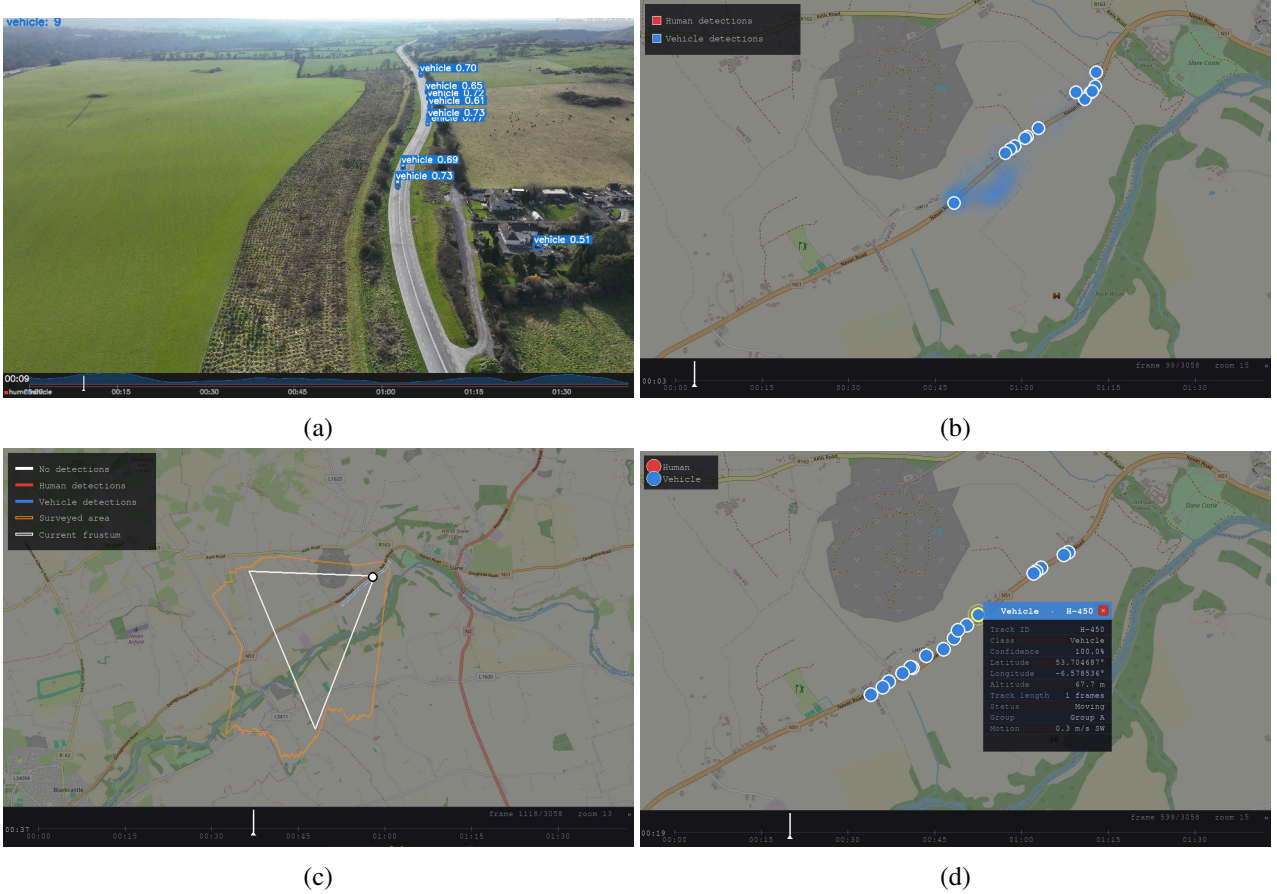


Figure 4: Four complementary visualisation modes for drone-based emergency operations.

4 Experiments

To evaluate the performance of the system on both public benchmarks and custom data, we conduct two complementary evaluations. The first is a standard COCO-style benchmark on the VisDrone Det2019 test dataset, providing a reproducible comparison against the broader aerial object detection literature. However, standard COCO size-based stratification into small, medium, and large is of limited value in the aerial domain. In the VisDrone Det2019 test set, 67.7% fall within the small category, 29.1% are medium, and only 3.2% are large. The overwhelming dominance of the small category means that AP_S effectively summarises the vast majority of objects regardless of the underlying operational scenario. Furthermore, COCO size thresholds are not actionable for drone operators: a pilot would have difficulty keeping objects above 32 pixels in width, but can be instructed to fly within 250m. We therefore conduct a second, distance-stratified evaluation on a bespoke held-out dataset with per-detection distance measurements, which directly characterises the operational envelope of each model configuration in terms meaningful for mission planning and pilot guidance.

Both evaluations compare performance across two training strategies - YOLOv8-x fine-tuned on complete full-resolution frames without tiling, and YOLOv8-x-SF fine-tuned on a combination of full-resolution frames and sliced 640×640 patches following the SF strategy recommended by [Akyon et al., 2022]; and three inference modes: Full Inference (FI), in which inference is carried out on unsliced frames; SAHI, in which full frames are partitioned into 640×640 tiles with 20% horizontal and vertical overlap and a class-aware NMM step resolves overlapping detections across tile boundaries; and FI + SAHI, in which sliced predictions are first merged via NMM before being combined with full-frame predictions in a second NMM pass. For both experiments, metrics are computed at an IoU threshold of 0.5. FPS is reported as inference throughput, measured on an NVIDIA RTX 6000 Ada Generation GPU.

4.1 VisDrone Det2019 Benchmark

Both models are evaluated on the VisDrone Det2019 test dataset under all three inference modes, following the standard COCO evaluation protocol. VisDrone annotations are remapped to two classes: human (VisDrone categories 1–2) and vehicle (categories 3–10); ignored regions (score= 0) and the others category (11) are excluded. This provides a standardised comparison of the effects of training strategy and inference mode on a widely used aerial detection benchmark.

4.2 Distance-Stratified Evaluation

The distance-stratified evaluation dataset was collected from two flight sequences over a rural area in Ireland using a DJI Matrice 4E, capturing video at 1920×1080 resolution and 30 FPS. Frames were extracted at 30-frame intervals, yielding 213 images across 3988 annotated object instances. Ground-truth bounding boxes were annotated manually, with slant range, i.e. direct diagonal distance not ground distance to the target, computed per annotation by projecting the bounding box centres using in-frame positioning as described in Section 3. The dataset distribution is shown in the rightmost column of Figure 5.

For each image, predictions are matched to ground-truth annotations using greedy IoU matching at a threshold of 0.5, with predictions ranked by confidence score. Precision, recall, and AP are computed per image per band and averaged across images. We report both discrete 100 m bands and cumulative bands of the form $[0, d_{\max})$, which show how aggregate performance changes as the distance ceiling is raised. Predictions from the full image are used as candidates regardless of their own projected distance, since the band assignment is defined by the ground-truth object rather than filtering predictions by their own estimated range, treating distance as a known measurement.

5 Results

5.1 VisDrone Det2019 Benchmark

Model	AP ₅₀	AP _S	AP _M	AP _L	AR _S	AR _M	AR _L	FPS
YOLOv8-x + FI	0.459	0.172	0.482	0.626	0.209	0.544	0.685	90.51
YOLOv8-x + SAHI	0.447	0.190	0.365	0.263	0.250	0.423	0.304	24.53
YOLOv8-x + FI + SAHI	0.479	0.178	0.428	0.529	0.238	0.509	0.634	18.76
YOLOv8-x-SF + FI	0.456	0.160	0.510	0.729	0.194	0.571	0.792	86.33
YOLOv8-x-SF + SAHI	0.587	0.256	0.531	0.609	0.312	0.598	0.691	20.37
YOLOv8-x-SF + FI + SAHI	0.551	0.213	0.509	0.621	0.270	0.582	0.717	15.03

Table 1: Mean COCO metrics averaged across all distance bands and classes, with inference throughput (FPS) measured on NVIDIA RTX 6000 Ada Generation.

Table 1 reports COCO detection metrics on the VisDrone Det2019 test set. YOLOv8-x-SF + SAHI achieves the strongest overall performance, attaining the highest AP (0.587), AP_S (0.256), AP_M (0.531), AR_S (0.312), and AR_M (0.598), demonstrating that sliced fine-tuning combined with sliced inference is the most effective configuration for this domain.

Full-frame inference consistently yields higher AP_L and AR_L, which is expected given that sliced inference is not optimised for large objects. The combined FI + SAHI mode improves over full-frame inference alone on AP but falls short of SAHI-only across most metrics.

Throughput varies substantially across inference modes: full-frame inference achieves approximately 86 FPS, while SAHI reduces this to around 20 FPS due to the overhead of slicing and NMM post-processing.

5.2 Distance-Stratified Evaluation

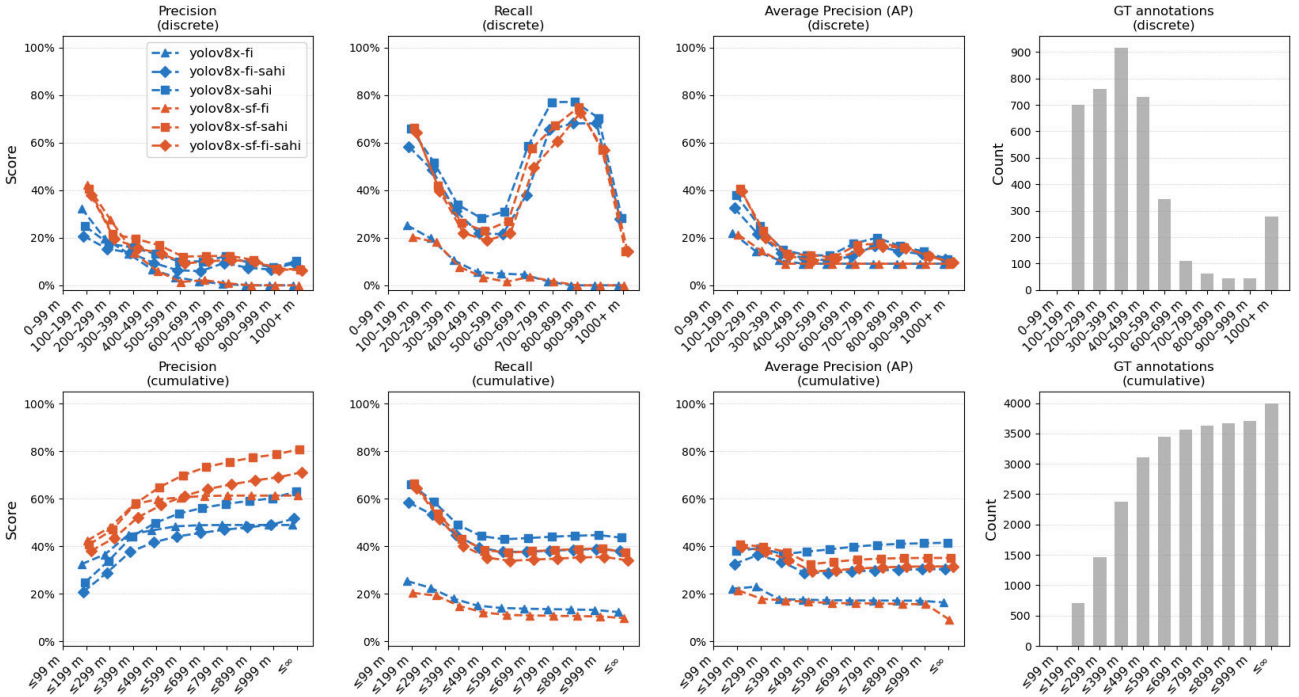


Figure 5: Mean detection performance per distance band at IoU=0.5, averaged across Human and Vehicle classes.

Figure 5 shows precision, recall, and AP stratified by slant range in 100 m discrete bands and cumulative bands up to 1 km.

Because ground-truth slant range is itself derived from the IFP pipeline rather than measured independently, the distance axis inherits IFP positioning error (up to approximately 25 m, Figure 2) rather than being an exact reference. Relative to the 100 m discrete bin width used in Figure 5, this error is large enough to shift the band assignment of detections near a bin boundary, but by no more than one band, while detections near bin centres are effectively unaffected. We therefore expect the reported per-band metrics to be subject to some boundary blur rather than systematic bias, and interpret the cumulative bands in Figure 5, which are less sensitive to single-band reassignment, as the more robust indicator of the operational envelope.

Across both model families, SAHI inference consistently improves recall beyond approximately 200 m, consistent with [Akyon et al., 2022]: at ranges where vehicles subtend only a small fraction of the full frame, localised slice-level context improves sensitivity relative to full-frame inference. This recall gain comes at the cost of precision on near-field objects, the expected fragmentation effect whereby large nearby objects span multiple slices, producing overlapping predictions not fully suppressed by NMM postprocessing.

The combined inference mode FI + SAHI, which applies NMM first to slice predictions and then merges with full-frame predictions in a second pass, does not substantially improve near-field precision over SAHI alone, suggesting the fragmentation effect persists across both NMM stages. The discrete band plots show a recall dip at medium range (300–500 m) and recovery at longer distances, which may reflect performance being dominated by a small number of salient objects at long range.

We note that both the distance-stratified evaluation and the IFP error validation underpinning it are based on limited data: 213 images from two flight sequences at a single site, and IFP error estimated from only 12 tracked objects. The trends reported above should therefore be considered preliminary; a larger and more varied evaluation dataset would be needed to establish firm conclusions.

Peak RAM usage is uniform across all three inference modes at ~1600 MB, whereas sliced inference showed an increase in peak VRAM usage (~2260 MB vs. ~450 MB at batch size 16).

6 Conclusion

We have presented a framework that addresses the situational awareness bottleneck in drone-supported emergency response. The VisDrone benchmark confirms the benefits of slicing at both training and inference time, with YOLOv8-x-SF + SAHI the most effective configuration. Our preliminary distance-based evaluation suggests a detection range of approximately 500 m and shows that SAHI consistently improves recall and precision as distance increases, offering an initial characterisation of the operational envelope for mission planning.

The four visualisation modes offer a principled decomposition of the high-dimensional observation stream, each targeting a distinct operator query pattern: *when was activity observed?*, *where is it concentrated?*, *what area did the drone cover?* and *what is this object doing?*

Future work will prioritise dataset quality and coverage as a prerequisite for validating the operational envelope with respect to extended conditions such as smoke/haze, low-light, and differing UAV and sensor platforms. In the longer term, geo-coded annotations will address the gap between existing datasets and real deployments, exploring combined visual and spatial tracking and re-identification, and behaviour and anomaly detection to extract higher-order intelligence from data.

Acknowledgments

This work was supported with the financial support of Research Ireland-funded LERO Research Centre for Software, grant 13/RC/2094_2.

References

- [Akyon et al., 2022] Akyon, F. C., Onur Altinuc, S., and Temizel, A. (2022). Slicing aided hyper inference and fine-tuning for small object detection. In *2022 IEEE International Conference on Image Processing (ICIP)*, page 966–970. IEEE.
- [International Telecommunication Union, Radiocommunication Sector (ITU-R), 2017] International Telecommunication Union, Radiocommunication Sector (ITU-R) (2017). Recommendation ITU-R P.834-9: Effects of Tropospheric Refraction on Radiowave Propagation. Technical report, ITU-R.
- [Jocher et al., 2023] Jocher, G., Qiu, J., and Chaurasia, A. (2023). Ultralytics YOLOv8.
- [Motion Imagery Standards Board, 2014] Motion Imagery Standards Board (2014). MISB ST 0601.8: UAS Datalink Local Set. Technical report, Motion Imagery Standards Board.
- [Paulin et al., 2024] Paulin, G., Sambolek, S., and Ivašić-Kos, M. (2024). Application of raycast method for person geolocalization and distance determination using UAV images in real-world land search and rescue scenarios. *Expert Systems with Applications*, 237:121495.
- [Rossi et al., 2025] Rossi, L., Morandini, F., Burglin, A., Bertrand, J., Wandon, C., Tollard, A., and Pieri, A. (2025). Dem-based uav geolocation of thermal hotspots on complex terrain. *Remote Sensing*, 17(23).
- [Torge and Müller, 2012] Torge, W. and Müller, J. (2012). *Geodesy*. De Gruyter, Berlin, Boston.
- [Zhang, 2000] Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(11):1330–1334.
- [Zhu et al., 2021] Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., and Ling, H. (2021). Detection and tracking meet drones challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 44(11):7380–7399.

Similarity-Augmented Roadside Object Triangulation from street-level images

Waqar Ahmad¹ and Vladimir A. Krylov¹

¹*ADAPT Research Centre, School of Mathematical Sciences, Dublin City University, Dublin, Ireland*

Abstract

Object geolocation from street-level imagery under weak supervision is challenging due to noisy detections and the absence of bounding-box annotations. This work addresses this task via a similarity-augmented MRF framework that integrates visual consistency into geometry-driven multi-view triangulation. Starting from weakly supervised object detection, this formulation enforces both visual and geometric consistency across paired views to estimate object locations. To discourage visually inconsistent configurations, a Siamese network is trained to estimate pairwise visual similarity. Label noise introduced by weak supervision is addressed using a Beta-distribution-based filtering strategy during Siamese training. We also propose a novel deterministic optimizer for the MRF model that significantly reduces computational requirements and improves reproducibility. Experiments on real-world street-level datasets from Amsterdam and Dublin demonstrate consistent improvements in precision, recall, and localization error over a geometry-only baseline. The implementation of our method is available at https://github.com/waqar3411/Similarity_Aug_MRF.

Keywords: Siamese networks, Beta distribution, object geolocation, image similarity, street-level imagery

1 Introduction

Accurate geolocation of roadside objects from street-level imagery is essential for applications such as urban mapping, infrastructure monitoring, and autonomous navigation. Unlike image-level geolocation, object-level geolocation requires estimating precise real-world object coordinates from multiple views under significant uncertainty in depth, bearing, and appearance. Weakly supervised detectors [1] further amplify this challenge by producing uncertain object locations that generate excessive amounts of low-confidence triangulation candidates. Recent work has therefore shifted toward geometry-driven multi-view inference under weak supervision, where object locations are estimated from noisy geometric cues [2]. While video-based and track-driven approaches improve stability by exploiting temporal continuity [3], they inherently assume ordered observations and reliable tracking assumptions that do not hold in large-scale, crowdsourced image datasets.

Classical learning-based cross-view geolocation estimates camera locations by matching ground-level images to georeferenced aerial or satellite imagery using learned metric representations [4–6]. While effective, these methods rely on strong appearance cues and dense georeferenced databases, making them impractical for object-level localization of custom roadside assets. Beyond purely geometric formulations, recent works [7, 8] explore joint object association and localization in 3D using tracking and graph-based reasoning. Such approaches rely on dense image streams and strong calibration, limiting their applicability to sparser street-level imagery. Recent foundation models such as Segment Anything Model (SAM) and DINO-based methods can improve object localization and visual feature extraction, while DUS3R [9], and VGGT [10], can support depth estimation and multi-view reconstruction. However, their use for globally geolocating roadside objects from sparse views with uncertain cross-view associations remains underexplored.

More recently, GNNs have been adopted to explicitly model relational structure among views, objects, or observations, enabling learned aggregation of geometric and contextual cues [11]. Nassar *et al.* [12] formulate

localization as a graph inference problem, in which spatial relationships among observations are encoded as edges and refined via message passing to improve global consistency. While effective in structured settings, most graph-based approaches require dense supervision or explicit object grouping.

In contrast, purely geometric triangulation-based approaches are prone to failure under dense view coverage, where accumulated measurement uncertainty and ambiguous geometric configurations degrade localization accuracy. To address this, object geolocation is formulated as an energy minimization problem within an MRF, enforcing multi-view geometric consistency through unary and pairwise terms [13]. While effective, this formulation relies primarily on geometric cues, which limit its ability to suppress visually inconsistent yet geometrically plausible configurations in weakly supervised settings. Under weak supervision, MRF-based triangulation groups detections into view pairs that may include both true multi-view observations and spurious pairs arising from false positives [2]. Since both satisfy geometric constraints, geometry alone is insufficient.

In our work, we integrate visual consistency into the MRF framework. Specifically, we train a Siamese network to estimate pairwise visual similarity and propose a novel Beta-distribution-based filtering strategy to remove visually inconsistent pairs during training. The resulting similarity scores are incorporated as an additive energy term in the MRF, penalizing unreliable configurations. We introduce a novel deterministic optimizer for the MRF model that significantly reduces computational requirements and boosts reproducibility. Experiments with real-world street-level imagery demonstrate consistent improvements in object geolocation accuracy.

2 Methodology

This work proposes a noise-aware geolocation framework that improves the performance of multi-view triangulation under weak supervision. The proposed method retains weakly supervised object detection and an MRF-based triangulation framework, while explicitly addressing false positives introduced by weak supervision via similarity-driven noise filtering, as shown in Fig. 1. A learned image similarity term is incorporated into the MRF inference at the inference stage to enhance robustness and geolocation accuracy; see Fig. 2.

In the absence of bounding-box annotations, we adopt the weakly supervised detection and MRF-based geolocation pipeline presented in [2], corresponding to Parts A and B of Fig. 1. Although this framework enables scalable roadside object detection and geolocation, it inevitably produces false-positive detections. Some of these false positives may satisfy the geometric constraints of the MRF and form apparently valid multi-view pairs, leading to incorrect triangulations and reduced geolocation accuracy. To address this limitation, the multi-view image pairs generated by the MRF are processed using the similarity-learning framework shown in Fig. 1C. A Siamese network extracts feature representations from each image pair and computes their cosine similarity. Since the generated pairs may contain incorrect cross-view associations, the similarity scores are modelled using a two-component beta-mixture distribution to identify and remove noisy pairs. The resulting filtered dataset is then used to train the noise-aware similarity model. The trained weakly supervised detection and similarity models are then employed at inference (Fig. 2) for similarity-augmented roadside object geolocation.

2.1 Noise filtering

To address the presence of invalid view pairs, we employ a Siamese network to learn similarity-based relevance between image pairs. The network consists of two weight-sharing CNN branches for visual feature extraction, followed by a fully connected neural network (FCNN) that serves as the classification head. The candidate image pairs generated by the MRF-based geolocation framework are subsequently used to construct the training data for the similarity estimation task. The training data is divided into positive pairs (class 1), formed from images associated with the same object, and negative pairs (class 0), formed from images of different objects. Due to detection noise, the positive class inherently contains both clean pairs corresponding to true multi-view observations and noisy pairs arising from false detections.

Each image pair (v_i, v_j) is passed through two weight-sharing branches of the network, producing deep feature embeddings $f(v_i)$ and $f(v_j)$. The cosine similarity s_{ij} between the embeddings is computed as: $\text{sim}(x, y) = x \cdot y / (\|x\| \|y\|)$, where, $s_{ij} \in [0, 1]$. To distinguish between genuine and noisy image pairs within

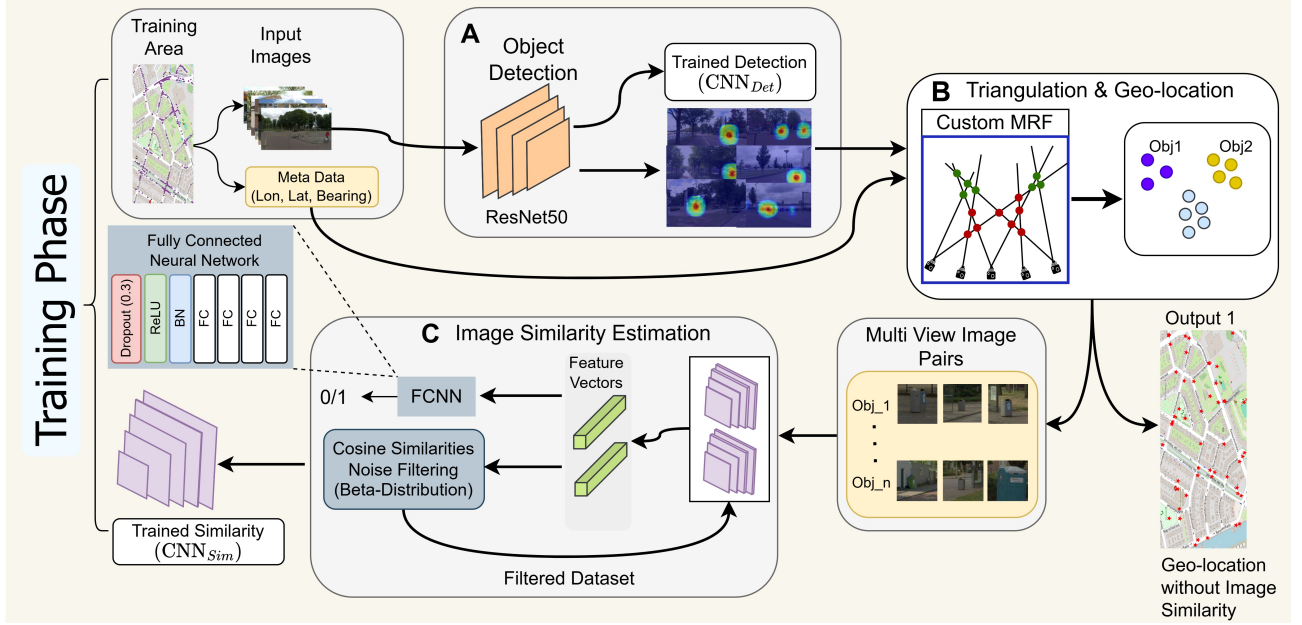


Figure 1: Proposed training pipeline: (A) weakly supervised object classification and Grad-CAM-based localization, (B) MRF-based triangulation to generate multi-view image pairs, and (C) Siamese similarity learning with Beta-mixture filtering.

class 1, we model the distribution of similarity scores using a two-component Beta mixture model, defined as

$$p(s) = \omega \text{Beta}(s | a_1, b_1) + (1 - \omega) \text{Beta}(s | a_0, b_0), \quad (1)$$

where $\text{Beta}(\cdot)$ denotes the Beta probability density function. The component $\omega \text{Beta}(s | a_1, b_1)$ captures high-similarity, visually consistent image pairs, while $(1 - \omega) \text{Beta}(s | a_0, b_0)$ models low-similarity pairs corresponding to noisy or mismatched observations. We adopt the Beta distribution for this task because the similarity scores are naturally constrained to the interval $[0, 1]$. Moreover, its flexibility allows it to model a wide range of distributional shapes, including uniform, skewed, and unimodal forms, making it an appropriate choice for characterizing the unknown similarity distributions associated with both genuine and noisy pairs.

The mixture parameters (a_k, b_k, ω) are estimated using the Expectation-Maximization (EM) algorithm, which is well-suited for mixture models with latent component assignments. EM alternates between estimating posterior probabilities and updating the mixture weights and Beta parameters. This iterative nature enables stable convergence despite overlapping components. More complex alternatives, such as stochastic or simulation-based Bayesian methods, are unnecessary in our setting due to the reliable convergence of the standard EM formulation. The EM algorithm follows the following two alternating steps to estimate and update the mixture parameters:

E-step: Given the current parameter estimates, we compute the posterior probabilities for any s_i as:

$$\gamma_0(s_i) = \frac{\omega_0 \text{Beta}(s_i; a_0, b_0)}{\omega_0 \text{Beta}(s_i; a_0, b_0) + \omega_1 \text{Beta}(s_i; a_1, b_1)}, \quad (2)$$

where ω_0 and $\omega_1 = 1 - \omega_0$ denote the mixture weights. The posterior probability of the second mixture component is obtained as the complement of the first, i.e., $\gamma_1(s_i) = 1 - \gamma_0(s_i)$.

M-step: Using the posteriors from the E-step, samples are assigned to the component with the higher posterior probability, i.e., $\arg\max_k \gamma_k(s_i)$. The shape parameters of each Beta component are then updated via maximum likelihood estimation as $(a_k, b_k) = \text{MLE}_{\text{Beta}}(\{s_i | \gamma_k(s_i) \geq 0.5\})$, $k \in \{0, 1\}$, where $\text{MLE}_{\text{Beta}}(\cdot)$ denotes Beta parameter estimation from the assigned samples.

Starting from a trivial initialization, the parameters α_k and β_k are refined iteratively. The mixture weight ω_k , which represents the current estimate of the proportion of samples associated with component k , is updated at

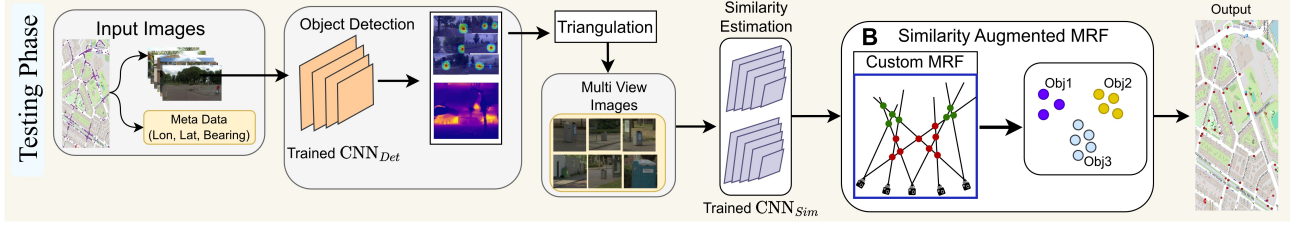


Figure 2: Inference pipeline: object detection and localization, triangulation-based view pairing, Siamese similarity estimation, and final geolocation using a similarity-augmented MRF.

each iteration, based on the posterior probability. Specifically, it is computed as:

$$\omega_k = \frac{|\{s_i \mid \gamma_k(s_i) \geq 0.5\}|}{|D|}, \quad k \in \{0, 1\} \quad (3)$$

where $|D|$ denotes the total number of samples, this update reflects the fraction of the dataset currently attributed to each mixture component.

The proposed training strategy alternates between similarity learning and noise filtering. We first train the Siamese network for K epochs to learn an initial similarity representation from the full dataset. Subsequently, the noise filtering procedure is applied to identify and remove unreliable pairs, and the network is fine-tuned on the filtered dataset for an additional K epochs. This staged fine-tuning, both before and after noise filtering, helps ensure that hard positive examples are learned by the model rather than incorrectly removed as noisy pairs. This alternating process of filtering and fine-tuning is repeated iteratively for C cycles. Training is terminated once the estimated proportion of filtered samples, ω_k , falls below a predefined threshold of 2%, and the final fine-tuned model is retained for evaluation.

2.2 Loss functions

The detection network is trained as a binary classifier using the BCE loss $\mathcal{L}_{CE}$. For similarity learning, a Siamese network is optimized with a joint loss comprising classification and metric objectives. The FCNN classification head is supervised using the cross-entropy loss $\mathcal{L}_{CE}$, while the shared CNN embeddings are regularized using a contrastive loss ($\mathcal{L}_c$),

$$\mathcal{L}_c = \frac{1}{N} \sum_{i=1}^N [y_i d_i^2 + (1 - y_i) \max(0, m - d_i)^2], \quad (4)$$

and cosine embedding loss ($\mathcal{L}_{COS}$),

$$\mathcal{L}_{COS} = \frac{1}{N} \sum_{i=1}^N \begin{cases} 1 - \cos(\mathbf{x}_i, \mathbf{y}_i), & y_i = 1, \\ \max(0, \cos(\mathbf{x}_i, \mathbf{y}_i) - m), & y_i = 0, \end{cases} \quad (5)$$

where d_i denotes the Euclidean distance between embeddings $(\mathbf{x}_i, \mathbf{y}_i)$ and m is the margin parameter.

The overall Siamese training objective is given as a weighted sum of the three contributing losses with a single weight: $\mathcal{L}_{total} = \mathcal{L}_{CE} + \lambda \mathcal{L}_{COS} + (1 - \lambda) \mathcal{L}_c$.

2.3 Information fusion with MRFs

Object detections in images combined with the image metadata are used to construct rays (in 2D) that originate at the camera's GPS coordinates (assumed identical to the camera center). The camera's viewing direction is obtained from the heading provided in the image metadata, while the object's position in the image is used to calculate its final bearing. The bearing (direction of the ray) is selected towards the middle of the segmented object. The rays create a 2D graph with nodes formed by the intersections of these rays. Edges are established between nodes on the intersected rays. An MRF model is employed to estimate binary decisions ($z_i \in \{0, 1\}$)

indicating the presence of the object of interest at a particular node i [13]. The MRF energy of a node z_i is governed by two geometric unary terms and one pairwise term. We extend this formulation by introducing an additive unary term u_2 , which incorporates visual information by rewarding or penalizing a node based on the Siamese similarity of the two views that generate it. Thus, the total energy is defined over all $N_{\mathcal{Z}}$ nodes as:

$$\mathcal{U}(\mathbf{z}) = \sum_{i=1}^{N_{\mathcal{Z}}} \sum_{j=0}^2 \alpha_j u_j(z_i) + (1 - \sum_{j=0}^2 \alpha_j) \sum_{i=1}^{N_{\mathcal{Z}}} \sum_{j=1}^{N_{\mathcal{Z}}} p(z_i, z_j) \quad (6)$$

The first unary term penalizes the discrepancy between the distance from the camera to the node estimated with monocular depth CNN (Δ_{ij}) and with triangulation (d_{ij}): $u_1(z_i) = z_i \sum_{j=1,2} \|\Delta_{ij} - d_{ij}\|$. The u_0 unary term promotes inclusion of terms on any ray R and is inversely proportional to the number of intersections on the ray: $u_0(z_i) = -z_i / (\sum_{x_n \in R} 1)$. The presence of this term ensures that the minimum energy is not reached on the empty configuration $z_i = 0$ for all i . The pairwise term penalizes distances between any two nodes simultaneously active on any view-ray R : $p(z_m, z_n) = \sum_{x_m, x_n \in R} z_m z_n \|x_m - x_n\|$, where x_i is a GPS position of node controlled by state z_i . The similarity unary term u_2 penalizes nodes generated from visually inconsistent view pairs: $u_2(z_i) = z_i(1 - s_i)$, where $s_i \in [0, 1]$ denotes the similarity score predicted by the Siamese network. Nodes supported by visually consistent views ($s_i \approx 1$) incur minimal penalty, while intersections formed by visually dissimilar views are penalized, thereby suppressing spurious nodes that are geometrically admissible but visually inconsistent. The contributions of the terms are governed by $\alpha_i \in [0, 1]$, $\sum \alpha_i < 1$.

The state $\mathbf{z} = \{z_i\}_{i=1, \dots, N_{\mathcal{Z}}}$ that minimizes the energy $\mathcal{U}$ (Eq. 6) is computed using the Quadratic pseudo-Boolean optimization (QPBO) algorithm [14], which is based on graph-cut optimizations. QPBO has two substantial advantages over the original Simulated Annealing-based optimization proposed in [13]: firstly, the optimization result is deterministic, which avoids the need to rerun the procedure multiple times to arrive at an averaged (stable) field configuration. Secondly, the procedure has predictable complexity and is fast. The use of QPBO is possible by eliminating the higher-order penalty terms originally used in [13], which did not allow energy function composition compatible with this type of optimization. Specifically, the stand-alone penalty term u_3 used in [13] has been replaced by the unary term u_0 (Eq. 6). The pairwise term is supermodular [14], but the number of such terms is relatively small, and empirically allows for optimal state assignments to all graph nodes using the QPBO/QPBOP algorithm.

3 Datasets and Experimentation

3.1 Dataset Description

Although several multi-view object geolocation datasets have been proposed, no widely accepted benchmark or evaluation protocol exists [15]. Existing results remain dataset- and implementation-specific, while relevant datasets such as Pasadena Multi-View [12] are not publicly accessible. Here, we employ two publicly available street-level datasets from Dublin and Amsterdam, using public waste bins as the target objects, to support future comparisons.

Dublin. street-level images are obtained via the Mapillary API ¹, selecting only images flagged as containing wastebins. Public street furniture data from the Smart Dublin bins dataset ² is used for spatial supervision: images within a 15 m radius of DCC bin locations are labeled as class 1 (9,900 images), while class 0 samples are selected from images located 30-100 m away from any bin. All images are resized to 1500 × 900. We evaluate on two test sets from Mapillary and the DCC dataset, covering 2.7 km of roads with images sampled every 3 m, six 68.77° views per location.

Amsterdam. Panoramic images are obtained via the Mapillary API, restricted to city-provided panoramas flagged as containing wastebins and captured with a fixed north-facing orientation (0°). The ground-truth locations of wastebins are obtained from the OSM API ³ and used to associate panoramas with object positions.

¹Mapillary: <https://www.mapillary.com>

²Smart Dublin: <https://data.smartdublin.ie/dataset/dcc-public-bin-locations>

³OpenStreetMaps: <https://www.openstreetmap.org>

Table 1: Geolocation performance on the Amsterdam and Dublin datasets against MRF [13]

Dataset	Method	Precision (%)	Recall (%)	Error (m)
Amsterdam	MRF	64.25	79.9	2.54
	Similarity-Aug. MRF	68.2	86.1	1.87
Dublin (Mapillary)	MRF	66.21	80.3	2.51
	Similarity-Aug. MRF	71.4	87.1	1.52
Dublin (DCC)	MRF	65.1	80.8	2.38
	Similarity-Aug. MRF	72.8	87.9	1.49

Table 2: Classification results on the Amsterdam dataset

Model	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)
ResNet-34	94.7	88.2	91.3	92
ResNet-50	95.81	89.10	92.33	93.1
Resnet-101	93.1	88	90.47	92.4

A total of 4,100 panoramas (8000×4000) are split into 2000×2000 sub-images and labeled using Mapillary masks, yielding 20,000 samples, of which 85% are used for training and 15% for testing.

3.2 Training Setup

The training follows a two-stage setup comprising weakly supervised object detection and Siamese-based similarity learning, with MRF-based geolocation performed at inference time.

Backbone architectures. ResNet-50 is used as the primary image classification backbone, with Grad-CAM employed to localize the object within each image. MobileNetV2 is adopted as the backbone feature extractor in the Siamese architecture due to its proven effectiveness in metric learning-based similarity estimation [16].

Implementation details. All models are implemented in PyTorch and trained on an NVIDIA Titan X GPU (12 GB). The detection network is trained for 20 epochs using Adam with a batch size of 8 and a step learning-rate scheduler (step size 5, $\gamma = 0.1$). For similarity learning, the Siamese network employs a MobileNetV2 backbone with an FCNN head (512, 512, 256, 128), batch normalization and ReLU activations, and dropout (0.3) applied to the final two layers. Training uses the Adam optimizer with an initial learning rate of 10^{-3} , a batch size of 32, and a learning-rate decay every 5 epochs. Optimization alternates between Siamese training and Beta-based pair filtering, with $K \in [3, 7]$ epochs per cycle until validation accuracy stabilizes, and terminates when the estimated noisy-pair ratio ω_k falls below 2%. The combined loss is optimized with $\lambda = 0.45$, identified via a grid search.

3.3 Results Analysis

We report results for weakly supervised object detection and geolocation on the Dublin and Amsterdam datasets, including impact analysis of Beta-distribution-based noise filtering.

Object detection performance. ResNet-34, ResNet-50, and ResNet-101 are evaluated for image classification on the Amsterdam dataset. As shown in Table 2, ResNet-50 achieves the best overall performance, with 95.81% precision, 89.10% recall, and 93.1% accuracy, and is therefore selected as the primary classification backbone. Using the selected ResNet-50 backbone, classification performance across all test sets is reported in Table 3. On the Amsterdam dataset, the model achieves 95.81% precision, 89.10% recall, and 93.1% accuracy. Comparable performance is observed on the Dublin datasets, with 93.1% precision, 87.2% recall, and 92.8% accuracy on Dublin (DCC), and 94.21% precision, 89.0% recall, and 93.0% accuracy on Dublin (Mapillary). This indicates that the detection model generalizes well across cities and data sources, providing reliable geolocation.

Effect of noise filtering. We next analyze the effect of the proposed Beta-distribution-based noise filtering on image pairs generated after detection and MRF-based triangulation. Approximately **8.5%** of pairs are filtered in the Dublin dataset and **12.3%** in the Amsterdam dataset, reflecting greater visual ambiguity in panoramic

Table 3: Classification results with Resnet-50 architecture

Test Set	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)
Amsterdam	95.81	89.10	92.33	93.1
Dublin (DCC)	93.1	87.2	90.05	92.8
Dublin (Mapillary)	94.21	89	91.53	93



Figure 3: Filtered image pairs after Beta-distribution-based denoising in Dublin (top) and Amsterdam (bottom)

imagery (Fig. 3). In the absence of ground-truth pair labels, manual inspection indicates that about **85%** of the removed pairs correspond to genuine mismatches. During inference, the pretrained Siamese network consistently assigns low similarity scores to the filtered pairs, confirming the effectiveness of denoising.

Geolocation performance. We now evaluate the geolocation performance of the proposed framework on the Amsterdam and Dublin datasets. At inference, images are processed by the pretrained detection model, triangulated to generate admissible intersections, and refined using similarity scores from the Siamese network.

Table 1 reports the geolocation results in terms of precision, recall, and absolute localization error. Compared to the baseline MRF, incorporating similarity cues consistently improves performance: on the Amsterdam dataset, precision increases by 4% and recall by 6.2%, while the mean localization error is reduced by 0.7 m; similar trends are observed on the Dublin datasets. These results show that integrating similarity-aware filtering into the MRF effectively suppresses spurious intersections arising from noisy detections, leading to higher precision and recall and substantially lower localization error. Overall, the proposed similarity-augmented MRF yields more robust and accurate geolocation across three diverse datasets. The QPBO solver for MRF reduces computational complexity by an order of magnitude and yields deterministic, and therefore reproducible, geolocation results.

4 Conclusion

This work presents a similarity-augmented MRF framework for object geolocation under weak supervision. By integrating Siamese-based visual similarity into a geometry-driven triangulation model and mitigating label noise via Beta-distribution-based filtering, the proposed method effectively suppresses false-positive detections. We deploy a new deterministic optimizer for MRF, resulting in more stable and computationally feasible results. Experiments on real-world street-level datasets from Amsterdam and Dublin show consistent gains in precision, recall, and localization accuracy over a geometry-only MRF baseline. Future work will focus on joint optimization of detection and similarity learning.

Acknowledgements

This work was supported by the ADAPT Centre for Digital Content Technology funded under the SFI Research Centres Programme (Grant 13/RC/2106_P2) and co-funded under the European Regional Development Fund.

References

- [1] Z. Li, Y. Wang, D. Xu, Y. Gao, and T. Zhao, “Tbnet: A texture and boundary-aware network for small weak object detection in remote-sensing imagery,” *Pattern Recognition*, vol. 158, p. 110976, 2025.
- [2] W. Ahmad and V. A. Krylov, “Roadside object geolocation from street-level images with reduced supervision,” in *2024 32nd European Signal Processing Conference (EUSIPCO)*, 2024, pp. 641–645.
- [3] D. Wilson, T. Alshaabi, C. Van Oort, X. Zhang, J. Nelson, and S. Wshah, “Object tracking and geo-localization from street images,” *Remote Sensing*, vol. 14, no. 11, 2022.
- [4] S. Hu, M. Feng, R. M. H. Nguyen, and G. H. Lee, “Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7258–7267.
- [5] Y. Shi, L. Liu, X. Yu, and H. Li, “Spatial-aware feature aggregation for image based cross-view geo-localization,” in *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [6] S. Zhu, T. Yang, and C. Chen, “Vigor: Cross-view image geo-localization beyond one-to-one retrieval,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5316–5325, 2020.
- [7] X. Weng, J. Wang, D. Held, and K. Kitani, “3d multi-object tracking: A baseline and new evaluation metrics,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, p. 10359.
- [8] Z. Pang, J. Li, P. Tokmakov, D. Chen, S. Zagoruyko, and Y.-X. Wang, “Standing between past and future: Spatio-temporal modeling for multi-camera 3d multi-object tracking,” in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2023, pp. 17 928–17 938.
- [9] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, “Dust3r: Geometric 3d vision made easy,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2024, p. 20697.
- [10] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny, “Vggg: Visual geometry grounded transformer,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2025, pp. 5294–5306.
- [11] Z. Ye, Y. J. Kumar, G. O. Sing, F. Song, and J. Wang, “A comprehensive survey of graph neural networks for knowledge graphs,” *IEEE Access*, vol. 10, pp. 75 729–75 741, 2022.
- [12] A. S. Nassar, S. D’Aronco, S. Lefèvre, and J. D. Wegner, “Geograph: Graph-based multi-view object detection with geometric cues end-to-end,” in *European Conference on Computer Vision (ECCV)*, 2020.
- [13] V. A. Krylov, E. Kenny, and R. Dahyot, “Automatic discovery and geotagging of objects from street view imagery,” *Remote Sensing*, vol. 10, no. 5, 2018.
- [14] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer, “Optimizing binary mrfs via extended roof duality,” in *IEEE conference on computer vision and pattern recognition*, 2007, pp. 1–8.
- [15] D. Wilson, X. Zhang, W. Sultani, and S. Wshah, “Image and object geo-localization,” *Int. J. Comput. Vision*, vol. 132, no. 4, p. 1350–1392, Nov. 2023.
- [16] H. Yang, Y. Chen, W. Wu, S. Pu, X. Wu, Q. Wan, and W. Dong, “A lightweight siamese neural network for building change detection using remote sensing images,” *Remote Sensing*, vol. 15, no. 4, 2023.

Efficient automatic traffic light alteration for sparse traffic flow

Afif Imtiyaz Khan Deshmukh, and Kenneth Dawson-Howe

School of Computer Science & Statistics, Trinity College, University of Dublin.

Abstract

This paper presents proof of concept for a novel traffic light control system which reduces average waiting times along with vehicle emissions. Cameras monitor arriving vehicles and pedestrians on all roads leading to the junction, arrival times are predicted, and these are used to change the traffic lights in favour of an approaching vehicle if there is no contention (i.e. no other vehicle or pedestrian approaching from another direction).

Keywords: Computer Vision Application, Vehicle and pedestrian detection, Optical Flow Estimation

1 Introduction

While some research has been performed on optimising traffic flow at busy junctions, very little consideration has been given to junctions with sparse traffic flows and little consideration seems to have been given to pedestrians. The work presented in this paper considers junctions with relatively small amounts of traffic, and also considers pedestrians when deciding how to set the traffic lights. The system tracks all vehicles and pedestrians on all roads and pavements leading to the junction and predicts their arrival times. Based on these estimates, in low traffic volume situations, the traffic lights at the junction can be set to prioritise the first vehicle/person to arrive at the junction, in this way eliminating unnecessary slowing/stopping time. In higher volume traffic situations the traffic lights can operate in the normal fashion.

2 State of the Art

A number of technologies exist for monitoring traffic, the main ones being inductive loops, video imaging and microwave radars. Inductive loop sensors consist of a coil which is excited by an oscillator, this causes it to generate a magnetic field in its surrounding area, this magnetic field resonates at a constant frequency which is monitored by an electronic detector. However, due to the nature of how inductive loops work, they are incapable of detecting the presence of pedestrians and as they are usually placed about 2 meters from the stop line they cannot be used to predict arrival times.. Microwave radars can detect vehicles and pedestrians, and are robust against lighting and weather variations. However, they are unable to detect stopped objects, which can be problematic in traffic-centric applications [Bernas et al., 2018].

Video imaging has been used extensively in traffic detection; for example, used to detect vehicles, count traffic flow to detect when vehicles break red lights, etc. [Guo et al., 2019] proposed a reinforcement learning based approach to traffic signal control at urban intersections, using SUMO to simulate different traffic patterns and junctions to improve waiting times. [Osman et al., 2017] describe a system for using cameras to determine traffic density on roads leading to a junction. The cameras have precalculated viewing angles and while the system is intended to optimise the amount of waiting time, this is specifically aimed at busy junctions, but no consideration is given to pedestrians. [Abbas et al., 2024] describe a system with high accuracy at vehicle detection under varying conditions and alter the green signal duration based on traffic volume. Again this system is targeted at high volume roads and no consideration is given to pedestrians.

The demand for adaptive traffic control systems is rising due to the increasing number of vehicles worldwide.

As of 2022 the SCATS system held the largest market share. SCATS uses information from vehicle detectors located in each lane at the stop line to adjust signal timings in response to variations in traffic demand and system capacity. It does not perform network-wide optimization but instead acts as a heuristic feedback system, adjusting timings based on changes in traffic flow from previous cycles [Hodgkiss and Ta, 2020]. While SCATS can reduce delays in low-traffic conditions, its reliance on data from prior cycles rather than the current moment means its response is not truly real-time, which can limit effectiveness when conditions change suddenly.

Pedestrian considerations are rare in adaptive traffic control research. In the survey by [Qadri et al., 2020], the authors note that most studies focus on traffic signal timings and their effects on vehicle delay and emissions, with little attention to pedestrian needs. One example is the work by [Yu et al., 2017], which models pedestrian delays under high pedestrian demand and unsaturated vehicle traffic, factoring in crossing type and vehicle behavior. However, this work only optimizes static signal timings based on historical data and does not incorporate real-time pedestrian or vehicle data.

3 Video Processing

We use cameras looking at each road approaching the junction, and in each video feed identify road areas and pavement areas, identify vehicles and pedestrians approaching the junction and then predict when they will arrive so that the traffic light can be changed/set optimally.

3.1 Identifying road and pavement regions

Cameras mounted on the traffic lights at intersections can view a specific area of roadway and pavement. To detect vehicles, cyclists and pedestrians approaching the intersection YOLOv8 is used. However, stationary vehicles parked near the intersection can be mistakenly classified as approaching traffic, and vehicles traveling in the opposite direction may also trigger false positives when determining vehicles that are relevant to the current signal phase. To remedy this, Farnebäck dense optical flow algorithm is used to estimate the motion of each detected object between consecutive frames.



Figure 1: Regions identified for vehicles (on the left in red) and for pedestrians (on the right in blue).

The detected vehicles and pedestrians are accumulated over a set period of time to generate a heatmap of active road and pedestrian areas, following the generation of the heatmap, thresholding is applied to this heatmap to define Regions of Interest for the road and pavement. Thresholding is performed by discarding all road and pedestrian regions with values below the median for vehicles and pedestrians, respectively, on their heatmaps (See Figure 1). This helps remove one-off detections, such as vehicles backing out of driveways or pedestrians crossing the road at undesigned areas. Note that whether a bicycle is treated as a vehicle or a pedestrian depends on whether it appears on the road or on the pavement (See Figure 2).



Figure 2: Whether a bicycle is treated as a vehicle or a pedestrian depends on whether it appears on the road or the pavement.

3.3 Predicting arrival times at the junction

To estimate the time remaining before a vehicle reaches the intersection, a Long Short-Term Memory (LSTM) neural network is used. For the use-case of traffic, vehicular motion across frames can be considered as sequential where small variations in velocity or position may indicate acceleration or deceleration. The model is trained using a sequence of 5 consecutive frame-wise feature vectors collected for each tracked vehicle, consisting of the bounding box location, the velocity of the bottom right corner together with the width & height of the bounding box. Each

input sequence corresponds to a unique vehicle, and the output is a single scalar: the number of frames remaining before the vehicle reaches the intersection. See Figure 3. The predicted arrival time for a pedestrian should be computed in a similar manner although probably over a longer number of frames, or by reducing the frame rate (as pedestrian exhibit slower and jerkier motion, relative to vehicles).

3.4 Evaluation

Our dataset for testing consisted of a 20-minute test video which was split into two parts. 80% was used for training and 20% for testing. The prediction of when the vehicles would arrive at the junction was compared with the actual arrival times. The mean average error was 20 frames (equivalent to 0.66s).

4 Traffic Simulation

Testing the system in a real-world traffic intersection poses significant safety risks since it involves altering real traffic lights. Hence to evaluate the effectiveness of the proposed methodology, a simulation based approach is adopted. Since the proposed system focuses on predicting the precise arrival time of detected vehicles and dynamically adjusting the traffic signals, it requires fine-grained analysis on an individual vehicle level. We make use of the Simulation of Urban Mobility (SUMO) model to simulate the behaviour and motion of individual vehicles in terms of position, speed and acceleration [Lopez et al., 2018].

4.1 Intersection Design

A custom intersection was designed in SUMO which consisted of a crossroads (where four roads meet together). Each road has two-way traffic, designated cycle lanes and a pedestrian crossing at the intersection. To simulate the prediction model (indicating when vehicles will arrive at the junction), SUMO's Traffic Control Interface (TraCI) is used. This provides the distance to the intersection and the speed of each vehicle which can be used to predict arrival times. A similar approach is used to detect the presence of pedestrians.

Vehicles are introduced into the simulation dynamically using a stochastic process that introduces variability in both the route choices and the departure intervals. For each vehicle, a random route is selected from the 12 possible routes (as vehicles come along 1 road out of 4 and may continue through the junction in 1 of 3 ways) and a departure time is based on a Gaussian distribution with a set mean interval and standard deviation of 5 seconds. To examine how the system performs across varying traffic intensities, the mean interval is varied across experiments to range from 1 to 18 seconds, effectively simulating traffic flow rates from 3 vehicles/minute to 60 vehicles/minute. This allows for the analysis of how congestion and emissions react to varying traffic conditions.

4.2 Default System

In order to observe the effects of the adaptive traffic light system, the generated traffic was also tested by using a standard timer based schedule. This default schedule assigns predetermined green, yellow and red light durations, while yellow light durations were kept consistent across both adaptive and the default schedule, the green and red light timings for the static schedule were determined using experimental evaluation to ensure a reasonable timing was set according to low traffic scenarios. For the simulated intersection used in this study, the most optimal static timings were found to be 20 seconds of green for each pair of parallel lanes, followed by 20 seconds for all four

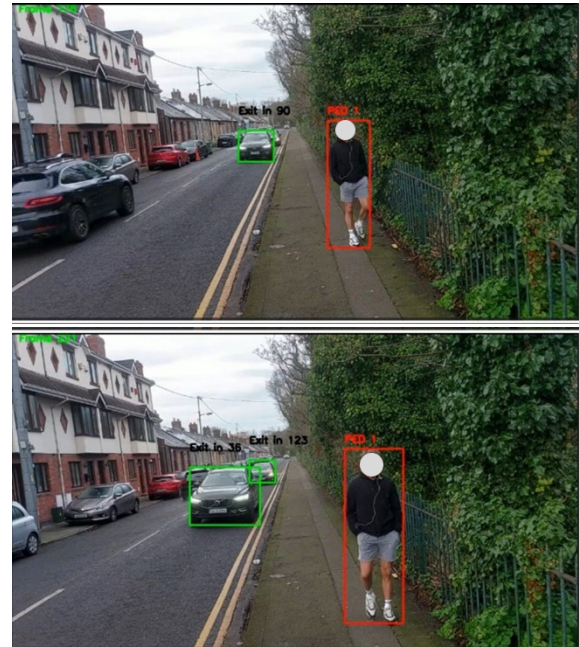


Figure 3: Changing predictions for the number of frames for the lead car to the traffic lights (90 frames to the traffic lights in the left image, and 36 frames to the traffic lights in the right image).

pedestrian crossings

4.3 Evaluation

SUMO can observe each vehicle in the simulation at a granular level, which allows for the extraction of detailed performance metrics such as individual and aggregated waiting times (i.e. the total duration a vehicle remains stationary due to red lights or congestion). The total combined waiting time of all vehicles passing through the intersection is calculated for each simulation run. The set of vehicles, their routes and frequency remain consistent across experiments when making comparisons to ensure a fair evaluation. See Figure 4.

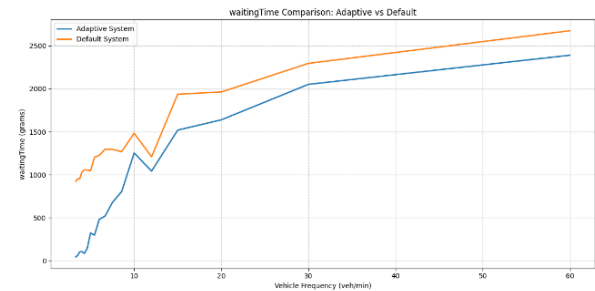


Figure 4: Comparison of the default traffic light control system vs. our adaptive system in terms of absolute average waiting time.

5 Conclusions

This paper has presented a proof of concept for traffic lights with sparse traffic flow, demonstrating separately that vehicular and pedestrian traffic can be located visually and that by predicting when that traffic will arrive at the junction, average waiting time along with emissions can be significantly reduced. Where the number of vehicles per minute was less than 7 the average waiting time dropped by more than 50% while at the same time emissions dropped by more than 20%.

References

- [Abbas et al., 2024] Abbas, S.K., Khan, M.U.G., Zhu, J., Sarwar, R., Aljohani, N.R., Hameed, I.A., Hassan, M.U., (2024) *Vision based intelligent traffic light management system using Faster R-CNN* In CAAI Transactions on Intelligence Technology: Volume 9, Issue 4, pp. 932-947
- [Bernas et al., 2018] Bernas, M., Płaczek, B., Korski, W., Loska, P., Smyła, J., Szymała, P., (2018) *A survey and comparison of low-cost sensing technologies for road traffic monitoring*, Sensors, vol. 18, no. 10, p. 3243.
- [Guo et al., 2019] Guo, M., et al. (2019) *A reinforcement learning approach for intelligent traffic signal control at urban intersections*. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pp. 4242–4247.
- [Hodgkiss and Ta, 2020] Hodgkiss, D., Ta, V.T., (2020) *Adapting future vehicle technologies for smart traffic control systems*, in Toward Sustainable and Economic Smart Mobility: Shaping the Future Of Smart Mobility. World Scientific, 2020, pp. 49–63.
- [Lopez et al., 2018] Lopez, P.A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., Wiessner, E., (2018) *Microscopic Traffic Simulation using SUMO*. In 2018 21st International Conference on Intelligent Transportation Systems (ITSC)}, pp.2575-2582.
- [Osman et al., 2017] Osman, T.,and Psyche, S., and Ferdous, S., Zaman, H., (2017) *Intelligent traffic management system for cross section of roads using computer vision*. In Proceeding of the IEEE 7th Annual Computing and Communication Workshop and Conference, pp.1-7
- [Qadri et al., 2020] Qadri, S.S.S.M., Gökçe, M.A., Öner, E., (2020) *State-of-art review of traffic signal control methods: challenges and opportunities*. In European Transport Research Review, vol. 12, no. 1, Oct. 2020.
- [Yu et al., 2017] Yu, C., Ma, W., Han, K.,Yang, X., (2017) *Optimization of vehicle and pedestrian signals at isolated intersections*. In Transportation Research Part B Methodological, vol. 98, p. 135–153, Jan. 2017.

Automatic Passing Vehicle to Cyclist Distance Estimation

Steven Bondaruk, and Kenneth Dawson-Howe

School of Computer Science & Statistics, Trinity College, University of Dublin, Dublin, Ireland

Abstract

This paper presents initial work on the development of a novel computer vision-based tool to automatically detect the distance between a cyclist and a passing vehicle from a camera on another vehicle (either in front or behind the bicycle being viewed). In a range of test videos the average error was only 88mm, making it feasible to use to assess how closely vehicles are passing bicycles.

Keywords: Computer Vision Application, YOLO, Object Detection

1 Introduction

For safety in Ireland (and many other countries) vehicles overtaking cyclists are required to leave a minimum clearance of 1m where the speed limit is 50km/h or less and a minimum clearance of 1.5m where the speed limit is over 50km/h. However, there appear to be no statistics around how closely vehicles overtake cyclists.

The research in this paper develops a simple system which can determine the clearance distance. A camera is mounted looking forwards or backwards on a bicycle with the intention of measuring the clearance distance for a cyclist either in front of or behind the cyclist with the camera.

2 State of the Art

Traditional distance estimation methods typically rely on stereo vision or LiDAR, but monocular systems are gaining popularity due to their cost-effectiveness and simplicity. [Huang et al., 2019] proposed a robust method for inter-vehicle distance estimation using monocular vision by combining vanishing point detection with object localization. Their framework demonstrated strong performance under varying road conditions and established the value of geometric cues in single-camera setups.

Expanding on this idea, [Liu et al., 2021] integrated convolutional neural networks (CNNs) with monocular inputs to estimate distances to vehicles. Their architecture combines object detection and regression tasks, highlighting the potential of deep learning to infer spatial relationships directly from image content without explicit calibration.

[Zhu and Fang, 2019] introduced a novel learning-based approach to predict object-specific distances from monocular images by encoding both object and context features. Their network was trained on datasets where ground truth distances were known, enabling it to generalize to new scenarios. This approach aligns closely with our use of known reference objects—like license plates—as scaling factors.

[Masoumian et al., 2021] further developed this direction by integrating deep learning-based object detection with monocular depth estimation models. Their work demonstrated the importance of combining traditional computer vision geometry with learned depth predictions for improved accuracy, especially in applications such as traffic monitoring.

[Kainz et al., 2015] and [Kul and Sayer, 2022] both proposed methods for inferring object size and wall dimensions, respectively, from 2D images using reference-based scaling, such as that used in this research.

3 System Design

The system takes a video as input, smooths the images to reduce noise, uses Yolo to find people, bicycles, vehicles and licence plates, and computes the distance between a person on a bicycle and a passing car (when the wheel of the car is in line with the wheel of the bicycle).

3.1 Cyclist Detection

To detect cyclists, a YOLOv11 based object detector is employed to identify bounding boxes corresponding to the class “person” and the class “bicycle”. Since YOLO models do not natively classify a “cyclist” class, cyclist detection is inferred by locating bounding boxes where the “person” class overlaps with the “bicycle” class. Among multiple detections, the largest person bounding box in the frame is prioritized, assuming the cyclist is closest to the camera. This helps distinguish actual cyclists from pedestrians or background figures and ensures that our measurement pertains to the target cyclist. See Figure 1.



Figure 1: The bicycle (yellow box), cyclist (red box), passing vehicle (white outline), and licence plate (orange box) are all located using YOLO. The distance from the vehicle’s wheel to the bicycle/person is measured (purple line) in pixels and converted to centimetres using the size of the licence plate

3.2 Vehicle Detection

To accurately localize the vehicle, an instance segmentation model is used, which provides pixel-wise outlines instead of simple bounding boxes. This is particularly important for identifying precise points like wheel locations. A YOLOv11 variant for segmentation model is used for vehicle instance detection. The segmented output enables fine-grained localization, essential for identifying the rear or front bottom edges of the vehicle that are used for distance reference points. See Figure 1.

A pre-trained YOLOv8 license plate detection model is integrated to detect the vehicle’s license plate. To filter out false positives, a positional verification step is introduced: detected license plate bounding boxes must lie within the region of the segmented vehicle instance. This ensures that only valid plates associated with vehicles in the scene are used for scaling the pixel distances into millimetres (as licence plate width is standard). See Figure 1.

3.3 Passing Distance Estimation

Once the license plate bounding box is located, its pixel width is extracted ($Width_{pixels}$). Given that standard license plate widths for private vehicles and taxis in Ireland are approximately 520 mm ($Width_{mm}$), these numbers allow us to scale pixel distances to millimetres where the distance is roughly in the same plane. Hence we need to identify a line between the bicycle and the vehicle which meets this criteria.

We need (a) a point in line with the bottom of the bicycle which is directly below the outside of the handlebars and (b) the bottom of a wheel of the vehicle when it is in (horizontal) line with point (a). See the purple line in Figure 1. Note that the line between the points (a) and (b) is determined for every frame and the one closest to the horizontal is used for distance estimation.

To locate the cyclist’s side boundary, the system identifies the end of the handlebars using the bicycle’s bounding box. The location depends on the relative position of the vehicle. If the vehicle is to the right of the cyclist, the bottom-right corner of the bicycle’s bounding box is used. If the vehicle is to the left, the bottom-left corner is chosen. This coordinate approximates the cyclist’s lateral boundary for distance measurement.

The vehicle’s wheel location is determined from its instance segmentation mask. The location is based on the position of the cyclist. If the cyclist is to the left of the vehicle, the system identifies the bottom-leftmost pixel of the vehicle segmentation. If the cyclist is to the right, it extracts the bottom-rightmost pixel. This point serves as the edge of the vehicle closest to the cyclist and is used as the reference point for distance calculation.

With the reference points on both the bicycle handlebar end and vehicle wheel edge identified, the system computes the horizontal pixel distance between them ($Distance_{pixels}$). This is then converted into millimetres ($Distance_{mm}$) using the licence plate width. This estimated value represents the real-world lateral distance between the cyclist and the passing vehicle in the given video frame. See Equation (1).

$$Distance_{mm} = (Distance_{pixels} \times Width_{mm}) / Width_{pixels} \quad (1)$$

4 Evaluation

To test the system we recorded 20 videos of a cyclist being overtaken by vehicles. In order to be able to determine the actual distance between the cyclist and the vehicles, the cyclist remained still, standing over a speed bump with fixed width sections. Sample processing is shown in Figures 2 and a plot of the ground truth vs. computed distances is shown in Figure 3 along with a bar chart summarising the errors in Figure 4. The mean average error was 88mm with a standard deviation of 108mm.

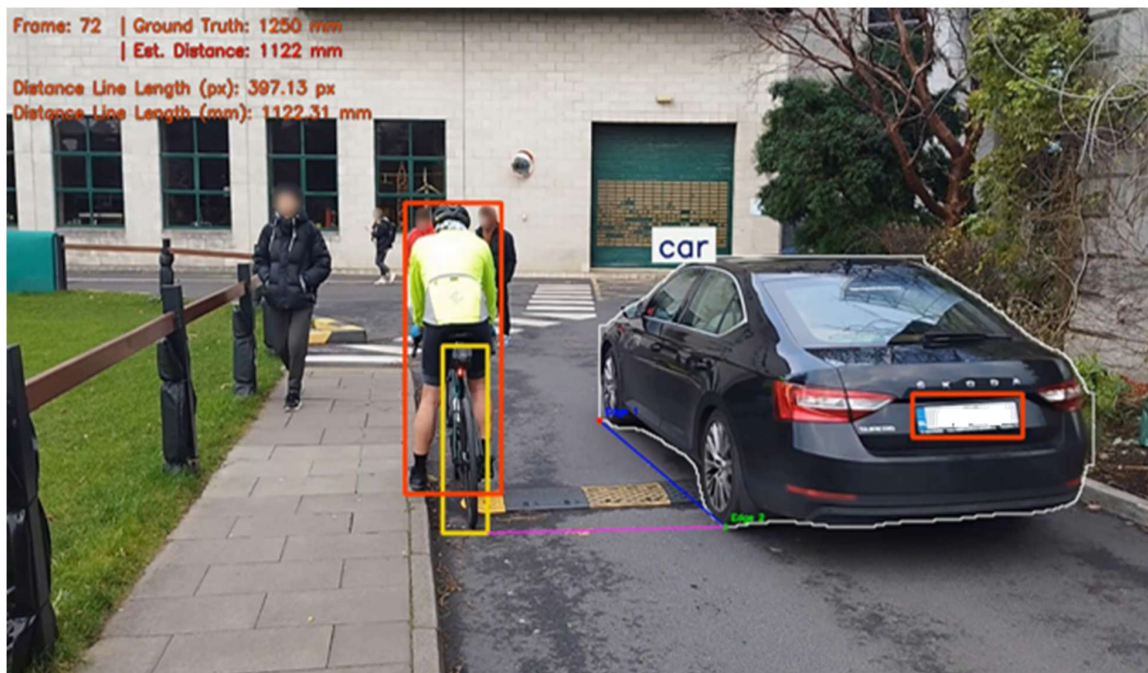


Figure 2: Sample processing and computed distances.

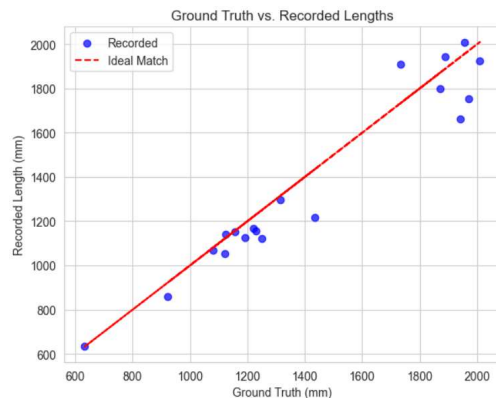


Figure 3: Distances computed between bicycles and passing vehicles (vertical) vs. the actual distance between the vehicle and the bicycle (horizontal). Each blue dot represents a single test case. The ground truth was computed by using the physical distance on the speed bumps. The diagonal red line represents perfect results so blue dots closer to the line are more accurate

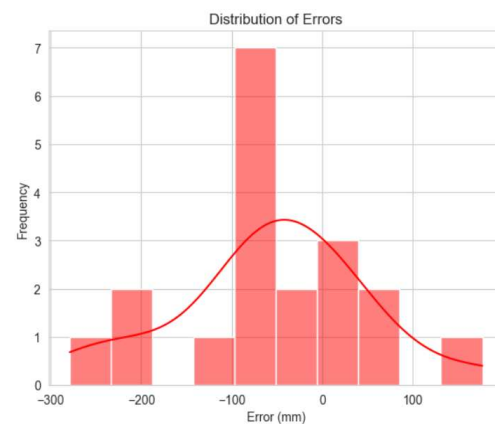


Figure 4: Bar chart summarising the errors across all test cases. The mean absolute error was 88mm with a standard deviation of 108mm

5 Conclusions

This initial research has confirmed the viability of the concept of measuring passing distance from a single uncalibrated video feed. The accuracy of the results was quite remarkable given that off-the-shelf YOLO models have been used to find the people, bicycles, cars and licence plates. We would expect further improvements in accuracy through further specialised image processing (to locate the boundaries of objects, such as the licence plates, more precisely). There is also potential to look at using dashcam footage from other vehicles to solve this problem, although in this circumstance the wheels of the passing vehicle are unlikely to be visible to the camera.

References

- [Huang et al., 2019] Huang, L., et al., (2019) *Robust inter-vehicle distance estimation method based on monocular vision*. In IEEE Access, vol. 7, pp. 46059–46070, Apr. 2019
- [Kainz et al., 2015] Kainz, O., Jakab, F., Horečný, M.W., Cymbalák, D., (2015) *Estimating the object size from static 2D image*, in Proc. Int. Conf. and Workshop on Comput. and Commun. (IEMCON), Vancouver, BC, Canada, Oct. 2015, pp. 1–5.
- [Kul and Sayer, 2022] Kul, S., Sayar, A., (2022) *Wall size prediction from 2D images with the help of reference objects*. In Innovations in Smart Cities Applications, Vol. 5, M. Ben Ahmed, A. A. Boudhir, İ. R. Karaş, V. Jain, and S. Mellouli, Eds. Cham: Springer, Mar. 2022, pp. 925–933.
- [Liu et al., 2021] Liu, Q., et al., (2021) *Vehicle distance estimation based on monocular vision and CNN*. In Proc. Int. Conf. Comput. Inf. Sci. Artif. Intell. (CISAI), Kunming, China, Sept. 2021, pp. 638–641
- [Masoumian et al., 2021] Masoumian, A., et al., (2021) *Absolute distance prediction based on deep learning object detection and monocular depth estimation models*. In Artificial Intelligence Research and Development, Oct. 2021, pp. 325–334.
- [Zhu and Fang, 2019] Zhu, J., Fang, Y., (2019) *Learning object-specific distance from a monocular image*. In Proc. IEEE/CVF Int. Conf. Computer Vision, 2019, pp. 3839–3848

A Computer Vision Tool to Aid Beginner Electric Bass Players with Fretboard Navigation

Zemyna Kulikauskaite, and Kenneth Dawson-Howe

School of Computer Science & Statistics, Trinity College, University of Dublin.

Abstract

This paper presents early research into the development of a computer vision-based learning tool designed to assist novice electric bass guitarists. The fretboard, strings, hand and finger positions are automatically detected. Musical input is parsed from standard bass tablature and synchronised with real-time finger placement and plucking detection to generate visual cues on the live video feed of their guitar for correct note positioning.

Keywords: Computer Vision Application, Edge Detection, Hand Pose Estimation, Optical Flow Estimation.

1 Introduction

For beginners learning to play the electric bass guitar, one of the most significant early challenges is navigating the fretboard [Harrison, 2010]. Traditional methods, such as printed tablature, require users to mentally translate

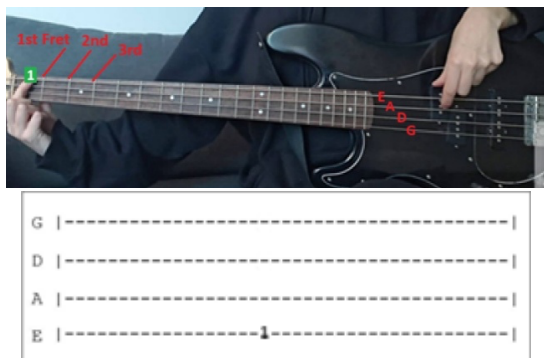


Figure 1: The relationship between the fretboard and the tablature (bottom), with the E string to be plucked while the 1st fret is held down

notation into physical movements on the instrument, often leading to cognitive overload, slow progress, and discouragement. This paper presents a novel computer vision-based learning tool designed to assist novice bassists by overlaying real-time visual guidance directly onto their instrument. A live video feed of the player's bass guitar, detects the fretboard and the four strings (typically tuned to E, A, D, and G), and tracks hand and finger positions using the MediaPipe Hands framework. Musical input is parsed from bass tablature and synchronised with real-time finger placement and plucking detection to generate on-instrument visual cues for correct note positioning. The system enforces deliberate practice by advancing only when the correct note is detected (See Figure 1).

2 State of the Art

Much work in this field has concentrated on Automatic Music Transcription (AMT) as opposed to aiding learning. Video is used in AMT as audio is insufficient (as the same note can often be played on multiple strings and at various fret positions in string instruments). This redundancy creates ambiguity in note identification when using audio-only systems. This limitation is particularly problematic for generating tablature notation, which is essential for guitarists and bassists who rely on precise finger placement information for learning and performance. For example, [Asmar, 2022] presented a novel approach for the automatic transcription of guitar music using computer vision and depth data from RGBD cameras. The process for detection begins with fretboard segmentation, where a deep learning model is trained to produce a binary mask of the fretboard area. This segmentation isolates the fretboard from the background, improving the accuracy of subsequent steps such as string and fret detection. Additionally, the system uses optical flow analysis to detect string plucking or picking motions, which is essential

because pressing a string without plucking does not produce sound. Their system focuses on generating tablature by solely relying on visual input, effectively eliminating the complexities associated with audio signal processing.

[Ghaleb et al., 2024] present a system that generates guitar tablature by combining video inputs with audio analysis. The visual component uses deep learning models to detect the fretboard and hand positions, even in challenging conditions such as occlusions or rapid note transitions. On the audio side, the system applies the Fourier Transform to convert time-domain signals into frequency-domain data, extracting precise note information.

The most similar work we could locate was by [Elashmawi et al., 2023] who present a hybrid system that integrates finger motion capture with note frequency recognition to assess guitarists' performance. This system does offline analyses videos being played by guitarists using acoustic guitars.

3 System Design

Our system has two inputs: (1) a live camera feed from a stationary camera showing the user playing the guitar and (2) the tablature of the music to be played. The live camera feed is processed to find the fretboard and the hands, and the tablature is processed to track the current/next note to be played.

Visual guidance is overlaid on the video feed to show finger placement and to highlight the note on the tablature. See Figure 2.

3.1 Bass guitar recognition

The fretboard is located using edge detection and the Hough transformation. The dominant near horizontal lines will come from the guitar strings. These are used to rotate the image so the guitar strings are roughly horizontal. The frets will then be roughly orthogonal lines. The positions of the frets relative locations are defined by the *12th-root-of-2* rule. See Figure 3. This formula defines the distance from the nut to the n -th fret as

$$x_n = L \times (1 - 2^{-n/12}) \quad (1)$$

where x_n is the distance from the nut to the n -th fret, L is the total scale length (distance between the nut and the bridge), and n is the fret number.

3.2 Plucking detection

MediaPipe Hands (Zhang, 2020) is used to detect the hands and finger positions of each hand. The hands are classified as being the fretting hand or the plucking hand depending on their position relative to the fretboard.

The fingertip positions are used both to determine what string is plucked and what fret of that string is depressed. Plucking is detected using a combination of fingertip tracking and motion analysis. The index and middle fingers of the plucking hand are tracked and the distance from each of the strings determined. A potential pluck is detected as a fingertip remaining near one string for several frames followed by a rapid upward transition to another string (or roughly that distance). Optical flow is used on the potentially plucked string to verify that motion has occurred on that string. Once a pluck is verified by movement on the string, the fret position is determined by analysing the fingertip

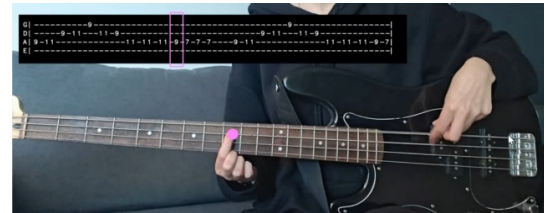


Figure 2: Live video feed with a highlight (pink dot) indicating where the player should be depressing the string/fret (the 9th fret on the A string). The tablature which is being played is shown above with the relevant note highlighted in the pink rectangle



Figure 3: Fret locations determined from the from the vertical lines between the strings using the *12th-root-of-2* rule

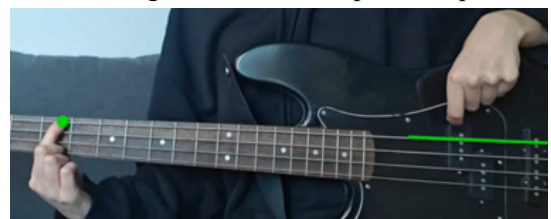


Figure 4: A pluck being detected by the system

position of any finger overlapping the plucked string. See Figure 4.

3.3 Musical Processing

The system accepts musical input in the form of tablature (a format which is widely used and intuitively understood by beginner bass players). The tablature is read from left to right and shows 4 lines (one for each bass guitar string G, D, A, E). Fret numbers are indicated on one string at a time indicating what string should be plucked and the number indicates what fret should be depressed. Fingertip positions are used both to determine what string is plucked and what fret of that string is depressed. See Figure 5.

Once a tablature is loaded the system displays the tablature over a live view of the bass guitar being played. The note to be played is highlighted in the tablature (by putting a pink rectangle around it) and the string and fret position are highlighted on the live video of the guitar (by putting a solid pink circle in the location of the fret which should be depressed).

Only once the relevant fret is depressed and the relevant string plucked does the system move onto the next note on the tablature.



Figure 5: The system interface showing the tablature (top) and a live video feed of the guitar with a pink dot highlighting the string/fret to be played

4 Evaluation

The system was tested by (1) objectively testing how accurately notes were recognised, and (2) subjectively assessing how useful the system seemed to users.

4.1 Assessment of note recognition

To assess the performance of the Fret-Nav system to recognise the correct notes played, a test video was recorded in which a sequence of 157 individual notes were plucked across all four strings of a standard four string precision model electric bass guitar. The notes detected were compared manually with the notes actually played resulting in an accuracy of 91% and a precision of 94% (See Table 1).

	Actual Positive	Actual Negative
Predicted Positive	151	9
Predicted Negative	6	-

Table 1: Pluck detection summary.

Most false positives were caused by ambiguous plucking gestures and finger movement in the test video that were misclassified as note events. A significant amount of the false positives ($\approx 44\%$) were concentrated on the E string. This can likely be attributed to the string's position at the top of the instrument and the lack of a higher string above it, a factor that limits the system's ability to verify plucking gestures through motion patterns that depend on placing a finger on the string above the one being played, as it does for the A, D, and G strings. The increased sensitivity of the E string detection is a side effect of the model's design and although a phantom string is used to account for the differences of the top string, it may require future calibration or separate plucking detection rules to prevent over-triggering.

False negatives were most commonly caused by errors in finger tracking. In several instances, a string pluck was correctly identified, but the MediaPipe-based finger tracking component reported fingertip positions that fell just outside the expected fret region. This could be due to imprecise fingertip landmark estimation or noise in the video data. These results highlight that the reliability of note identification is highly dependent on the accuracy of both components — pluck detection and fret positioning — and even minor inaccuracies in either can result in

missed detections.

4.2 User study

Our initial subjective assessment of the system involved six different participants with varying levels of prior musical experience but no experience of playing the electric bass guitar. All participants received a short introduction to two-finger plucking technique and how to read tablature. They were given one hour to learn and practice the same sequence. Half of the participants used the traditional method to learn and the other half used the Fret-Nav system.

Following the session, each participant completed a structured survey combining Likert Scale questions and open-ended responses. Table 2 provides an overview of the study results for the Likert Scale questions.

Survey Category	Traditional Group	Fret-Nav Group
<i>Engagement</i>	Agree to Strongly Agree	Agree to Strongly Agree
<i>Ease of Learning</i>	Neutral to Agree	Agree to Strongly Agree
<i>Learning Method Effectiveness</i>	Mostly Agree	Mostly Strongly Agree
<i>Need for Additional Guidance</i>	Neutral to Agree	Neutral to Disagree
<i>Ability to Self-Correct</i>	Mostly Neutral	Mostly Agree

Table 2: Summary of user survey results.

Subjectively, the system improved learners’ confidence, self-correction ability, and engagement compared to traditional learning methods. Participants viewed the tool as intuitive, effective, and enjoyable. While areas for refinement were identified — such as pluck detection calibration and UI adjustments — the results from this pilot study strongly support the tool’s value as a beginner-oriented learning aid. The combination of real-time feedback and intuitive visual guidance clearly reduces the cognitive load associated with early-stage fretboard navigation.

5 Conclusions

This paper presents a novel live application of computer vision in music education for the bass guitar. A prototype system was developed and tested showing potential to aid in the learning process. The initial system described in this paper achieved a note detection accuracy of 91% and precision of 94%. A user study comparing the use of Fret-Nav to traditional learning techniques showed increased engagement, self-correction ability, and reduced cognitive load among participants using the tool.

References

- [Asmar, 2022] Asmar, M. (2022). *A Computer Vision-Based Automatic Transcription of Guitar Music from RGBD Videos* Mémoire de maîtrise, Polytechnique Montréal. PolyPublie. <https://publications.polymtl.ca/10470/>
- [Elashmawi et al., 2023] Elashmawi, W.H., Emad, J., Serag, A., Khaled, K., Yehia, A., Hager Sobeah, M., Ali, A. (2023) *A novel approach for improving guitarists’ performance using motion capture and note frequency recognition*. Applied Sciences, 13(10), 2023.
- [Ghaleb et al., 2024] Ghaleb, A., Eslam Elsadawy, E., Essam, I., Zaky, S., Abdelhakim, M., Fahim, N., Bayoumi, R., Hindy, H., (2024) *Taptotab: Video-based guitar tabs generation using ai and audio analysis*. International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), pages 148–155.
- [Harrison, 2010] Harrison, Eli (2010) *Challenges facing guitar education*. Music Educators Journal, 97(1):50–55, September 2010.

A Hybrid Deployment Framework for Industrial Quality Control Using GelSight Tactile Sensing

Kaviyashree Damodaran, Antonio Gambale, Dermot Kerr, Sonya Coleman

Intelligent Systems Research Centre, Ulster University, U.K.

Abstract

Tactile sensing offers a potential alternative to vision-based defect detection for reflective, glossy, or low-texture materials common in precision manufacturing. This paper addresses the binary classification of surface defects using GelSight tactile imagery, which captures high-resolution geometric deformations through an elastomeric interface. The primary challenge is the significant “domain gap” between natural images used for pre-training deep learning models and the specific tactile patterns of GelSight sensors, which is further exacerbated by the scarcity of large-scale tactile datasets. To address this, we compare a classical Histogram of Oriented Gradients (HOG) and Support Vector Machine (SVM) pipeline against three ImageNet-pretrained convolutional neural networks (CNNs), including MobileNet V2, ResNet-18 and EfficientNet-B0. The results indicate that in data-constrained tactile perception, hand-engineered features continue to offer unmatched precision and computational efficiency. We recommend a layered deployment strategy for robotic quality control: utilising HOG-SVM for real-time, precision-driven filtering to minimise false positives, and MobileNet V2 for safety-critical scenarios where maximising defect capture is paramount.

1 Introduction

Tactile sensing is a solid solution in scenarios where visual cues alone are insufficient to capture fine-grained geometric details. A new type of tactile sensor, GelSight, provides high-resolution tactile images by measuring surface deformation through an elastomer interface, enabling the detection of defects as small as 1–2 microns, including scratches, dents, and irregularities [Yuan et al., 2017]. GelSight has emerged as a powerful modality for surface inspection in robotics and is well suited to industrial quality control tasks [Yuan et al., 2017].

Extensive research has established the GelSight sensor as a leading modality for high-resolution tactile perception, particularly in fields requiring detailed surface geometry and contact mechanics. For example, GelSight sensors have been used for accurate classification of 13 materials captured from 42 distinct indoor and outdoor items [Shuvo et al., 2026]. Additionally GelSight has been widely utilised in robotics for complex tasks such as dexterous manipulation, slip monitoring, and object grasping, where it enables real-time 3D geometry reconstruction and force estimation through photometric stereo techniques [Cui et al., 2023]. In industrial metrology, tactile sensing can be used for measuring micro-scale features like roughness, scratch depth, and hole diameters on diverse materials, including transparent acrylics where vision-based methods typically struggle. Recent GelSight developments include the “digital fingertip”, a ruggedised, miniaturised version designed for defense robotics to provide real-time tactile intelligence for autonomous systems [Metrology News, 2026]. Furthermore, research in biomedical engineering has introduced specialised variations like TacScope, which can detect subsurface tissue abnormalities up to 5 mm deep, demonstrating the sensor’s versatility in surgical applications [Prince et al., 2025].

In this work, we address the binary classification of defect versus no defect regions using GelSight tactile images of painted manufactured wood products. The defects considered include localised lumps, subsurface adhesive failures beneath the finish, and uneven or degraded paint layers, all appearing at varying spatial locations across the panel and presenting subtle, low contrast tactile signatures. The objective of this study is to compare a

strong classical baseline based on Histogram of Oriented Gradients (HOG) features with a Support Vector Machine (SVM), against a range of ImageNet pretrained convolutional neural networks. While deep learning methods have achieved strong performance in many vision-based tasks [Krizhevsky et al., 2012, Howard et al., 2017], their effectiveness in tactile domains remains uncertain due to differences in data characteristics and reliance on large scale pre-training [Raghu et al., 2019]. The comparison focuses not only on standard performance metrics such as F1 score, precision, recall, and Area over the curve (AUC), but also on robustness to perturbations and practical deployability in resource constrained industrial settings. Through this analysis, we aim to identify the most suitable approach for tactile defect detection under realistic data and operational constraints.

2 Methodology

2.1 Dataset and Preprocessing

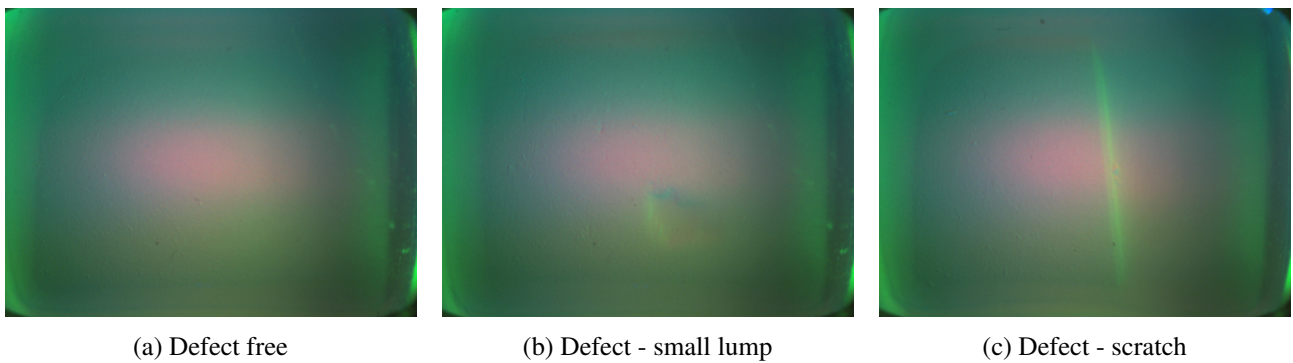


Figure 1: GelSight tactile imagery samples acquired from painted manufactured wood panels under controlled robotic scanning conditions: (a) a defect-free surface baseline; (b) a localised subsurface micro-lump defect presenting subtle geometric deformation; and (c) a macro-scale surface scratch showing sharp directional gradient changes

The tactile image dataset (samples of which are presented in Figure 1) was collected using a robotic manipulator (Universal Robots UR5e) to apply motion parameters across both defect and non-defect regions on painted manufactured wood products using the GelSight elastomeric sensor [Adelson et al., 2011]. This controlled setup ensures reproducible acquisition of images by creating reliable surface scanning conditions critical for reliable model evaluation. The motion parameters were fixed as: 20 mm/s velocity, 0.2 m/s² acceleration, and -0.25N tool contact force [Universal Robots, 2023]. This controlled acquisition ensures reproducible sensor-surface interaction across all 320 high-resolution 860×860 pixel images approximating robotic industrial inspection conditions [Xie et al., 2021]. The dataset consists of 192 defect and 128 non-defect images stratified into a conventional 80/10/10 train-validation-test split using a fixed random seed, preserving an exact 60:40 defect-to-no-defect ratio across all subsets [Pedregosa et al., 2011]. This eliminates stochastic partitioning variation, enabling fair comparison between classical learning and deep learning approaches while reflecting quality assurance sampling where defects constitute 30–50% of inspected batches.

For Deep learning approaches, images are resized, Contrast Limited Adaptive Histogram Equalization (CLAHE) enhancement applied [Shorten and Khoshgoftaar, 2019], with random horizontal flips and ± 0.05 rad rotations for pose robustness. Data augmentation is applied on-the-fly only during training. We utilise the `RandomFlip` layer configured for horizontal shifting alongside `RandomRotation(factor=0.05)`. Here, the factor is interpreted as a fraction of a full 2π rotation, which results in $\pm 18^\circ$ of geometric rotation [Chollet, 2017]. These augmentations introduced controlled pose variation consistent with industrial robot repeatability constraints and improved robustness to boundary artifacts introduced by padding, without distorting defect textures [Shorten and Khoshgoftaar, 2019]. Classical methods used the original feature representations

without augmentation, but were trained and tested on the same splits, ensuring that performance differences come from the models themselves rather than differences in data exposure.

2.2 Classical Baseline: HOG-SVM

In order to process images using Histogram of Oriented Gradients (HOG) feature extraction, the images were first resized to 128×128 pixels then converted to greyscale [scikit-image Developers, 2020]. This balances detail preservation with computational efficiency. HOG features were extracted using 9 orientations covering 0° – 180° , with 8×8 pixel cells grouped into 2×2 blocks (50% overlap) and L2-Hys normalisation [Dalal and Triggs, 2005]. Feature vectors of 8100 dimensions were generated which effectively captured key indicators for surface defects including: the local gradients, edge directions, and contact patterns in the tactile images [Dalal and Triggs, 2005]. This preprocessing step standardises feature scales, ensuring equitable contributions to the decision boundary and prevents any single dimension from dominating due to magnitude disparities [Pedregosa et al., 2011]. A linear SVM, implemented within scikit-learn’s `StandardScaler-SVM` pipeline [Pedregosa et al., 2011], using the default decision threshold of $t = 0.5$, was trained to classify the HOG features. The hyperparameters of the model were optimised by 5-fold stratified `GridSearchCV` on $C \in \{0.1, 1, 10\}$ with `class_weight = balanced` to maintain the inherent 60:40 defect: no-defect imbalance, selecting $C = 0.1$ which maximised cross-validation F1-score whilst preventing overfitting on the 320-image dataset [Pedregosa et al., 2011, Ghosh et al., 2019]. HOG-SVM is well known to excel on small, structured datasets, enabling visualisation of the model’s learned feature weights to identify the edge orientations most critical for defect discrimination [Dalal and Triggs, 2005]. This interpretable classical pipeline establishes a robust baseline for evaluating deep learning approaches in data-constrained industrial tactile inspection scenarios [Chang and Lin, 2011].

2.3 Deep Learning Models

MobileNetV2 [Sandler et al., 2018], ResNet-18 [He et al., 2016], and EfficientNet-B0 [Tan and Le, 2019] were selected as the pre-trained architectures used to classify the GelSight tactile images. Together, they cover a range of model complexities, from lightweight networks suitable for edge deployment to more computationally intensive models that serve as accuracy references. All models were trained on identical 224×224 CLAHE-preprocessed images [Pizer et al., 1990], with a stratified 60:40 defect to non-defect ratio preserved across the train, validation, and test splits. This ensures that performance differences reflect model capacity [Pedregosa et al., 2011]. In order to allow models to retain useful low-level tactile features, while adapting the higher-level layers to specific patterns, certain parameters were selected. Training was controlled using early stopping based on validation AUC (patience = 15 epochs, minimum delta = 0.001) [Prechelt, 1998]. In terms of fine-tuning, the final 20 layers of each network were updated using the Adam optimiser [Kingma and Ba, 2014] with a learning rate of 1×10^{-5} .

Threshold optimisation addressed classification boundary sensitivity in MobileNetV2, where the default 0.5 decision boundary yielded suboptimal F1-scores due to tactile image variability. F1-optimal thresholds were determined via precision-recall curve analysis on the validation set, selecting the threshold that maximised the F1-score for imbalanced defect detection [Davis and Goadrich, 2006]. To ensure robustness, candidate thresholds identified from the validation set were further evaluated using stratified K -fold cross-validation [Pedregosa et al., 2011]. These candidates included Youden’s index ($t = 0.835$) [Youden, 1950] and a recall-constrained threshold ($t = 0.615$). Cross-validation was then used to evaluate intermediate operating regions, resulting in a final fixed threshold of $t = 0.73$ for all subsequent testing and robustness experiments. This final threshold provides a balanced trade-off between precision and recall under tactile image variability.

3 Results

We conducted performance evaluation of the various machine learning methods using the test set of 19 defects and 13 non-defects. The results are presented in Table 1 and it highlights distinct strengths of different models for

industrial deployment scenarios. HOG-SVM achieved the strongest overall performance with perfect precision and superior F1-score compared with other methods, demonstrating unmatched data efficiency for low-data tactile inspection. This shows that the classical method, though only using a small number of effective parameters, performs well, thus emphasising the effectiveness of gradient-based feature representations in low-data tactile inspection environments. On the other hand, MobileNetV2 achieves perfect recall with a balanced precision. This establishes that the model positively ensures that all defect instances are detected, making it particularly suitable for industrial inspection scenarios with high value products.

Table 1: Performance benchmarking using the GelSight test set; models are ranked by their F1-score

Method	F1	Recall	Precision	AUC	Params
HOG-SVM	0.919	0.850	1.000	0.986	378
MobileNetV2 (A5)	0.790	1.000	0.667	0.914	3.5M
ResNet-18	0.769	0.714	0.833	0.896	11.2M
EfficientNet-B0	0.333	0.214	0.750	0.725	4.1M

ResNet-18 is potentially a suitable choice for quality control scenarios due to its apparent balanced and consistent performance, whilst EfficientNet-B0 demonstrates moderate capability, suggesting potential use in pre-screening roles, though it did not match the stability of other models. All the CNNs were evaluated at the optimised threshold ($t = 0.73$) to standardise the decision boundary across all deep learning models while HOG-SVM used the default $t = 0.5$ [Youden, 1950]. Additionally, robustness analysis examined the effects of additive noise, blur, brightness variation, and rotation perturbations on the models. The performance of the HOG-SVM pipeline degraded significantly under noise and rotation perturbations. On the other hand, brightness variation did not affect it. This robustness is due to the fact that gradient based descriptors are sensitive to high frequency and orientation perturbation. In contrast, MobileNetV2’s predicted outputs for all perturbations were very similar at the chosen threshold ($t = 0.73$) suggesting saturation of threshold rather than robustness. This indicates that the model focused on defect recall at the expense of sensitivity to input variations, with stable model predictions but low discriminative adaptation under perturbation.

4 Discussion

The observed performance differences indicate a clear domain gap between tactile image data and the natural image distributions used for CNN pre-training [Yuan et al., 2017]. Consistent with this mismatch, HOG-SVM achieved a higher F1 score (0.919) than MobileNetV2 (0.790), ResNet-18 (0.769), and EfficientNet-B0 (0.333), indicating stronger adaptation to tactile features. Although HOG-SVM and the CNNs use the same dataset and Keras augmentation, CNNs retain ImageNet pre-training bias toward semantic/texture features [Deng et al., 2009]. In contrast, HOG-SVM extracts gradient-based features directly from GelSight’s geometric deformations [Dalal and Triggs, 2005, Adelson et al., 2011]. This domain mismatch is amplified in small datasets, preventing effective adaptation of CNN feature hierarchies [Chollet, 2017].

Model efficiency analysis shows that HOG-SVM provides strong performance with only 378 parameters, while the CNN-based models require millions of parameters, such as MobileNetV2 (3.5M), ResNet-18 (11.2M), and EfficientNet-B0 (4.1M) (see Table 1) [He et al., 2016]. With only a few thousand effective parameters (grid cells $\times$ 9 bins $\times$ L2 normalisation), the performance of data-constrained regimes depends more on feature-signal alignment than representational capacity as adding complexity without data only increases the difficulty of optimisation [Prechelt, 1998]. The lightweight CNN architectures such as EfficientNet-B0 results further reinforce this point. Despite being designed for efficiency, they rely on carefully balanced scaling and training dynamics that assume access to larger, more diverse datasets [Tan and Le, 2019, Howard et al., 2019]. In this setting, they exhibit instability and underfitting, likely due to sensitivity to learning rate selection and limited ability to specialise high-level features for tactile geometry [Kingma and Ba, 2014].

Post-hoc error analysis revealed that false negatives (Fig. 2(a)) arise from edge artefacts and surface patterns

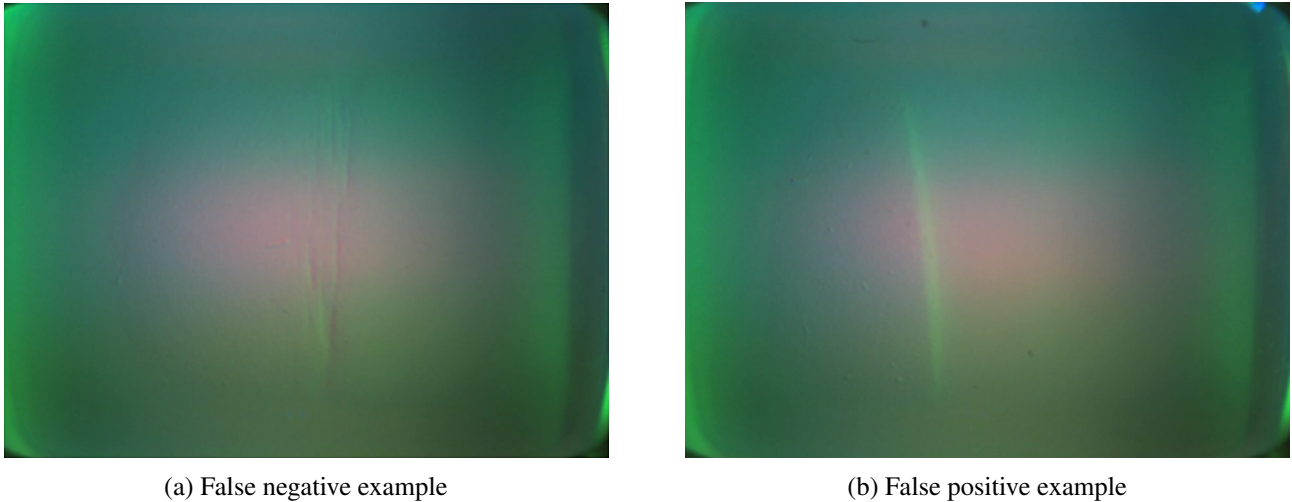


Figure 2: Primary failure modes observed during validation and test partitions: (a) False Negative (FN) classification due to low-contrast tactile signatures of real defects; and (b) False Positive (FP) where high-frequency edge artifacts or natural wood grain structures mimic true defect gradients

mimicking defect gradients (e.g., lifted wood tabs, non-conforming grain), while false positives (Fig. 2(b)) correspond to faint, low-contrast deformations (e.g., small lumps, uneven paint) that are difficult to distinguish from background variation [Davis and Goadrich, 2006]. Targeted augmentation and data collection focusing on glossy and low-contrast defects may help mitigate these failure modes [Shorten and Khoshgoftaar, 2019].

These findings support the use of hybrid classical and machine learning approaches for low-data tactile inspection, particularly where surface geometry rather than semantic content governs defect discrimination. Failure cases reinforce this observation, showing that edge artefacts and low-contrast defects benefit from explicit gradient sensitivity. Future work should explore hybrid HOG–CNN ensembles to combine robust geometric representations with learned hierarchical features, particularly as larger datasets become available.

5 Conclusion and Recommendations

This research addresses the critical challenge of high-resolution tactile defect detection in industrial settings where visual inspection is often hindered by surface reflectivity or low texture. Using a dataset of GelSight tactile images captured from painted manufactured wood products, the paper compares a classical hand-engineered approach against modern deep learning models. The core problem identified is the "domain gap" between natural images used for pre-training convolutional neural networks (CNNs) and the specific geometric deformations captured by GelSight sensors, a challenge further compounded by the limited availability of large-scale tactile datasets.

The methodology involved a rigorous comparison between a HOG-SVM pipeline and three ImageNet-pretrained CNN architectures: MobileNet V2, ResNet-18, and EfficientNet-B0. The study utilised 320 high-resolution images with a stratified split to ensure a consistent defect to non-defect ratio across training and testing. While the CNNs underwent threshold optimisation and data augmentation to improve robustness, the classical HOG-SVM model focused on extracting gradient-based features directly from the tactile surface deformations.

The outcomes demonstrate a significant trade-off between precision and recall across the tested models, with precision ranging from 0.667 to 1.000 and recall ranging from 0.214 to 1.000. The HOG-SVM model emerged as the most efficient, achieving a perfect precision of 1.000 and the highest F1-score while utilising only a fraction of the parameters required by deep learning architectures. Conversely, MobileNet V2 achieved perfect recall, ensuring that every defect was detected, though at the cost of more false positives. Other lightweight models like EfficientNet-B0 struggled with instability, suggesting that representational capacity is less important

than feature-signal alignment in data-constrained tactile environments.

In conclusion, the paper recommends a layered deployment strategy for industrial robotic inspection. HOG-SVM is proposed as a primary classifier for real-time, precision-driven applications to minimise false alarms, while MobileNet V2 is better suited for recall-critical scenarios followed by human verification. Future work should explore hybrid HOG-CNN ensembles to combine the strengths of geometric feature extraction with hierarchical learned features as larger, more diverse tactile datasets become available for complex robotic quality control [Dalal and Triggs, 2005, Yuan et al., 2017].

References

- [Adelson et al., 2011] Adelson, E. H., Johnson, K., Raj, A., and Cole, F. (2011). Microgeometry capture using an elastomeric sensor. In *ACM SIGGRAPH 2011 Talks*.
- [Chang and Lin, 2011] Chang, C.-C. and Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27.
- [Chollet, 2017] Chollet, F. (2017). *Deep Learning with Python*. Manning Publications.
- [Cui et al., 2023] Cui, S., Wang, S., Wang, R., Zhang, S., and Zhang, C. (2023). Learning-based slip detection for dexterous manipulation using gestereo sensing. *IEEE Transactions on Neural Networks and Learning Systems*, 35(10):13691–13700.
- [Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893. IEEE.
- [Davis and Goadrich, 2006] Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, pages 233–240.
- [Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. IEEE.
- [Ghosh et al., 2019] Ghosh, S., Das, N., Das, I., and Maulik, U. (2019). Understanding deep learning techniques for image segmentation. *ACM Computing Surveys (CSUR)*, 52(4):1–35.
- [He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.
- [Howard et al., 2019] Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V., and Adam, H. (2019). Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1314–1324.
- [Howard et al., 2017] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- [Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- [Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 25.

- [Metrology News, 2026] Metrology News (2026). Gelsight to develop rugged 'digital fingertip' sensors for intelligent robotics. <https://metrology.news/gelsight-to-develop-rugged-digital-fingertip-sensors-for-intelligent-robotics/>. Metrology and Quality News - Online Magazine. Accessed: 2026-05-06.
- [Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.
- [Pizer et al., 1990] Pizer, S. M., Amburn, E. P., Austin, J. D., Cromartie, R., Geselowitz, A., Greer, T., Romeny, B. t. H., Zimmerman, J. B., and Fuchs, H. (1990). Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3):355–368.
- [Prechelt, 1998] Prechelt, L. (1998). Early stopping—but when? In *Neural Networks: Tricks of the Trade*, pages 55–69. Springer.
- [Prince et al., 2025] Prince, M. R. I., Athar, S., Zhou, P., and She, Y. (2025). Tacscape: A miniaturized vision-based tactile sensor for surgical applications. *Advanced Robotics Research*, n/a(n/a):e202500117.
- [Raghu et al., 2019] Raghu, M., Zhang, C., Kleinberg, J., and Bengio, S. (2019). Transfusion: Understanding transfer learning with applications to medical imaging. In *International Conference on Learning Representations (ICLR)*.
- [Sandler et al., 2018] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520.
- [scikit-image Developers, 2020] scikit-image Developers (2020). scikit-image: Image processing in python. <https://scikit-image.org/>.
- [Shorten and Khoshgoftaar, 2019] Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60.
- [Shuvo et al., 2026] Shuvo, M. G. R., Coleman, S., Kerr, D., and Quinn, J. (2026). Material classification using visio-tactile sensor for haptic feedback generation. In Rönning, J. and Filipe, J., editors, *Robotics, Computer Vision and Intelligent Systems*, pages 364–382, Cham. Springer Nature Switzerland.
- [Tan and Le, 2019] Tan, M. and Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning (ICML)*, pages 6105–6114.
- [Universal Robots, 2023] Universal Robots (2023). *URScript User Manual*. Universal Robots. Version 5.9.
- [Xie et al., 2021] Xie, X., Cai, J., and Tang, G. (2021). Deep learning for surface defect detection: A review. *Sensors*, 21(23):7952.
- [Youden, 1950] Youden, W. J. (1950). Index for rating diagnostic tests. *Cancer*, 3(1):32–35.
- [Yuan et al., 2017] Yuan, W., Dong, S., and Adelson, E. H. (2017). Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, 17(12):2762.

Parameter-Efficient Dense Prediction for Cross-View Geo-Localisation in Dense Urban Environments

Lucas Scott, Sonya Coleman, Dermot Kerr, Philip Vance, Yunzhou Zhang

School of Computing, Engineering and Intelligent Systems, Ulster University, Northern Ireland

Abstract

Cross-view geo-localisation (CVGL) aims to estimate the geographic position of a ground-level query image by matching it against overhead reference imagery. Despite recent advancements in transformer-based architectures, robust matching in dense urban environments remains impeded by substantial viewpoint variations and structural ambiguities. Recent frameworks, such as CVSegGuide, have integrated semantic segmentation maps to provide high-level scene context and reduce this domain gap. This study evaluates the efficacy of such semantic components by isolating the segmentation task from the broader CVGL pipeline. We perform a rigorous comparison between the established Multi-Scale Transformer (MST) baseline and several U-Net family architectures. Using the Brooklyn and Queens dataset, we evaluate locally trained MST, U-Net, Lite U-Net, and U-Net++ models. Quantitative results demonstrate that convolutional encoder-decoder architectures consistently provide superior land-cover masks compared with the isolated transformer baseline. Specifically, the 5-level U-Net achieves a peak mean Intersection over Union ($mIoU$) of 74.17%, while the 4-level Lite U-Net demonstrates exceptional parameter efficiency, reaching 71.58% $mIoU$ with only 4.662M parameters. These findings suggest that dedicated segmentation architectures offer more reliable semantic guidance than current unified transformer branches, providing a robust foundation for future attention-guided CVGL systems.

Keywords: Cross-View Geo-Localisation, Semantic Segmentation, Parameter Efficiency, Dense Prediction, Urban Land-Cover Mapping.

1 Introduction

Cross-View Geo-Localisation (CVGL) facilitates autonomous navigation by aligning ground-level imagery with georeferenced satellite or aerial maps. While traditional global navigation satellite systems (GNSS) are susceptible to signal denial and jamming, visual localisation provides a passive and resilient alternative [Yue et al., 2025, Yao et al., 2024]. However, the drastic disparity in perspectives between street-level panoramas and overhead images creates a significant “domain gap”, complicating direct feature matching in complex urban scenes. Historically, models relied on low-level visual features; however, recent shifts toward high-level semantic understanding suggest that structural landmarks provide more robust cues for localisation than transient visual textures [Castaldo et al., 2015].

To mitigate the domain gap, researchers have increasingly utilised semantic segmentation maps as a bridge between ground and overhead perspectives. By describing scene information through categories such as buildings, roads, and vegetation, segmentation-guided systems can focus on permanent spatial configurations while ignoring transient environmental changes. The CVSegGuide framework [Xavier et al., 2025] exemplifies this approach, using land-cover maps to guide transformer-based attention. However, the performance of these guidance mechanisms is strictly upper-bounded by the fidelity of the segmentation component. If the underlying segmentation branch fails to accurately delineate building footprints or road boundaries, the resulting attention maps will introduce spatial ambiguity into the CVGL pipeline.

This dependency necessitates a modular evaluation of the segmentation component. While integrated CVGL models often optimise for retrieval or coordinate regression, the internal segmentation branch must be rigorously assessed as an independent dense prediction task to ensure maximum reliability. By prioritising the development and deployment of high-accuracy segmentation methods, we can provide future CVGL systems with the most precise semantic priors possible.

Within semantic segmentation, the U-Net architecture has remained a foundational framework due to its encoder-decoder structure and skip connections, which preserve fine-grained spatial information while progressively learning higher-level representations [Ronneberger et al., 2015]. These properties make U-Net particularly well suited to aerial land-cover mapping, where accurate boundary delineation between roads, buildings, vegetation and background regions is essential. Although transformer-based architectures have recently gained prominence for their ability to model long-range contextual relationships [Zheng et al., 2021], convolutional U-Net variants continue to offer strong localisation performance, architectural simplicity, and favourable computational efficiency. This makes the U-Net family a relevant and necessary benchmark when evaluating segmentation backbones for CVGL-oriented semantic priors.

This paper benchmarks the Multi-Scale Transformer (MST) backbone against established convolutional encoder-decoder architectures to identify the most effective framework for aerial land-cover mapping. We investigate the U-Net family, specifically analysing standard U-Net, Lite U-Net, and U-Net++, to determine if the local inductive biases of convolutions offer advantages over the global attention mechanisms used in current transformer-based branches. Our primary contributions include: (i) a performance comparison of transformer versus convolutional architectures for urban segmentation, (ii) a data-lineage study using the Brooklyn and Queens dataset, and (iii) an analysis of parameter efficiency across varying model depths.

The remainder of this paper is organised as follows: Section 2 details the dataset and class-reduction strategy; Section 3 describes the experimental methodology and model configurations; Section 4 presents and analyses the quantitative and qualitative results; and Section 5 discusses the implications for future integrated CVGL systems.

2 Dataset

The experiments utilise the Brooklyn and Queens (Near/Remote) dataset, which provides paired overhead and ground-level imagery for dense urban environments. Each overhead image is 256×256 pixels with a ground sampling distance (GSD) of approximately 30 cm [Workman et al., 2017, Workman et al., 2022]. To ensure a controlled comparison with the benchmarks established in CVSegGuide [Xavier et al., 2025], we filter the dataset to include only aerial images with at least one corresponding ground-level panorama. This yields a training set of 34,876 images from the Brooklyn subset and a validation set of 7,414 images from the Queens subset.



Figure 1: Example images from the Brooklyn (left two) and Queens (right two) subsets of the dataset.

While the original masks provide eight semantic classes, we retain the four primary categories: *Tree Canopy*, *Buildings*, *Roads*, and *Other Impervious*. This consolidation serves a dual purpose: it ensures di-

rect comparative alignment with the baseline framework and mitigates the class-imbalance issues caused by the extreme sparsity of categories such as *Water*, *Bare Soil*, and *Railroads* in the study area. Representative samples from the filtered dataset are shown in Figure 1, as visually evidenced in these examples, the four classes are well-represented across diverse urban scenes, capturing a balanced mix of structural features ranging from dense residential areas to road networks.

Pixels belonging to these omitted classes are excluded from the loss calculation during training and masked during evaluation. As demonstrated in Table 1, the resulting class distribution remains highly consistent between the training and validation subsets, ensuring that our performance metrics reflect the structurally significant urban features necessary for stable cross-view matching.

Table 1: Pixel distribution across Brooklyn (Training) and Queens (Validation) subsets.

Class	Brooklyn (Training)		Queens (Validation)	
	Pixels (M)	%	Pixels (M)	%
Tree Canopy	396.8	18.26	90.9	20.42
Buildings	708.3	32.59	125.6	28.21
Roads	399.3	18.37	94.5	21.24
Other Impervious	668.7	30.77	134.1	30.14

3 Methodology

To evaluate the impact of architectural selection on semantic guidance, we employ a two-stage experimental design: a baseline reconstruction of the CVSegGuide MST model [Xavier et al., 2025] followed by a controlled comparison against several convolutional alternatives. This modular approach allows for the independent assessment of segmentation quality before investigating its downstream effects on localisation.

3.1 Architectural Configurations

The primary comparison evaluates the Multi-Scale Transformer (MST) against three variants of the U-Net family. While the MST baseline evaluates the original framework’s outputs, our locally trained configurations use strictly aerial imagery to isolate architectural performance.

- **Local MST:** A transformer-based backbone that extracts feature representations at multiple spatial resolutions, upsampled to produce four-channel land-cover maps [Fan et al., 2021].
- **U-Net:** A convolutional encoder-decoder architecture utilising skip connections to preserve spatial detail across downsampling layers [Ronneberger et al., 2015].
- **Lite U-Net:** A low-complexity variant designed to assess the viability of lightweight architectures in resource-constrained CVGL applications [Li et al., 2024].
- **U-Net++:** A nested encoder-decoder structure with dense skip pathways designed to refine feature fusion between the encoder and decoder [Zhou et al., 2018].

For each U-Net variant, we implement both **4-level** and **5-level** depths. The 5-level configurations introduce an additional downsampling stage, effectively expanding the receptive field and parameter capacity to determine if performance gains result from structural innovation or simple model scale.

3.2 Training and Evaluation Protocol

All models are trained from scratch using randomly initialised weights and follow a standardised training protocol using 256×256 aerial inputs. In alignment with the baseline replication [Xavier et al., 2025], no RGB normalisation or data augmentation is applied. Training is conducted on the Brooklyn subset using the Adam optimiser ($lr = 1 \times 10^{-5}$) and a batch size of 5 for a maximum of 100 epochs. Early stopping is triggered after five epochs without improvement in validation micro Intersection over Union (IoU). The optimisation objective weights Binary Cross-Entropy (BCE) and Dice loss equally to balance pixel-level classification with region-level segmentation quality [Mao et al., 2023, Sudre et al., 2017].

Performance is primarily measured using the macro-averaged Mean IoU ($mIoU$) across the four retained classes ($C = 4$). For a specific class c , the metric is computed via the confusion-matrix formulation:

$$IoU_c = \frac{TP_c}{TP_c + FP_c + FN_c}, \quad mIoU = \frac{1}{C} \sum_{c=1}^C IoU_c \quad (1)$$

where TP_c , FP_c , and FN_c represent true positives, false positives, and false negatives, respectively. To ensure a robust assessment, these components are accumulated across the entire Queens validation set prior to calculation, mitigating the impact of image-level variance.

While $mIoU$ serves as the definitive metric for model comparison, the micro IoU formulation is utilised for training-time checkpointing. By aggregating counts across all pixels regardless of class, micro IoU provides a stable optimisation signal:

$$IoU_{\text{micro}} = \frac{\sum TP_c}{\sum TP_c + \sum FP_c + \sum FN_c} \quad (2)$$

This dual-metric approach ensures that optimisation reflects total pixel accuracy while final evaluations respect the equal importance of each environmental landmark. Training and inference were performed on a GTX 1080 GPU with 32 GB of RAM.

4 Results and Discussion

The performance analysis presented in Table 2 reveals a substantial accuracy gap between the Multi-Scale Transformer (MST) and the U-Net family. This section discusses the quantitative trade-offs in parameter efficiency and architectural depth, followed by a qualitative examination of segmentation consistency.

Table 2: Performance comparison using the Queens validation set and per-class IoU and macro mIoU.

Model	Level	Params	Per-Class IoU (%)				mIoU
			Tree	Bldgs	Roads	Other	
MST	—	55.45M	58.04	74.50	72.98	52.49	64.50
U-Net	4	36.96M	68.80	81.03	82.14	60.41	73.09
U-Net	5	148.12M	69.63	83.01	83.76	60.30	74.17
Lite U-Net	4	4.66M	68.04	80.76	79.60	57.94	71.58
Lite U-Net	5	8.28M	66.47	79.43	78.48	57.69	70.52
U-Net++	4	36.62M	68.02	82.58	82.99	60.13	73.43
U-Net++	5	147.67M	69.65	82.56	83.09	60.03	73.83

4.1 Quantitative Performance and Parameter Efficiency

The U-Net family consistently outperformed the MST across all evaluation metrics. Notably, the standard 4-level U-Net achieved a 13.3% relative improvement in $mIoU$ over the MST baseline while utilising 33% fewer

parameters. This disparity suggests that the local inductive biases of convolutional structures are more effective for capturing the precise geometric boundaries of urban landmarks than the global attention mechanisms of the MST.

Increasing architectural depth from Level 4 to Level 5 yielded marginal gains for standard and nested architectures. For instance, the Level 5 U-Net achieved the highest overall accuracy (74.17%), but required a four-fold increase in parameters (111M additional weights) to secure a 1.08% improvement. Conversely, the Lite U-Net family showed an inverse relationship at increased depths, with performance decreasing from 71.58% to 70.52%. This suggests that for lightweight models, excessive depth may lead to optimisation difficulties or over-parameterisation relative to the feature complexity of aerial imagery.

The Lite U-Net (Level 4) stands out as the most efficient configuration, outperforming the MST by 7% *mIoU* while being approximately 12 times smaller. This makes it a compelling candidate for real-time CVGL applications where inference latency and memory overhead are critical constraints.

4.2 Qualitative Analysis

Visual inspection of the segmentation masks (Figure 2) confirms the quantitative findings. The MST outputs exhibit significant fragmented noise and a lack of structural coherence, particularly at the edges of buildings. These “salt-and-pepper” artifacts would likely introduce ambiguity if used to guide a downstream localisation model.

In contrast, the U-Net architectures produce much cleaner, contiguous segments. The standard U-Net and U-Net++ excel at delineating sharp transitions between *Buildings* (red) and *Roads* (grey), preserving the rectangular geometry of urban blocks. While the 5-level models capture finer details through their expanded receptive fields, the 4-level variants already provide the “homography-like” stability necessary for effective cross-view matching. These results indicate that dedicated convolutional encoder-decoders are superior for providing the reliable semantic priors required for high-accuracy localisation.

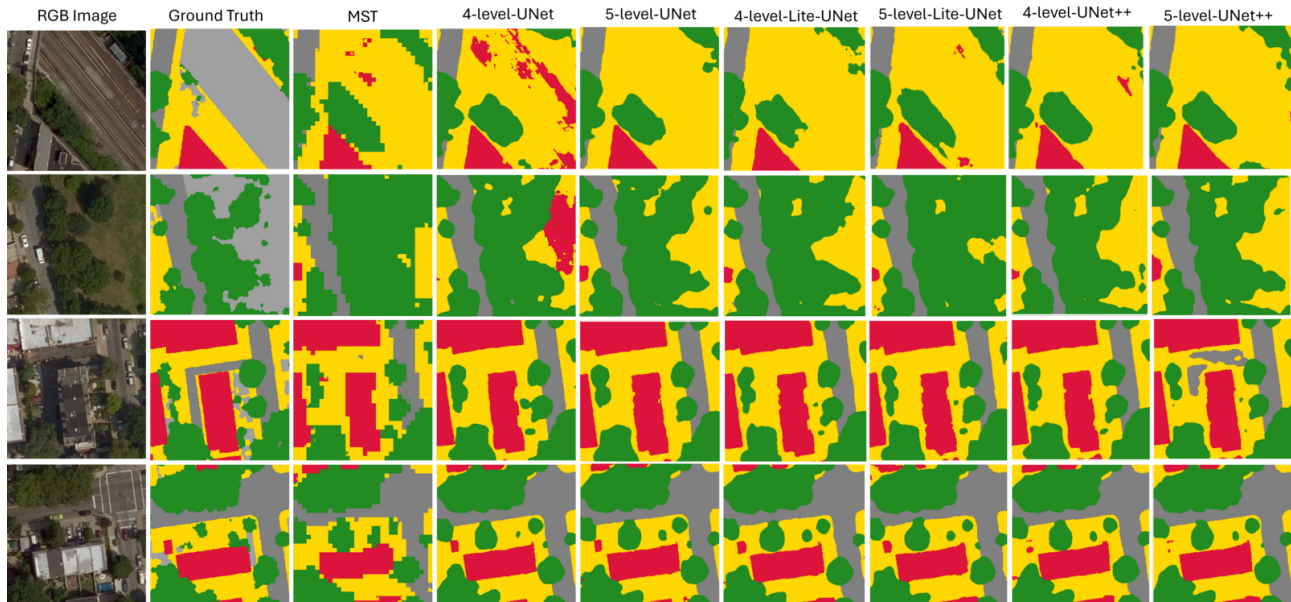


Figure 2: Qualitative comparison on Queens validation samples. Columns: RGB image, Ground Truth, MST, U-Net (L4), U-Net (L5), Lite U-Net (L4), Lite U-Net (L5), U-Net++ (L4), and U-Net++ (L5). Colors: Green (Tree Canopy), Red (Buildings), Grey (Roads), Yellow (Other Impervious).

5 Conclusion

This study establishes that aerial semantic segmentation, as a foundational component for cross-view geolocalisation, can be significantly improved through architectural isolation and the application of convolutional inductive biases. By benchmarking the Multi-Scale Transformer (MST) against various U-Net family models, we demonstrate that standard convolutional encoder-decoder structures provide a more robust basis for urban land-cover mapping. The 5-level U-Net achieved the highest overall fidelity with 74.17% mIoU, while the 4-level Lite U-Net emerged as a highly efficient alternative, delivering competitive accuracy with a significantly reduced parameter footprint.

These findings suggest that while transformers are effective for high-level spatial reasoning in CVGL pipelines, they are outperformed by convolutional models in the specific task of dense land-cover prediction. This study has several implications for the design of future localisation systems: (i) it demonstrates that high-fidelity masks of stable landmarks like buildings and roads can be achieved with relatively low computational overhead, and (ii) it confirms that increasing model depth yields diminishing returns for this specific urban dataset.

Future research will integrate these optimised convolutional segmentation backbones back into unified transformer-based CVGL pipelines. By providing high-fidelity semantic guidance, we aim to investigate the degree to which improved segmentation accuracy directly translates into reduced spatial ambiguity and enhanced localisation precision across diverse and challenging urban environments.

References

- [Castaldo et al., 2015] Castaldo, F., Zamir, A., Angst, R., Palmieri, F., and Savarese, S. (2015). Semantic cross-view matching. In *Proceedings of the 2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, ICCVW '15, page 1044–1052, USA. IEEE Computer Society.
- [Fan et al., 2021] Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., and Feichtenhofer, C. (2021). Multiscale vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6824–6835.
- [Li et al., 2024] Li, B., Zhang, Y., Ren, Y., Zhang, C., and Yin, B. (2024). Lite-unet: A lightweight and efficient network for cell localization. *Engineering Applications of Artificial Intelligence*, 129:107634.
- [Mao et al., 2023] Mao, A., Mohri, M., and Zhong, Y. (2023). Cross-entropy loss functions: Theoretical analysis and applications. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J., editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 23803–23828. PMLR.
- [Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.
- [Sudre et al., 2017] Sudre, C., Li, W., Vercauteren, T., Ourselin, S., and Cardoso, M. (2017). Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10553 LNCS:240–248.
- [Workman et al., 2022] Workman, S., Rafique, M. U., Blanton, H., and Jacobs, N. (2022). Revisiting near/remote sensing with geospatial attention. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Acceptance rate: 25.33%.

- [Workman et al., 2017] Workman, S., Zhai, M., Crandall, D., and Jacobs, N. (2017). A unified model for near and remote sensing. In *IEEE International Conference on Computer Vision (ICCV)*. Acceptance rate: 28.9%.
- [Xavier et al., 2025] Xavier, N. A., Shiguemori, E. H., Maximo, M. R., and Shah, M. (2025). A guided approach for cross-view geolocalization estimation with land cover semantic segmentation. *Biomimetic Intelligence and Robotics*, 5(2):100208.
- [Yao et al., 2024] Yao, F., Lan, C., Wang, L., Wan, H., Gao, T., and Wei, Z. (2024). Gnss-denied geolocalization of uavs using terrain-weighted constraint optimization. *International Journal of Applied Earth Observation and Geoinformation*, 135:104277.
- [Yue et al., 2025] Yue, Z., Sun, C., Zhang, X., Tang, C., Gao, Y., and Li, K. (2025). A semantic segmentation-based gnss signal occlusion detection and optimization method. *Remote Sensing*, 17(15).
- [Zheng et al., 2021] Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P. H., and Zhang, L. (2021). Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6881–6890.
- [Zhou et al., 2018] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., and Liang, J. (2018). Unet++: A nested u-net architecture for medical image segmentation.

Transformers Cross-Attention Audio-Visual Fusion

Lucas Hindes, Nazmul Siddique, Philip Vance, Sonya Coleman

School of Computing, Engineering and Intelligent Systems, Ulster University, Northern Ireland

Abstract

Multimodal learning has become increasingly important within modern artificial intelligence systems, particularly in large multimodal models that jointly process visual, textual, and audio information. Recent multimodal architectures increasingly employ pre-trained modality-specific encoders to extract contextual latent representations prior to fusion, enabling richer preservation of acoustic and temporal information within speech signals. This work proposes a transformer-based audio-visual fusion pipeline for multimodal object recognition using the SpokenCOCO dataset. Pre-trained Vision Transformer and Whisper encoders were employed to extract modality-specific visual and audio embeddings, which were subsequently fused using a bidirectional cross-attention mechanism. The proposed architecture utilised frozen pre-trained encoders alongside lightweight downstream classifiers to evaluate the effectiveness of multimodal contextual fusion. Experimental evaluation demonstrated that multimodal fusion consistently outperformed single-modality representations across all evaluation metrics. These findings demonstrate the effectiveness of direct audio-visual fusion for multimodal contextual understanding and provide a foundation for future transformer-based multimodal grounding and object recognition systems.

Keywords: computer vision, machine learning, object classification, audiovisual and multimedia systems

1 Introduction

Transformer-based architectures have become the foundation of modern artificial intelligence (AI) systems, such as Large Language Models (LLMs) and Large Multimodal Models (LMMs). Early transformer-based language models such as BERT [Devlin et al., 2019] and GPT3 [Brown et al., 2020] demonstrated that transformer architectures can model long-range contextual relationships through self-attention mechanisms [Vaswani et al., 2017]. More recently, transformers have been extended beyond single-modality language processing towards multimodal learning systems capable of jointly processing text, images, audio and speech [Wu et al., 2024]. Several recent multimodal transformer architectures have demonstrated the effectiveness of combining multiple modalities for contextual understanding and generative tasks. Flamingo [Alayrac et al., 2022] introduced a visual-language model capable of conditioning Chinchilla language models [Hoffmann et al., 2022] on visual inputs, while ViCA [Liu et al., 2026] employed both self-attention and cross-attention mechanisms for visual-text reasoning. Beyond visual-language learning, transformer-based multimodal systems have also been applied to audio-visual tasks. Llama-AVSR [Cappellazzo et al., 2025] proposed an audio-visual speech recognition framework using LMMs while Macaw-LLM [Lyu et al., 2023] demonstrated text generation using images, audio and text inputs. These architectures highlight the increasing importance of multimodal learning within modern AI systems. Multimodal learning aims to model interactions between different data modalities in a manner similar to how humans integrate sensory information to develop contextual awareness of their environment [Shu et al., 2025]. By jointly processing multiple modalities, AI systems can exploit complementary information between data sources, allowing one modality to compensate when another is noisy, ambiguous, or incomplete. In audio-visual learning, combining speech and visual information can improve contextual understanding and robustness within recognition tasks [Ye et al., 2024]. To combine information from multiple modalities, multimodal systems employ fusion methods at different stages of the learning pipeline. Feature fusion combines modality representation after feature extraction but prior to classification [Li and Tang, 2026] while output fusion combines

predictions or outputs after classification [Li and Tang, 2026]. Common multimodal fusion methods include concatenation, [Gao et al., 2020] linear, [Li et al., 2017, Wu and Han, 2018] non-linear [Charte et al., 2018] and attention-based fusion approaches, such as cross-attention [Qian et al., 2023]. Among these approaches, transformer-based cross-attention mechanisms have become increasingly popular due to their ability to model contextual relationships between modalities. Recent multimodal architectures increasingly employ pre-trained modality-specific encoders to extract contextual latent representations prior to fusion [Wu et al., 2024]. Within audio-visual learning, transformer-based speech encoders provide temporally contextualised audio embeddings that preserve acoustic and prosodic information beyond textual transcriptions alone [Shu et al., 2025]. Similarly, transformer-based visual encoders produce contextual patch-level representation capable of modelling spatial relationships within images. These representations provide rich modality-specific contextual information that can be subsequently fused for downstream multimodal learning tasks.

This work investigates transformer-based audio-visual fusion using speech and visual embeddings extracted from Whisper and Vision Transformer pre-trained encoders. The proposed approach employs bidirectional cross-attention mechanisms to model contextual interactions between audio and visual modalities prior to fusion. Early feature-level fusion is performed by combining contextualised modality representations after bidirectional cross-attention, enabling the model to jointly learn relationships between speech and visual information. This paper contributes: a transformer-based multimodal fusion pipeline combining pre-trained ViT and Whisper encoders using cross-attention fusion and an evaluation of frozen pre-trained encoders with lightweight downstream classifiers.

2 Datasets

Many widely used computer vision datasets are single-modality or text-image based rather than speech-image based. Examples include the Large Vocabulary Instance Segmentation (LVIS) dataset [Gupta et al., 2019], which contains images for large-scale object recognition, and the GQA dataset [Hudson and Manning, 2019], which provides image-question-answer pairs for visual reasoning tasks. To enable speech modality learning, the dataset would either contain recorded speech, which can be expensive and time-consuming to create, or use Text-to-Speech (TTS) models to generate synthetic audio using the text captions from pre-existing text-image datasets such as SpeechCOCO [Havard et al., 2017] or how [Yang et al., 2025] created synthesised speech data from GQA textual questions and answers, using TTS model SpeechT5 [Ao et al., 2022]. For this work, datasets containing real human speech were preferred over synthetically generated speech datasets. Although TTS-generated datasets provide a scalable method for introducing the speech modality, synthetic speech often lacks speaker diversity or natural variation, unlike human speech [Hsu et al., 2021]. Based on these selection criteria, the following datasets were selected for use in this research.

MSCOCO [Lin et al., 2014] is an image-caption dataset containing 123,287 images and 605,495 captions, where each image is paired with 5 human-written captions. The dataset contains annotations for 80 object categories, with multiple object instances potentially present within a single image, as seen in Figure 1. The dataset additionally provides bounding box annotations for object instances present within the images.

SpokenCOCO [Hsu et al., 2021] is an additional dataset that contains spoken audio recordings corresponding to the textual captions from MSCOCO. These speech samples were collected through Amazon Mechanical Turk by displaying the caption text for the speakers to read out loud. The dataset contains recordings from 2353 speakers, primarily consisting of American English speakers, and has approximately 742 hours of spoken audio. The spoken captions are aligned with their corresponding MSCOCO images and textual captions through shared image identifiers provided within the dataset metadata. A challenge within both datasets is the severe class imbalance across object categories. For example, the *hair dryer* category contains only 198 images, whereas the *person* category contains 66,808 images. To enable computationally efficient experimentation while reducing the effects of severe class imbalance, a balanced subset of SpokenCOCO was constructed. The balancing process involved randomly selecting image-audio pairs containing at least one instance of a target class until each category contained 1000 training samples, 500 validation samples, and 500 testing samples. As MSCOCO images may contain multiple object categories simultaneously, selected samples were not restricted to

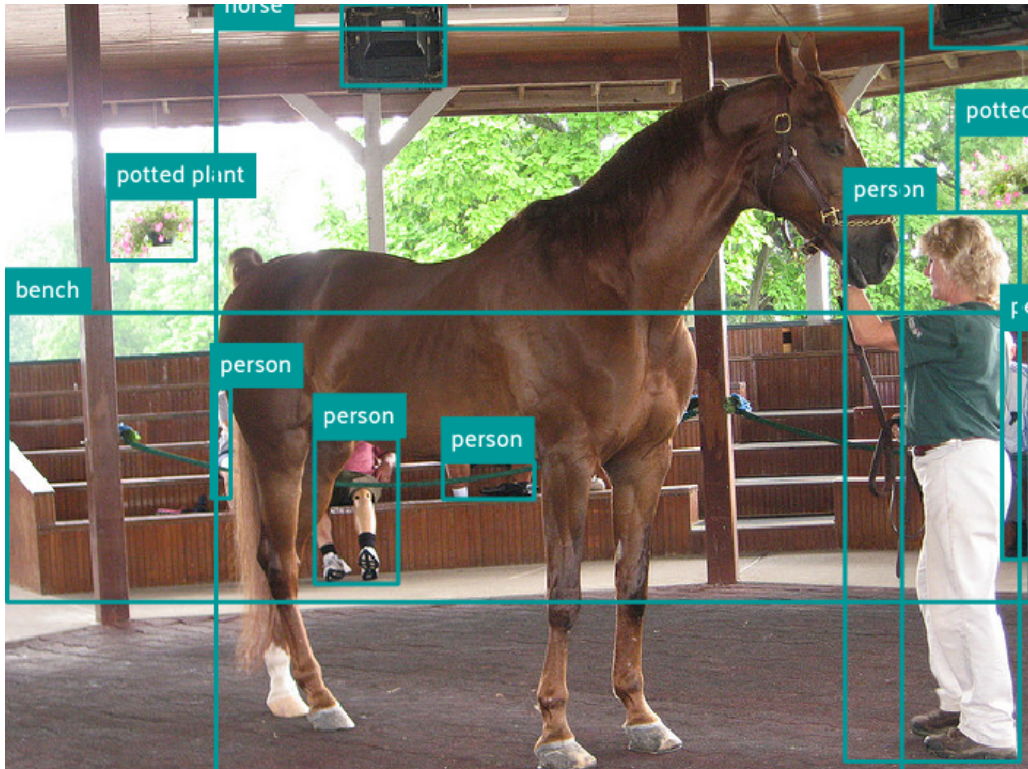


Figure 1: Example image from the MSCOCO dataset with its associated caption: "A woman interacts with a horse in a stable."

single-label scenes. 8 object classes were randomly selected from categories containing at least 2000 available samples within the original dataset. The selected object classes were *person*, *backpack*, *airplane*, *skis*, *kite*, *baseball glove*, *TV*, and *clock*. Classes with insufficient representation were excluded to ensure consistent sampling across training, validation, and testing splits. This process resulted in a balanced subset containing 16,000 paired image-audio pair samples across training, validation, and testing splits. The balanced subset provides a controlled experimental setting for evaluating transformer-based audio-visual fusion while reducing the influence of severe class imbalance present in the original datasets. The key statistics of the original COCO datasets and the balanced subset are summarised in Table 1.

Dataset	Images	Text Captions	Audio Captions	Classes	Speakers	Class Balance
MSCOCO	123,287	605,495	–	80	–	Imbalanced
SpokenCOCO	123,287	605,495	592,187	80	2,353	Imbalanced
Balanced Subset (Ours)	16,000	16,000	16,000	8	–	Balanced

Table 1: Summary of MS-COCO, Spoken-COCO and balanced subset datasets.

3 Methodology

The methodology in this work will cover multimodal audio-visual learning and classification. Several key stages include modality-specific feature extraction, bidirectional cross-attention between extracted audio and visual representations, contextualised feature fusion, and downstream classification. An overview of the proposed methodology is shown in Figure 2.

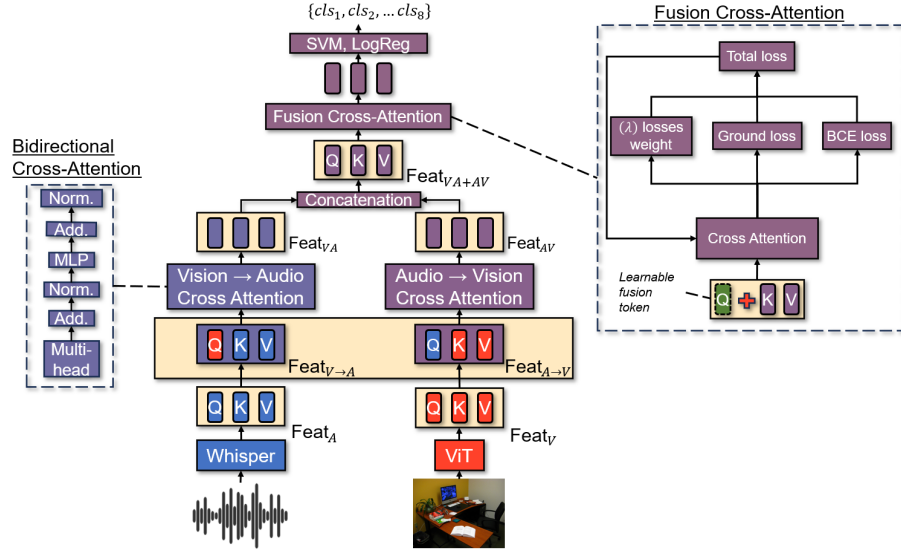


Figure 2: Overview of the proposed audio-visual fusion pipeline. Speech and image inputs are encoded using Whisper and ViT encoders to obtain modality-specific features (audio=A, visual=V). Bidirectional cross-attention modules perform audio-to-vision and vision-to-audio contextualisation ($\rightarrow$). The contextualised features are concatenated (+) and processed by a second fusion cross-attention module to produce a joint multimodal embedding for object classification. During training, classification and grounding losses are jointly optimised.

Audio-Visual Feature Extraction To obtain the representations used for multimodal fusion, pre-trained transformer-based encoders were employed for both the visual and audio modalities. Separate modality-specific encoders were used to preserve modality-specific feature representations prior to fusion.

Vision Transformer (ViT) [Dosovitskiy et al., 2021] was employed as a visual modality encoder. The model was pre-trained on ImageNet-22k [Ridnik et al., 2021]. Input images were processed and resized using Hugging Face `AutoImageProcessor` to match the ViT image size requirement of 224×224 resolution. After preprocessing, images were partitioned into non-overlapping 16×16 patches to produce $14 \times 14 = 196$ patch tokens with an additional learnable classification token appended, resulting in a total sequence length of 197 tokens. The image patches produced are flattened and linearly projected into an embedding vector before the positional embeddings are added to preserve spatial information. Embedded patch tokens are processed through the transformer encoder using self-attention mechanisms.

Whisper [Radford et al., 2023] was utilised as the audio modality encoder. Whisper is a transformer-based speech model pre-trained on its original dataset of 680k hours of labelled multilingual speech data. The "small" model variant was selected based on hardware constraints compared to other models and is a compromise of computational size and representational performance. Audio waveforms were first converted to floating-point tensors and resampled to 16 kHz, matching the input sampling rate expected by Whisper. The Hugging Face `WhisperProcessor` was then used to convert the raw audio waveforms into log-Mel spectrogram input features. During batch processing, padding was applied to produce fixed-size inputs, while longer audio sequences were truncated. Processed input features were passed through the Whisper encoder, and the final hidden states were extracted as the audio feature representation. Similar to ViT, the Whisper encoder applies self-attention to model contextual relationships across temporal audio embeddings. However, rather than modelling spatial relationships between image patches, Whisper captures contextual dependencies between speech representations across the audio sequence.

Both transformer encoders utilise the self-attention mechanism defined as follows [Vaswani et al., 2017]:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

where Q , K , and V represent the query, key, and value matrices, respectively, and d_k denotes the dimensionality

of the key vectors used for scaling. The Q matrix represents the contextual information each token seeks, the K matrix represents the semantic information contained within each token, and the V matrix contains the contextual representations that will be aggregated after attention weighting. The attention operation computes similarity scores between tokens, enabling the model to selectively aggregate contextual information across the input sequence. Following feature extraction, the pipeline produces two sequences of contextual embeddings: visual patch embeddings from ViT and temporal audio embeddings from Whisper. These embeddings are feature matrices, X_a and X_v for audio and visual matrices, respectively, as seen in Figure 2.

Both encoders were frozen to prevent updates to the pre-trained weights while reducing the training complexity of the overall pipeline and retaining the strong generalised representations learned from pretraining these models [Cappellazzo et al., 2025]. Each encoder produces embeddings with a feature dimensionality of 768, allowing compatibility during the cross-modal fusion process.

Bidirectional Cross-Attention To enable interaction between the extracted X_a and X_v matrices, bidirectional cross-attention fusion is used [Gao et al., 2024]. The purpose of this fusion stage is to incorporate contextual information from one modality matrix into the other, creating feature representations which are jointly informed of both speech and visual content. Unlike self-attention seen in the feature extraction using the transformer-based encoders, cross-attention allows tokens from one modality to attend to the tokens in the other modality. For this pipeline, two cross-attention mechanisms are used bidirectionally: Audio-to-Visual (A2V) – audio features attend to visual features; Visual-to-Audio (V2A) – visual features attend to audio features.

Cross-attention enables both modalities to become mutually context-aware. The audio representation emphasises visually relevant information, while the visual representations can incorporate contextual cues derived from speech. For further explanation, take the A2V cross-attention operation, where X_a is projected into query representations while X_v is projected into the key and value representations. The audio embeddings determine which visual tokens should receive attention. The resulting representation produces audio-informed visual contextualisation by allowing audio embeddings to selectively attend to semantically relevant visual regions. The attention operation for the i -th attention head is defined as [Vaswani et al., 2017]:

$$\text{head}_i = \text{softmax} \left(\frac{(X_a W_i^Q)(X_v W_i^K)^T}{\sqrt{d_k}} \right) (X_v W_i^V) \quad (2)$$

where W_i^Q , W_i^K , W_i^V denote the learnable projections for the query, key, and value representations of the i -th attention head, respectively, and d_k represents the dimensionality of the key vectors for scaling. The softmax operation normalises the similarity scores between audio queries and visual keys.

The cross-attention mechanisms employ multi-head attention with 8 attention heads, consistent with the original attention mechanism [Vaswani et al., 2017]. Following the cross-attention stage, multi-layer perceptron (MLP) layers apply non-linear transformation to the contextualised feature embeddings produced from the A2V and V2A mechanisms to refine the modality-enhanced features and learn higher-level feature interactions within each modality. Each MLP expands the embedding dimension from 768 to 3072 using an expansion ratio of 4, applies a Gaussian Error Linear Unit (GELU) activation function, and then projects the representation back to 768 dimensions.

Fusion Cross-Attention The A2V and V2A embeddings are concatenated along the feature dimension to form a joint multimodal representation [Cappellazzo et al., 2025]. This representation is processed using a secondary cross-attention module sharing the same architectural configuration as the bidirectional fusion modules. A learnable fusion token is introduced to aggregate contextual information from both modalities into a unified representation. This token acts as the query representation, while the concatenated embeddings provide the corresponding key and value representations. Through attention, the fusion token attends to the audio and visual information to create the final fused embedding.

To train the fusion module for multilabel object presence prediction, binary cross-entropy with logits was employed. The loss function enables independent probability estimation for each object category within the

multilabel prediction task. Binary cross-entropy with logits is defined as:

$$\mathcal{L}_{\text{BCE}}(\ell, y) = -[y \cdot \log(\sigma(\ell)) + (1 - y) \cdot \log(1 - \sigma(\ell))] \quad (3)$$

where ℓ represents the predicted logits, y denotes the ground truth label and $\sigma(\cdot)$ is the sigmoid activation function. While the SpokenCOCO subset is balanced for this paper, the original dataset is not balanced; a weighted version of the formula was used:

$$\mathcal{L}_{\text{BCE}} = -[w^+ \cdot y \cdot \log(\sigma(\ell)) + (1 - y) \cdot \log(1 - \sigma(\ell))] \quad (4)$$

where w^+ represents the positive class weighting factor. This weighting formulation is particularly relevant for scaling the proposed pipeline to the full SpokenCOCO dataset for future work, where greater class imbalance is present.

Grounding supervision was incorporated to encourage alignment between spoken object references and visually relevant regions. Grounding loss was applied to the A2V attention maps to encourage alignment between attended visual regions and the corresponding object annotations. The grounding formula is represented as:

$$\mathcal{L}_{\text{ground}} = f_{\text{ground}}(\text{Attn}_{a \rightarrow v}, \mathcal{B}) \quad (5)$$

where $\text{Attn}_{a \rightarrow v}$ denotes the audio-to-visual attention maps and $\mathcal{B}$ represents the corresponding grounding annotations.

The final training objective equation combines both losses:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ground}} + \lambda_{\text{joint}} \cdot \mathcal{L}_{\text{BCE}} \quad (6)$$

where λ_{joint} controls the relative contribution of the classification objective. This weighting factor prevents either loss component from dominating optimisation during training and enables balanced learning between localisation and semantic prediction tasks. Training was performed using the AdamW optimiser with a learning rate of 3×10^{-5} , weight decay of 0.1, batch size of 8, and 15 epochs. All experiments were conducted using an NVIDIA GeForce GTX 1080 GPU. The output representation of the fusion token was used as the final multimodal embedding for downstream classification.

Classifiers The fused multimodal embeddings were evaluated using two downstream classifiers: Support Vector Machines (SVMs) [Hearst et al., 1998] and Logistic Regression [Sperandei, 2014]. These classifiers were selected due to their effectiveness on high-dimensional embedding representations and their common use for evaluating pre-trained feature spaces without requiring end-to-end fine-tuning. Logistic Regression provides probabilistic linear classification, while SVMs maximise the separating margin between classes within the learned embedding space.

4 Results

To evaluate the proposed pipeline, the audio and visual modalities were first assessed independently to establish baseline performance. The fusion module was then evaluated in two configurations: an untrained configuration and a trained configuration using the hyperparameters selected in the Fusion Cross-Attention section. This comparison enabled evaluation of the impact of training the multimodal fusion module on overall performance.

Table 2 presents the classification performance of the proposed pipeline across single-modality and multimodal fusion configurations. The experiments compare audio-only embeddings, visual-only embeddings, multimodal untrained fusion, and multimodal trained fusion using the optimal hyperparameters obtained through grid search. Performance was evaluated using Micro F1, Macro F1, and mean Average Precision (mAP). Micro F1 measures overall classification performance using the global counts of true positives, false positives, and false negatives, while Macro F1 computes the unweighted average of per-class F1 scores, providing greater sensitivity to class-level performance differences. mAP evaluates ranking quality across classes by combining

precision and recall characteristics. A 5-fold Group KFold cross-validation strategy was used to reduce data leakage between training and validation splits by ensuring that samples associated with the same image were not simultaneously present in both partitions [Raschka, 2018].

The results demonstrate that multimodal fusion substantially outperforms single-modality representations across all evaluation metrics. Visual-only embeddings consistently achieved stronger performance than audio-only embeddings, indicating that the visual modality contains more discriminative information for the selected object categories. However, incorporating audio representations through cross-attention fusion further improved classification performance, demonstrating that speech-derived contextual cues provide complementary information beyond visual features alone.

Configuration	Classifiers	Micro F1	Macro F1	mAP
Audio-only	SVM	$73.912 \pm 0.315\%$	$70.180 \pm 0.367\%$	$77.524 \pm 0.631\%$
	Logistic Regression	$75.087 \pm 0.421\%$	$72.021 \pm 0.494\%$	$79.345 \pm 0.576\%$
Image-only	SVM	$84.496 \pm 0.103\%$	$82.592 \pm 0.249\%$	$88.744 \pm 0.215\%$
	Logistic Regression	$85.151 \pm 0.259\%$	$83.626 \pm 0.419\%$	$89.665 \pm 0.263\%$
Untrained Fusion	SVM	$89.574 \pm 0.386\%$	$88.709 \pm 0.567\%$	$92.724 \pm 0.809\%$
	Logistic Regression	$90.460 \pm 0.380\%$	$89.897 \pm 0.486\%$	$93.994 \pm 0.725\%$
Trained Fusion	SVM	$92.110 \pm 0.456\%$	$91.279 \pm 0.620\%$	$93.847 \pm 0.735\%$
	Logistic Regression	$92.160 \pm 0.500\%$	$91.380 \pm 0.650\%$	$94.131 \pm 0.660\%$

Table 2: Classification performance of single-modality and multimodal fusion configurations evaluated using Micro F1, Macro F1, and mAP.

Notably, even the untrained fusion configuration outperformed both individual modalities, suggesting that cross-modal contextual integration provides complementary information prior to optimisation of the fusion module. The relatively strong performance of the untrained fusion configuration suggests that the pre-trained transformer encoders already produce highly transferable modality-specific representations. Optimising the fusion module further improves cross-modal alignment and contextual integration, leading to the highest overall classification performance. Logistic regression achieved marginally higher performance than SVM across most configurations, suggesting that the fused embedding space produced by the transformer-based encoders was sufficiently linearly separable for lightweight probabilistic classification.

5 Conclusion and Future Work

The experimental results demonstrate that the proposed cross-attention fusion pipeline improves classification performance compared to both unimodal representations and multimodal untrained fusion configurations. The controlled, balanced subset enabled clearer evaluation of modality fusion behaviour while reducing the influence of severe class imbalance present in the original SpokenCOCO dataset. Experimental evaluation demonstrated that multimodal fusion consistently outperformed single-modality configurations across all evaluation metrics. The results further showed that integrating speech-derived contextual information with visual representations improved classification performance beyond visual-only embeddings, highlighting the complementary nature of audio-visual feature fusion. Additionally, the strong performance of the untrained fusion configuration suggests that pre-trained transformer encoders provide highly transferable modality-specific representations prior to fusion optimisation. Training the fusion module further improved cross-modal contextual alignment and overall classification performance. However, the use of a balanced subset limits direct comparison with state-of-the-art (SOTA) methods evaluated on the full SpokenCOCO dataset, where substantial class imbalance can significantly affect performance metrics. Future work will evaluate both the proposed pipeline and comparable SOTA architectures under full-dataset conditions to enable fairer comparative analysis. Further development will additionally explore end-to-end fine-tuning of the modality encoders, larger multimodal datasets, and more advanced fusion strategies for improving cross-modal grounding and contextual alignment.

References

- [Alayrac et al., 2022] Alayrac, J.-B., Donahue, J., Luc, P., et al. (2022). Flamingo: A visual language model for few-shot learning. *NeurIPS*, 35:23716–23736.
- [Ao et al., 2022] Ao, J. et al. (2022). SpeechT5: Unified-modal encoder-decoder pre-training for spoken language processing. In *ACL*, pages 5723–5738.
- [Brown et al., 2020] Brown, T. et al. (2020). Language models are few-shot learners. In *NeurIPS*, volume 33, pages 1877–1901.
- [Cappellazzo et al., 2025] Cappellazzo, U. et al. (2025). Large language models are strong audio-visual speech recognition learners. In *ICASSP*, pages 1–5.
- [Charte et al., 2018] Charte, D., Charte, F., García, S., del Jesus, M. J., and Herrera, F. (2018). A practical tutorial on autoencoders for nonlinear feature fusion: Taxonomy, models, software and guidelines. *Information Fusion*, 44:78–96.
- [Devlin et al., 2019] Devlin, J. et al. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186.
- [Dosovitskiy et al., 2021] Dosovitskiy, A. et al. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, pages 611–631.
- [Gao et al., 2020] Gao, J., Li, P., Chen, Z., and Zhang, J. (2020). A survey on deep learning for multimodal data fusion. *Neural Computation*, 32(5):829–864.
- [Gao et al., 2024] Gao, S. et al. (2024). Avsegformer: Audio-visual segmentation with transformer. In *AAAI*, volume 38, pages 12155–12163.
- [Gupta et al., 2019] Gupta, A. et al. (2019). LVIS: A dataset for large vocabulary instance segmentation. In *CVPR*, pages 5351–5359.
- [Havard et al., 2017] Havard, W., Besacier, L., and Rosec, O. (2017). SPEECH-COCO: 600k visually grounded spoken captions aligned to MSCOCO data set. In *GLU 2017 International Workshop on Grounding Language Understanding*, glu_2017, page 42–46.
- [Hearst et al., 1998] Hearst, M. et al. (1998). Support vector machines. *IEEE Intelligent Systems and their Applications*, 13(4):18–28.
- [Hoffmann et al., 2022] Hoffmann, J. et al. (2022). An empirical analysis of compute-optimal large language model training. *NeurIPS*, 35:30016–30030.
- [Hsu et al., 2021] Hsu, W.-N. et al. (2021). Text-free image-to-speech synthesis using learned segmental units. In *ACL-IJCNLP*, pages 5284–5300.
- [Hudson and Manning, 2019] Hudson, D. A. and Manning, C. D. (2019). GQA: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, pages 6693–6702.
- [Li and Tang, 2026] Li, S. and Tang, H. (2026). Multimodal alignment and fusion: A survey. *International Journal of Computer Vision*, 134(3):103.
- [Li et al., 2017] Li, Z., Yang, L., and Zhou, F. (2017). FSSD: Feature fusion single shot multibox detector. *arXiv preprint arXiv:1712.00960*.
- [Lin et al., 2014] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In *ECCV*, pages 740–755.
- [Liu et al., 2026] Liu, W. et al. (2026). ViCA: Efficient multimodal LLMs with vision-only cross-attention. *arXiv preprint arXiv:2602.07574*.
- [Lyu et al., 2023] Lyu, C. et al. (2023). Macaw-LLM: Multi-modal language modeling with image, audio, video, and text integration. *arXiv preprint arXiv:2306.09093*.
- [Qian et al., 2023] Qian, X., Wang, Z., Wang, J., Guan, G., and Li, H. (2023). Audio-visual cross-attention network for robotic speaker tracking. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:550–562.
- [Radford et al., 2023] Radford, A. et al. (2023). Robust speech recognition via large-scale weak supervision. In *ICML*, volume 202, pages 28492–28518.
- [Raschka, 2018] Raschka, S. (2018). Model evaluation, model selection, and algorithm selection in machine learning. *arXiv preprint arXiv:1811.12808*.
- [Ridnik et al., 2021] Ridnik, T., Ben-Baruch, E., Noy, A., and Zelnik-Manor, L. (2021). ImageNet-21K Pretraining for the Masses. In *NeurIPS Datasets and Benchmarks*.
- [Shu et al., 2025] Shu, F., Zhang, L., et al. (2025). Audio-visual LLM for video understanding. In *ICCV Workshops*, pages 4305–4314.
- [Sperandei, 2014] Sperandei, S. (2014). Understanding logistic regression analysis. *Biochemia medica*, 24(1):12–18.
- [Vaswani et al., 2017] Vaswani, A. et al. (2017). Attention Is All You Need. In *NeurIPS*, volume 30, pages 5998–6008.
- [Wu and Han, 2018] Wu, A. and Han, Y. (2018). Multi-modal circulant fusion for video-to-language and backward. In *IJCAI*, pages 1029–1035.
- [Wu et al., 2024] Wu, S. et al. (2024). NExT-GPT: Any-to-any multimodal LLM. In *ICML*, volume 235, pages 53366–53397.
- [Yang et al., 2025] Yang, W., Wei, J., Lu, W., and Li, L. (2025). You only speak once to see. In *ICASSP*, pages 1–5.
- [Ye et al., 2024] Ye, Q. et al. (2024). Cat: Enhancing multimodal large language model to answer questions in dynamic audio-visual scenarios. In *ECCV*, pages 146–164.

WeldNet: Vision-Based Weld Inspection for Intelligent Manufacturing Systems

Sushil Sharma¹, Oran Connors^{1,2}, Reenu Mohandas¹, Joe Brennan², and Ciarán Eising¹

¹*firstname.lastname@ul.ie* and ²*firstname.lastname@omc.ie*

¹*University of Limerick, Ireland*, ²*OMC Technologies Limited, Ireland*

Abstract

Manual weld inspection is often slow, inconsistent, and prone to subjectivity, especially in high-precision manufacturing industries. This paper presents a WeldNet: AI-based weld defect detection system using deep learning and computer vision for real-time industrial inspection. The proposed methodology detects weld defects with high accuracy while being optimised for edge deployment in manufacturing environments. Our approach improves inspection consistency, reduces production delays, and supports faster quality assurance workflows. The system also establishes a foundation for future intelligent welding systems with automated feedback and adaptive process control, contributing towards Industry 4.0 manufacturing.

Keywords: Machine Vision, Weld Inspection, Deep Learning, Industry 4.0, YOLOv8

1 Introduction

Industrial weld inspection is critical for ensuring structural integrity, safety, and product quality in advanced manufacturing systems [Torres-Torres et al., 2026]. However, existing inspection pipelines are still largely manual and rely heavily on human expertise, leading to inconsistent evaluations and limited scalability in high-throughput production environments. To address these limitations, we propose a real-time deep learning-based weld defect detection methodology leveraging computer vision techniques. The system is designed for efficient and accurate detection of weld defects from images and video streams using lightweight object detection models trained on a YOLO-formatted weld dataset. It is robust to real-world variations such as lighting changes, texture diversity, and noise. Beyond defect detection, the framework integrates automated weld logging and intelligent feedback. For each detected defect, the system provides repair suggestions and preventive guidance, enabling both quality control and operator support. Overall, this work contributes an efficient and deployable weld inspection system that improves detection accuracy, reduces manual inspection effort, and enhances traceability for industrial applications aligned with Industry 4.0 requirements.

2 Background

Weld inspection is an important process in industries such as oil and gas, pharmaceuticals, and structural manufacturing, where safety and reliability are critical. Traditionally, weld quality is assessed manually by inspectors. While widely used, this approach is time-consuming and subjective, and it can lead to inconsistent results. Recent advances in artificial intelligence and computer vision have enabled automated defect detection systems that assist and expedite operator-led weld joint inspection. In particular, deep learning-based object detection models have shown strong performance in identifying visual defects in industrial settings when trained and fine-tuned on large annotated datasets. This work develops a deep learning-based weld defect detection system that identifies defects in images and video streams. In addition to detection, the system supports basic reporting

and logging for traceability and quality tracking in manufacturing workflows. The goal is to improve inspection consistency, reduce human error, and support more efficient and reliable industrial weld quality control.

3 Methodology

This work proposes a lightweight vision-based framework for automated weld defect detection in intelligent manufacturing systems. As illustrated in Figure 1, the pipeline consists of image acquisition, preprocessing, data annotation, YOLOv8n-based model training, and evaluation. The goal is accurate defect detection with low computational cost for real-time industrial deployment.

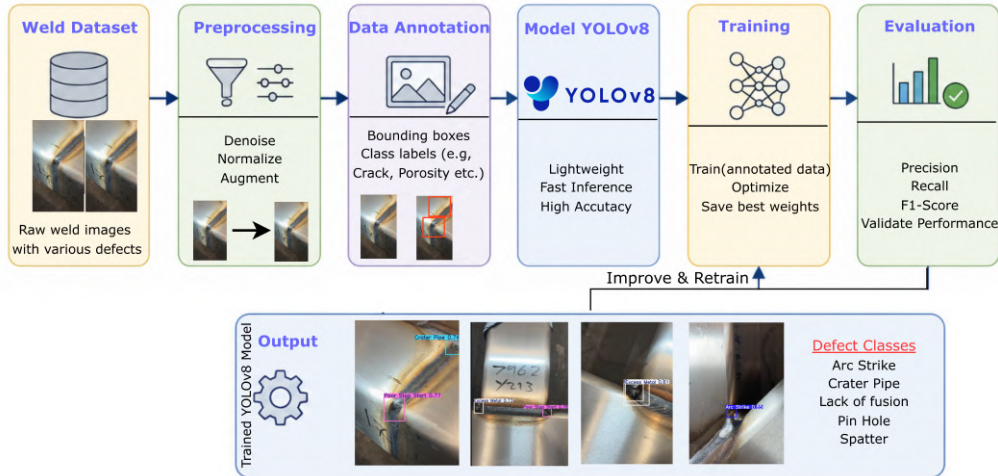


Figure 1: Proposed methodology for automated weld defect detection.

3.1 Data Preprocessing and Augmentation

Weld images containing defects such as arc strike, crater pipe, lack of fusion, pinhole, and spatter are collected during industrial processes. To improve robustness, images are normalized as

$$I_{\text{norm}} = \frac{I - \mu}{\sigma + \epsilon} \quad (1)$$

where μ and σ are dataset mean and standard deviation.

Data augmentation is applied using a stochastic transformation function:

$$I' = \mathcal{T}(I) \quad (2)$$

which includes geometric and photometric variations such as rotation, scaling, flipping, and brightness changes.

3.2 Detection Model

A YOLOv8n-based detector is used due to its efficiency and anchor-free design. The model learns a mapping:

$$f_{\theta}(I) \rightarrow (b_i, c_i, s_i)_{i=1}^N \quad (3)$$

where b_i is the bounding box, c_i is the defect class, and s_i is the confidence score.

Bounding box overlap is measured using Intersection over Union (IoU):

$$\text{IoU}(b, b^*) = \frac{|b \cap b^*|}{|b \cup b^*|} \quad (4)$$

3.3 Training Objective

The model is trained using a multi-task loss:

$$\mathcal{L} = \lambda_1 \mathcal{L} * \text{box} + \lambda_2 \mathcal{L} * \text{cls} + \lambda_3 \mathcal{L}_{\text{diff}} \quad (5)$$

where $\mathcal{L}_{\text{box}}$ is based on IoU/CIoU, $\mathcal{L}_{\text{cls}}$ is classification loss, and $\mathcal{L}_{\text{diff}}$ refines localization.

Performance is evaluated using Precision, Recall, F1-score, and mean Average Precision (mAP). During inference, the trained model predicts defect locations and categories in real time, enabling efficient weld quality inspection in industrial environments.

4 Results

This section evaluates the YOLOv8n-based weld defect detection methodology on a multi-class dataset containing Arc Strike, Crater Pipe, Lack of Fusion, Pin Hole, and Spatter defects. YOLOv8n was selected due to its strong balance between detection accuracy and real-time inference performance. Performance is assessed using Precision, Recall, F1-score, mAP@0.5, and inference speed (FPS).

4.1 Ablation Study

An ablation study is conducted to evaluate the contribution of each component in the proposed pipeline, including normalization, data augmentation, and CIoU-based regression loss. Results in Table 1 show that standard score normalization improves stability under varying illumination, while data augmentation enhances generalization. CIoU loss further improves bounding box accuracy and overall localization performance.

Table 1: Ablation study of key components in the proposed methodology.

Configuration	Precision	Recall	mAP@0.5	F1-score
Baseline (YOLOv8)	88.4	86.1	87.2	87.2
+ Standard Normalization	89.7	87.5	88.6	88.6
+ Data Augmentation	91.2	89.3	90.4	90.2
+ CIoU Loss (Full Model)	93.5	91.8	92.6	92.6

4.2 Comparison with State-of-the-Art Methods

To demonstrate the effectiveness of the proposed method, we compare it with several state-of-the-art object detection models, including Faster R-CNN [Ren et al., 2016], SSD [Liu et al., 2016], YOLOv5 [Jocher et al., 2022], and YOLOv7 [Wang et al., 2023]. All models are evaluated under the same dataset and experimental settings.

As shown in Table 2, the proposed YOLOv8n-based framework achieves the best trade-off between accuracy and inference speed. While two-stage detectors such as Faster R-CNN provide competitive accuracy, they suffer from significantly higher computational cost. In contrast, the proposed method maintains real-time performance while achieving the highest mAP.

Table 2: Comparison with state-of-the-art detection methods.

Method	Backbone	Precision	Recall	mAP@0.5	FPS
Faster R-CNN [Ren et al., 2016]	ResNet-50	91.8	90.2	90.5	12
SSD [Liu et al., 2016]	VGG-16	87.6	85.3	86.1	45
YOLOv5 [Jocher et al., 2022]	CSPDarknet	92.1	90.8	91.3	68
YOLOv7 [Wang et al., 2023]	E-ELAN	93.0	91.5	92.1	72
Proposed YOLOv8n	C2f Backbone	93.5	91.8	92.6	78

4.3 Qualitative Analysis

This section presents sample results from the welding dataset to visually evaluate the proposed method. Figure 2 shows that the YOLOv8n model successfully detects welding defects with accurate bounding boxes.

The model performs well across different conditions, including changes in lighting, weld texture, and background noise. It is also able to correctly identify multiple defect classes such as cracks, porosity, lack of fusion, and spatter. Overall, the qualitative results demonstrate that the model provides reliable and consistent defect detection, with good localization and correct class predictions, even in challenging cases.

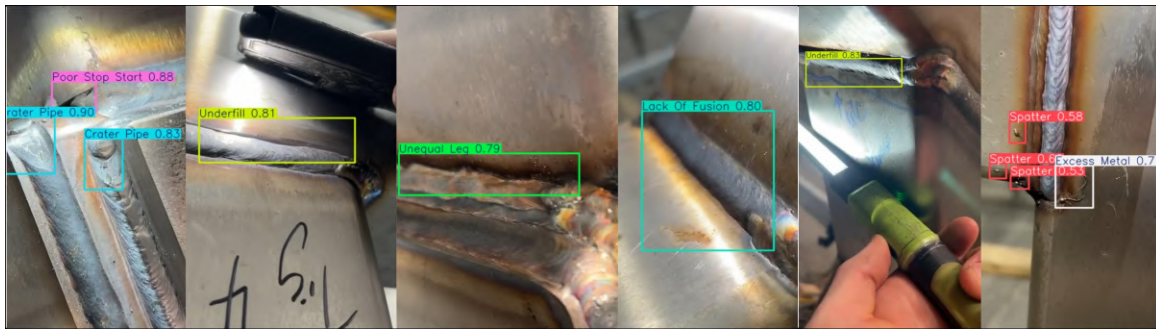


Figure 2: Qualitative analysis of the welding dataset shows promising results for defect detection. The YOLOv8n model performs effectively in identifying and detecting weld defects.

5 Conclusion & Future Work

This work demonstrates effective weld defect detection using a YOLOv8n-based methodology under standard industrial conditions. However, performance may degrade in more challenging scenarios such as low illumination, heavy noise, and partial occlusion. Future work will therefore focus on improving robustness in these conditions and optimizing the model for real-time deployment on edge and resource-constrained industrial devices. In addition, multi-modal fusion with thermal or infrared imaging will be explored to improve defect visibility.

References

- [Jocher et al., 2022] Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., Kwon, Y., Michael, K., Fang, J., Wong, C., Yifu, Z., Montes, D., et al. (2022). ultralytics/yolov5: v6. 2-yolov5 classification models, apple m1, reproducibility, clearml and deci. ai integrations. *Zenodo*.
- [Liu et al., 2016] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer.
- [Ren et al., 2016] Ren, S., He, K., Girshick, R., and Sun, J. (2016). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149.
- [Torres-Torres et al., 2026] Torres-Torres, M., Velasquez, K., Escorcia-Gutierrez, J., and Valls, A. (2026). Visual weld quality inspection system for identifying defects in gmaw joints with semantic segmentation on pre-trained models. *International Journal of Computational Intelligence Systems*, 19(1):110.
- [Wang et al., 2023] Wang, C.-Y., Bochkovskiy, A., and Liao, H.-Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7464–7475.

Enhancing an Emotion-Labelled Video Dataset with Pose Estimation Key-points for Face, Body and Hands

Ciara Sempey, Bryan Scotney, Shuai Zhang, Matthew Burns, Jorge Martinez Carracedo

Ulster University

Abstract

Research in automated emotion recognition from video has traditionally focused on facial expression as the primary indicator of emotional state. More recently, increasing attention has been given to incorporating features from body posture and gesture alongside facial expression to improve the performance of emotion recognition models. Although many emotion-labelled video datasets exist, few incorporate pose estimation data, and many of those that do are collected in lab-based environments.

To address this limitation, we introduce a new emotion-labelled pose estimation dataset derived from the Multimodal EmotionLines Dataset (MELD). This dataset combines MELD's emotion annotations with speaker-specific, detailed pose estimation key-points from face, body and hands, extracted from the corresponding video recordings. The MELD-Keypoints v0.1 release contains pose estimation data extracted from 4,059 MELD video clips and 283,093 frame-level keypoint records. To date, 5,601 video clips have been processed through the complete extraction pipeline, of which 72.5% successfully produced speaker-specific pose data.

Keywords: Emotion Recognition, Computer Vision, Pose Estimation.

1 Introduction

Automated emotion recognition is a rapidly growing area of research as seen in the increased number of papers published on the topic since 2012 [Pan et al., 2023] with many applications including health and social care, education, sentiment analysis and human computer interaction [Pan et al., 2023, Pereira et al., 2024]. In particular there is opportunity to support the care and treatment of people living with conditions such as Parkinson's disease or dementia, in a care home environment where changes in emotional state can be a symptom of the disease and a reaction to medication changes; in such cases an automated emotion recognition tool would be useful for monitoring emotional wellbeing over time [Song, 2024].

Humans express emotion through audio, visual, physiological and textual signals and correspondingly there are various methods for observing these different signals [Pereira et al., 2024]; taking inspiration from how we as humans observe emotion in others this study focuses on visual signals as expressed in video. There are many emotion-labelled video datasets available, some that have been specifically designed for that purpose [Bänziger et al., 2012, Busso et al., 2008, Miranda-Correa et al., 2021, Park et al., 2020], and others which have been created from existing material like movies or tv shows and adapted for emotion research [Poria et al., 2019, Dhall et al., 2012 and Jiang et al., 2020]. A limitation that we observed in some of the lab-based video datasets was that there was minimal furniture and often only a single participant, potentially reducing the ability of trained models to generalise to real-world setting. The datasets which used clips from film or television, are acted datasets but the settings are usually in real-world type environments, with the challenges that can bring, such as movement around and interactions with other people and objects in the environment.

The main contribution of this paper is MELD-Keypoints, a novel speaker-specific pose estimation extension of the Multimodal EmotionLines dataset (MELD). By combining OpenPose pose estimation with pyppbox-based identity tracking, the dataset augments the MELD emotion annotations with detailed key-point data for face, body

and hands, for the target speaker associated with each utterance. This provides a new resource for investigating multimodal emotion recognition in complex, real-world, multi-person environments.

2 Related Work

Historically, emotion recognition research has focused on text analysis, audio signal analysis, physiological signals, and in the context of image and video data, primarily on using facial expressions to determine emotion [Lu et al., 2024]. Data for physiological signals, such as EEG, heart rate and temperature, requires the use of a wearable device which often has to touch the skin; the user must remember to wear it and if they find it uncomfortable they may not be diligent about doing so [Cai et al., 2023]. The benefit of video data is that it can gather information for everyone onscreen including potentially valuable context information about the environment and interactions taking place with others; however issues with privacy are a concern as are the challenges of missing data that will occur as a result of occlusion as people move around others and furniture or turn away from the camera, as well as people moving in and out of the camera field of view [Geetha et al., 2024].

Whilst emotion recognition research using image or video data has mainly focused on facial expressions, there has been a shift to skeleton-based emotion recognition [Lu et al., 2024]. Humans communicate their emotions via facial expressions, and also via their body posture and hand gestures, therefore capturing this type of information has the potential to improve emotion categorisation [Lu et al., 2024]. This shift has been aided by the variety of methods to obtain skeleton data, whether by using motion capture [Busso et al., 2008], a depth-sensing camera like Kinect [Zhang, 2012 and Li et al., 2016] or applying pose estimation tools to RGB video to extract body pose key-points, such as OpenPose [Cao et al., 2021, Raaj et al., 2019, Simon et al., 2017], AlphaPose [Fang et al., 2023], MediaPipe [Bazarevsky et al., 2020].

There have been several models created to define emotions and most fall into one of two types: categorical or dimensional [Younis et al., 2024]. One of the most popular categorical models known as the Ekman model [Ekman and Friesen, 2003], focuses on six main emotions: anger, disgust, fear, happiness, sadness and surprise. This model is frequently used in emotion recognition datasets such as MELD [Poria et al., 2019], AFEW [Dhall et al., 2012] and DFEW [Jiang et al., 2020], usually with a seventh category of neutral.

Several emotion-labelled datasets including GEMEP, IEMOCAP, AMIGOS and K-EmoCon were collected in controlled laboratory environments with limited environmental complexity and participant movement [Bänziger et al., 2012, Busso et al., 2008, Miranda-Correa et al., 2021, Park et al., 2020]. In contrast, AFEW and DFEW were created from film and television content to support the research of emotion in more natural, real-world type environments [Dhall et al., 2012, Jiang et al., 2020]. MELD – Multimodal EmotionLines Datasets [Poria et al., 2019], is a similar ‘in-the-wild’ dataset which contains clips from the tv show ‘Friends’. A benefit of this dataset is that the main 6 characters appear in the majority of the videos, which will provide a large variety of emotion data for each of the 6 characters. Many clips have more than one person in the frame and there is movement of people around other people or furniture or turning away from the camera, providing instances of occlusion which is something we would expect tackle in real-world situations.

Most of these datasets are RGB video datasets and would, therefore, need to be processed through a pose estimation tool in order to extract key-point data. Two exceptions are, IEMOCAP – the Interactive Emotional Dyadic Motion Capture Database [Busso et al., 2008], which contains motion-capture data and detailed facial markers, and EMILYA [Fourati and Pelachaud, 2014] which focuses on bodily emotion expression and did not include facial expression data. Both datasets were collected in laboratory environments and do not provide the combination of natural multi-person interactions and detailed face, body and hand key-points targeted by this work.

3 Methodology

In this section we will introduce the dataset that we have chosen to augment with pose estimation key-points for face, body and hands and the tools used to accomplish this task.

3.1 MELD Dataset

For this study we selected Multimodal EmotionLines Dataset (MELD) [Poria et al., 2019], a video dataset containing 13,848 video clips from the TV show Friends. MELD is an extension of EmotionLines [Hsu et al., 2018], which was created using the scripts from ‘Friends’ which had been divided into sequences of utterances, with each utterance assigned an emotion label, using the Ekman model [Ekman and Friesen, 2003] of 7 emotion categories plus neutral and a sentiment label (positive, negative, neutral) making this dataset useful for sentiment analysis and emotion categorisation from text.

The team that created MELD identified an opportunity to enhance and extend the original EmotionLines dataset by including video and audio to enable researchers to explore emotion recognition from visual and audio signals as well as text, with potential to combine these signals for multi-modal emotion recognition. To this end the emotion labelling for each utterance was repeated, with a new set of reviewers who read the text, watched the video and listened to the audio, before making their assessment. This new set of reviewers who had the benefit of visual and audio signals did in some cases make different decisions to the prior group who relied solely on the text.

The MELD dataset contains the video clips of each utterance, which is identified by a video ID, a dialog ID, an utterance ID, the name of the person who is saying the utterance, the text of the utterance and an emotion label and sentiment label. In our augmented dataset we have used the emotion label rather than the sentiment label as for the purposes of our project, we were more interested in changes in emotion, particularly when applied to a care home setting where we would be tracking emotion changes of residents over time.

3.2 OpenPose

OpenPose is a 2D pose estimation tool which can detect up to 135 key-points for multiple people and is capable of real-time processing [Cao et al., 2021, Raaj et al., 2019, Simon et al., 2017]. It takes a bottom-up approach, unlike tools such as YOLO [Redmon et al., 2016], which first detect people in frame before finding the key-points, OpenPose first finds the key-points by using heatmaps to identify body joints (elbows, wrists, knees etc.) and then Part Affinity Fields (PAFs) to create vector maps to link the different body joints, before finally pairing joints together to create the skeleton.

We selected OpenPose because of its three comprehensive key-point models for face, body and hand which provided a significant amount of pose estimation data as input to features for emotion recognition models. We also chose it for its performance in being able to provide body pose key-points for multiple people on-screen, which is the case for many of the videos within the MELD dataset.

However, as the emotion label for each video in MELD is only connected to the person who says the utterance in that video, and there are often multiple people onscreen, pyppbox was selected to identify, track and extract only the key-points for the utterance speaker in each frame.

3.3 pyppbox

Pyppbox is an open-source tool that combines several other open-source tools within the PoseTreID framework for person detection across multiple frames. [Siv et al., 2020, Siv et al., 2022]. For the Identification module we used FaceNet, a facial recognition model, which we pre-trained to recognize the six main characters from Friends, using an image dataset from Kaggle [Kalabasi, 2022].

Due to the changes made, the tool was able to process videos and provide an output that contains the frame number, the people identified in that frame, an identity label each person, and a bounding box indicating the location of that identified person within the frame.

3.4 Creating a New Augmented Dataset

The end-to-end process used to create this extension of the MELD dataset with the existing data augmented with pose estimation key-points is illustrated in Figure 1. The MELD video dataset is processed through both OpenPose and pyppbox. OpenPose returns a folder of JSON files for each video, containing a JSON file for every frame and

hand, body and face key-points for each person it detects in the frame. Pyppbox, whose reidentification model FaceNet has been pre-trained to recognise the six main Friends characters, will return a text file for each video that contains an entry for each person it detects in each frame, an identify label for that person and bounding box and mini-frame coordinates.

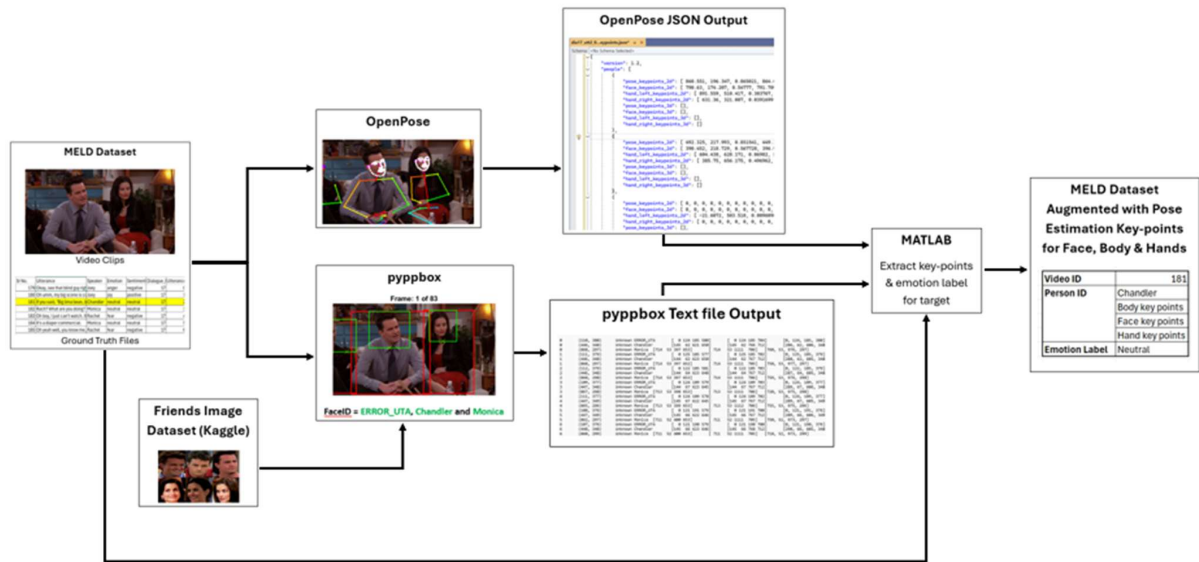


Figure 1: End-to-End Process to create new version of MELD dataset augmented with detailed pose estimation key-points

Combining the Ground Truth files from MELD, the OpenPose output JSON files and the pyppbox output text files; for each video it identifies from the MELD ground truth file the target person who has said the utterance and extract the bounding box coordinates for that person, for every frame from the relevant pyppbox output text file. Then accessing the corresponding OpenPose JSON file, for each frame, the OpenPose body key-points were used to construct a bounding box for each detected individual. These candidate bounding boxes were compared with the pyppbox bound box corresponding to the target speaker. Matching was performed using a combined score based on Intersection over Union (IoU) and the proportion of valid pose key-points located within the pyppbox bounding box. The highest-scoring OpenPose detection was selected and the corresponding face, body and hand key-points extracted. This is then collated into the new dataset that contains the video ID, the identity of the target person and the face, body and hand key-points for that person in every frame of the video and the emotion label which was also obtained from the MELD ground truth file.

4 Results

All 13,848 MELD video clips have been processed through OpenPose, with a single failure caused by a corrupted video clip. To date 5,607 video clips have been processed through pyppbox, again with the same single failure. Combining the available OpenPose and pyppbox outputs produced 5,601 successfully processed videos, from which 4,059 speaker-specific videos were added to the MELD-Keypoints v0.1 dataset, representing 72.5% of the processed videos. The remaining 1,542 videos were excluded because the target speaker was not one of the six principal characters, could not be identified by pyppbox, or due to missing data. All exclusions were recorded in tracking tables for subsequent analysis.

Split	MELD Videos	OpenPose Outputs	Pyppbox Outputs	MATLAB Processed	Videos Added to New Dataset
Dev	1,112	1,112	1,112	1,111	829
Test	2,747	2,747	1,000	1,000	732
Train	9,989	9,988	3,495	3,490	2,498
Total	13,848	13,847	5,607	5,601	4,059

Table 1: Progress of new dataset generation

The current release is MELD-Keypoints v0.1 and the link to download this version of the dataset and associated MATLAB processing framework, is publicly available through a GitHub Repository¹.

We have carried out an initial analysis of the balance of classes and character distribution of the current dataset.

Emotion	Count	%
Neutral	1832	45.10%
Joy	632	15.60%
Anger	538	13.30%
Surprise	503	12.40%
Sadness	296	7.30%
Fear	132	3.30%
Disgust	126	3.10%
Total	4059	100%

Table 2: Emotion Label Distribution

Character	Count	%
Ross	785	19.30%
Joey	728	17.90%
Phoebe	663	16.30%
Chandler	653	16.10%
Monica	636	15.70%
Rachel	594	14.60%
Total	4059	100%

Table 3: Character Distribution

5 Discussion

The current MELD-Keypoints v0.1 release demonstrates that large-scale speaker-specific key-point extraction from MELD is feasible. The extraction pipeline successfully generated pose data for 4,059 videos and 283,093 frame-level records, corresponding to 72.5% of the videos processed through the complete pipeline.

Whilst there are other emotion recognition video datasets with accompanying pose estimation data available, these have been typically lab-based with acted out emotions [Busso et al., 2008]. However MELD, whilst being an acted dataset, is set in real-world type environments like living rooms, coffee shops and offices, with people moving around furniture and other people, similar to what we would expect to observe in a true real-world dataset, such as in a care home setting. The occlusion which occurs when one person moves in front of another or turns away from the camera is something we are keen to investigate and is a benefit of this dataset, firstly because these are typical scenarios in the original MELD dataset, but also that the OpenPose has shown good performance at handling occlusions and the reidentification and tracking elements of pyppbox have ensured that even when a person turns away from the camera, if they have previously been identified, then they continue to be tracked and linked to that identify and the key-point data that is still visible can be extracted, for example we may be missing some or all of the face key-points when they turn away but we still have body and hand key-points. This gives us scope to investigate how to handle occlusion in an emotion recognition system.

OpenPose was selected for its comprehensive key-point models for face, body and hand and its ability to return results for multiple people and handle occlusions well, however when we have carried out the next stage of our study to consider which combination of features from face, body and hands works best for emotion recognition,

¹ <https://github.com/CSempey/MELD-Keypoints>

this will allow us to assess if we need this level of key-point detail and whether an alternative, less detailed pose estimation model would be sufficient.

Although the scenarios are acted the emotions expressed are recognisable, though we would anticipate that any emotion recognition models trained on this data would be fine-tuned on datasets similar to the type the models would be used on, for example, care home video data.

A major factor limiting dataset coverage is that the FaceNet model used within pypbbox was trained only on the six principal Friends characters. Consequently, pose data can only be extracted when the target speaker belongs to this group. Although as we saw in MELD-Keypoints v0.1, data for these 6 main characters has accounted for 72.5% of the data processed so far. Expanding the facial recognition model to additional recurring characters is expected to increase coverage in future releases.

Looking further ahead, once we have created a suitable emotion recognition model, as we already have OpenPose key-point data for everyone in the video, there is an opportunity use the model to expand the dataset to incorporate emotion labels for everyone onscreen, enabling further research into emotion within groups. It is even possible to combine the data for sequence of videos that belong to the same dialog, so that you could look at emotion changes during conversation or over a longer period of time than just an utterance.

6 Conclusions

In this paper we have highlighted the need for large-scale, good quality emotion-labelled video datasets augmented with pose estimation data, set in real-world type environments with multiple people interacting and moving around other people and furniture, causing occlusion, moving in and out of camera view; this would support the development of emotion recognition models that could be more readily applied to a real-world environment. To address this need, we introduced MELD-Keypoints, a speaker-specific extension of the MELD video dataset that augments existing emotion annotations with face, body and hand key-points extracted using OpenPose and linked to identified speakers using pypbbox. The current MELD-Keypoints v0.1 release contains pose data for 4,059 videos and 283,093 frame-level key-point records. By providing emotion labels together with detailed facial, body and hand motion data in complex social scenes, MELD-Keypoints represents a valuable resource for future multimodal emotion recognition research

References

- [Pan et al., 2023] Pan, B., Hirota, K., Jia, Z., & Dai, Y. (2023). A review of multimodal emotion recognition from datasets, preprocessing, features, and fusion methods. *Neurocomputing*, 561(126866), 126866. <https://doi.org/10.1016/j.neucom.2023.126866>
- [Pereira et al., 2024] Pereira, R., Mendes, C., Ribeiro, J., Ribeiro, R., Miragaia, R., Rodrigues, N., Costa, N., & Pereira, A. (2024). Systematic review of emotion detection with computer vision and Deep Learning. *Sensors (Basel, Switzerland)*, 24(11), 3484. <https://doi.org/10.3390/s24113484>
- [Song, 2024] Song, S. (2024). Emotion detection of elderly people in nursing homes based on AI robot vision. *Soft Computing*, 28(23–24), 13989–14002. <https://doi.org/10.1007/s00500-023-08350-2>
- [Cai et al., 2023] Cai, Y., Li, X., & Li, J. (2023). Emotion recognition using different sensors, emotion models, methods and datasets: A comprehensive review. *Sensors (Basel, Switzerland)*, 23(5), 2455. <https://doi.org/10.3390/s23052455>
- [Geetha et al., 2024] Geetha, Mala, Priyanka, & Uma. (2024). Multimodal Emotion Recognition with Deep Learning: Advancements, challenges, and future directions. *An International Journal on Information Fusion*, 105(102218), 102218. <https://doi.org/10.1016/j.inffus.2023.102218>
- [Zhang, 2012] Zhang, Z. (2012). Microsoft Kinect sensor and its effect. *IEEE Multimedia*, 19(2), 4–10.

<https://doi.org/10.1109/mmul.2012.24>

- [Li et al., 2016] Li, S., Cui, L., Zhu, C., Li, B., Zhao, N., & Zhu, T. (2016). Emotion recognition using Kinect motion capture data of human gaits. *PeerJ*, 4(e2364), e2364. <https://doi.org/10.7717/peerj.2364>
- [Fang et al., 2023] Fang, H.-S., Li, J., Tang, H., Xu, C., Zhu, H., Xiu, Y., Li, Y.-L., & Lu, C. (2023). AlphaPose: Whole-body regional multi-person pose estimation and tracking in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6), 7157–7173. <https://doi.org/10.1109/TPAMI.2022.3222784>
- [Bazarevsky et al., 2020] Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., & Grundmann, M. (2020). BlazePose: On-device Real-time Body Pose tracking. In *arXiv [cs.CV]*. <https://doi.org/10.48550/arXiv.2006.10204>
- [Younis et al., 2024] Younis, E. M. G., Mohsen, S., Houssein, E. H., & Ibrahim, O. A. S. (2024). Machine learning for human emotion recognition: a comprehensive review. *Neural Computing & Applications*. <https://doi.org/10.1007/s00521-024-09426-2>
- [Ekman and Friesen, 2003] Ekman, P., Friesen, W.V.: Unmasking the face: a guide to recognizing emotions from facial clues. Ishk (2003)
- [Poria et al., 2019] Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E., & Mihalcea, R. (2019). MELD: A multimodal multi-party dataset for emotion recognition in conversations. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.
- [Dhall et al., 2012] Dhall, A., Goecke, R., Lucey, S., & Gedeon, T. (2012). Collecting large, richly annotated facial-expression databases from movies. *IEEE Multimedia*, 19(3), 34–41. <https://doi.org/10.1109/mmul.2012.26>
- [Jiang et al., 2020] Jiang, X., Zong, Y., Zheng, W., Tang, C., Xia, W., Lu, C., & Liu, J. (2020). DFEW: A large-scale database for recognizing dynamic facial expressions in the wild. In *arXiv [cs.CV]*. <https://doi.org/10.48550/arXiv.2008.05924>
- [Bänziger et al., 2012] Bänziger, T., Mortillaro, M., & Scherer, K. R. (2012). Introducing the Geneva Multimodal expression corpus for experimental research on emotion perception. *Emotion (Washington, D.C.)*, 12(5), 1161–1179. <https://doi.org/10.1037/a0025827>
- [Busso et al., 2008] Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S., & Narayanan, S. S. (2008). IEMOCAP: interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42(4), 335–359. <https://doi.org/10.1007/s10579-008-9076-6>
- [Miranda-Correa et al., 2021] Miranda-Correa, J. A., Abadi, M. K., Sebe, N., & Patras, I. (2021). AMIGOS: A dataset for affect, personality and mood research on individuals and groups. *IEEE Transactions on Affective Computing*, 12(2), 479–493. <https://doi.org/10.1109/taffc.2018.2884461>
- [Park et al., 2020] Park, C. Y., Cha, N., Kang, S., Kim, A., Khandoker, A. H., Hadjileontiadis, L., Oh, A., Jeong, Y., & Lee, U. (2020). K-EmoCon, a multimodal sensor dataset for continuous emotion recognition in naturalistic conversations. *Scientific Data*, 7(1). <https://doi.org/10.1038/s41597-020-00630-y>
- [Fourati and Pelachaud, 2014] Fourati, N., & Pelachaud, C. (2014). Emilya: Emotional body expression in daily actions database. *Proceedings of the Language Resources and Evaluation Conference*, 3486–3493.
- [Keshari and Palaniswamy, 2019] Keshari, T., & Palaniswamy, S. (2019). Emotion recognition using feature-level fusion of facial expressions and body gestures. *2019 International Conference on Communication and Electronics Systems (ICCES)*.
- [Hsu et al., 2018] Hsu, C.-C., Chen, S.-Y., Kuo, C.-C., Huang, T.-H., & Ku, L.-W. (2018). EmotionLines: An Emotion Corpus of Multi-Party Conversations. *Proceedings of the Language Resources and Evaluation Conference*.

- [Cao et al., 2021] Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., & Sheikh, Y. (2021). OpenPose: Realtime multi-person 2D pose estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1), 172–186. <https://doi.org/10.1109/TPAMI.2019.2929257>.
- [Raaj et al., 2019] Raaj, Y., Idrees, H., Hidalgo, G., & Sheikh, Y. (2019). Efficient online multi-person 2D pose tracking with recurrent spatio-temporal affinity fields. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4615–4623.
- [Simon et al., 2017] Simon, T., Joo, H., Matthews, I., & Sheikh, Y. (2017). Hand Keypoint Detection in Single Images Using Multiview Bootstrapping. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Redmon et al., 2016] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Siv et al., 2020] Siv, R., Mancas, M., Sreng, S., Chhun, S., & Gosselin, B. (2020). People tracking and re-identifying in distributed contexts: PoseTReID framework and dataset. *2020 12th International Conference on Information Technology and Electrical Engineering (ICITEE)*.
- [Siv et al., 2022] Siv, R., Mancas, M., Gosselin, B., Valy, D., & Sreng, S. (2022). People tracking and re-identifying in distributed contexts: Extension study of PoseTReID. *2022 9th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 337–342.
- [Kalabasi, 2022] Kalabasi, A. (2022). *Images of friends character for face recognition*. Kaggle. <https://www.kaggle.com/datasets/amiralikalbasi/images-of-friends-character-for-face-recognition>

Bundled Feature Encoding: A Shift-Tolerant Image Representation for Hyperdimensional Computing

C.T. Levins, D. Kerr, S.A. Coleman, J.P. Quinn

University of Ulster

Abstract

Hyperdimensional Computing (HDC) is a brain-inspired, lightweight machine learning paradigm that represents information using high-dimensional vectors. While HDC has shown significant promise in computer vision, the field currently lacks a robust and consistent framework for encoding image data. Existing whole-picture pixel-encoding methods are highly sensitive to spatial shifts, viewpoint variations, and background redundancy, strictly limiting their viability for real-world robotic perception. To address these limitations, this paper introduces Bundled Feature Encoding (BFE), a novel framework comprising three distinct methods: Column-Based, Row-Bundled, and Block-Bundled Feature Encoding. By binding distinct pixel segments into robust spatial features before bundling them into a single global image hypervector, BFE captures structural information that remains stable under perspective shifts and inherent camera noise. Experimental evaluations using real-world robotic navigation data demonstrate that one-dimensional BFE approaches, particularly row and column bundling, significantly improve similarity preservation during continuous movement while reliably maintaining orthogonality for dissimilar scenes. These results highlight BFE's potential as an efficient, shift-tolerant encoding paradigm for autonomous mobile robot vision and other resource-constrained perception systems.

Keywords: Hyperdimensional computing, Vector Symbolic Architecture, Computer Vision, Image Encoding, Feature Recognition

1 Introduction

Hyperdimensional Computing (HDC), also known as Vector Symbolic Architecture (VSA), is an emerging computational paradigm inspired by the distributed representations found in biological neural systems [Kleyko et al., 2023a, Plate, 1995, Kanerva, 1996, Gayler, 1998]. Information is encoded using very high-dimensional vectors whose statistical properties enable robustness to noise, efficient similarity comparison, and lightweight learning. These characteristics make HDC an attractive alternative to conventional machine-learning approaches, particularly for resource-constrained platforms such as mobile robots and embedded systems.

Despite growing interest in applying HDC to vision tasks, image encoding remains one of the least mature areas in the field [Kleyko et al., 2023b]. Existing methods typically fall into three categories: pixel-based encoding, feature-enhanced encoding, and hybrid neural-HDC approaches. Pixel-position encoding, while conceptually simple, suffers from extreme sensitivity to spatial shifts, viewpoint changes, and background redundancy. Hybrid methods improve accuracy but sacrifice the computational efficiency that makes HDC appealing. Feature enhanced encoding methods have mainly been used to develop inputs for image classifiers which calculate the most likely match between hypervectors. Hybrid methods have been developed as high efficiency alternatives to deep neural networks and perform moderate to well with far less computational cost. Although all these HDC image encoding strategies often use feature extraction, they may not work well for the direct image matching required for tasks such as Visual Place Recognition (VPR), a process where a robot determines its current location by matching its live camera view against a stored database of previously seen

images. As a result, there is a lack of practical frameworks for hyperdimensional encoding of images in a way that preserves meaningful structure while remaining lightweight for robotic applications. Therefore, this paper introduces Bundled Feature Encoding (BFE), a novel technique designed to address these limitations by capturing stable features that remain consistent across viewpoint variations. The method aims to improve robustness while maintaining the efficiency required for real-time robotic perception.

2 Hyperdimensional Computing

Binary hypervectors are a predominant representation scheme in Hyperdimensional Computing (HDC), relying on four core operations: similarity measurement, bundling, binding and permutation [Gayler, 1998]. Binary hypervector representations are generally preferred over bipolar, real, and complex hypervectors because computing devices can execute binary operations much more efficiently. An example of a binary vector is illustrated in Equation (1), when it's dimension d (number of bits) is sufficiently large (typically 10,000), it is called a hypervector.

$$\mathbf{v} \in \{0, 1\}^d : [\mathbf{v} = (0, 0, 1, 0, 1, 1, 0, \dots, 0)] \quad (1)$$

Hypervectors possess unique mathematical properties that enable robust, highly distributed information encoding. For binary hypervectors, comparing vectors is highly hardware-efficient using normalised Hamming similarity, which measures the proportion of matching bits. In high-dimensional spaces, randomly generated vectors are inherently orthogonal; they are completely uncorrelated and typically share 50% of their bits (a similarity of 0.5). A similarity of 1 indicates identical vectors, whereas a similarity of 0 means the vectors are exact complements (a bitwise XOR of the two would yield a vector of 1s).

Bundling creates a single hypervector that represents a set or collection of multiple hypervectors. For binary hypervectors, bundling is achieved via component-wise majority voting. In the event of a tie, the result is assigned either stochastically (randomly) or deterministically (e.g., biased toward zero). This process ensures the resulting hypervector remains the same dimensionality as its members. In a bundled hypervector, each component (bit for binary) matches the corresponding component of its constituents with a probability greater than 0.5. Consequently, the bundled representation maintains a measurable similarity to each of its constituent vectors. Although individual bits may be flipped by the majority-vote process, acting as local interference, the massive scale of high-dimensional space ensures that the overall similarity remains statistically stable and reliable. This “holographic” property allows bundling to reliably encode relationships and sets.

Binding is the operation used to associate two hypervectors, creating a compositional representation of a relationship. This is typically used to form role/filler pairs, key/value associations, or symbol/attribute links, such as binding the attribute “red” to the object “circle”. For binary hypervectors, binding is implemented as a bitwise Exclusive OR (XOR) operation as illustrated in Equation (2).

$$\mathbf{C} = \mathbf{A} \oplus \mathbf{B} \quad (2)$$

The binding operation is characterised by two essential properties: orthogonality and invertibility. First, binding acts as a similarity-destroying (information-scrambling) transformation where the resulting hypervector $\mathbf{C}$ is dissimilar to both operands $\mathbf{A}$ and $\mathbf{B}$. Specifically, if $\mathbf{B}$ is a random hypervector, $\mathbf{C}$ shares approximately 50% of its bits with $\mathbf{A}$, rendering the vectors orthogonal in high-dimensional space. This stochastic behaviour ensures that binding binary hypervectors produces a result that does not interfere with the identity of its constituents. Second, the operation is invertible because XOR is its own inverse. This makes unbinding straightforward: knowing one operand allows recovery of the other via $\mathbf{A} = \mathbf{C} \oplus \mathbf{B}$, enabling the reliable retrieval of encoded information.

Permutation in hyperdimensional computing provides a consistent, reversible method for encoding sequence and structure. While mathematically representable by complex permutation matrices, practical HDC almost exclusively employs a simple circular shift of the vector's components for hardware efficiency [Schlegel et al., 2022] (e.g., an order-1 shift moves (a, b, c, d, e) to (b, c, d, e, a)). Crucially, the permutation

operation is invertible, distributes over component-wise operations, and acts as an information scrambler, producing a vector that is completely dissimilar to the original.

With these foundational operations established, hyperdimensional computing can be applied to encode complex, real-world data. Whole-image Pixel Position Encoding (WPPE) encodes an image by combining pixel intensities and their coordinates [Mitrokhin et al., 2019, Kleyko et al., 2019]. A set of hypervectors is generated to represent pixel intensities (256 is typical for an 8-bit image). The entire set can be orthogonal by generating random hypervectors, or the similarity between intensities can be controlled by generating only one random hypervector for the lower bound (zero) interval. The next hypervector is then produced by randomly flipping bits in the hypervector, a flip probability can be adjusted to control how similar the neighbouring hypervectors are. This is repeated across the full set. The Hamming similarity of the hypervectors representing the 256 greyscale intensity values can be visualised using heatmaps, as shown in Figure 1. The three figures illustrate the effect of different flip probabilities on the representational similarity. Figure 1(a) uses a flip probability of 0.5, resulting in random, orthogonal hypervectors for all distinct grey values (shown as a uniform dark background with a similarity of 0.5) and a sharp, 1-pixel-wide diagonal line where identical values have a similarity of 1.0. Figure 1(b) shows that reducing the flip probability to 0.015 preserves similarity between close shades of grey while leaving non-similar shades orthogonal. Finally, Figure 1(c) demonstrates that a flip probability of 0.0045 creates a wider, smoother band of similarity across the intensity range, with only the most distant pixel intensities approaching orthogonality.

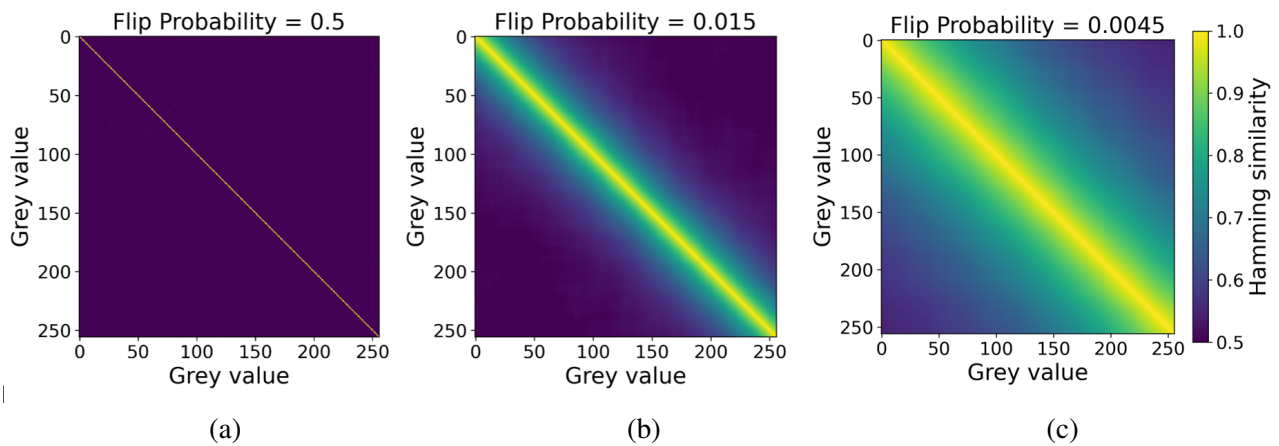


Figure 1: Heatmaps illustrating the Hamming similarity between hypervectors representing 256 greyscale intensity values at different flip probabilities.

To encode spatial positions, the base intensity hypervector for each pixel is systematically permuted (circularly shifted). The total shift amount is determined by the pixel's x and y coordinates. To prevent spatial collisions, where different coordinate pairs might accidentally produce the same total shift, a prime number multiplier (e.g., $P = 31$) is applied to one of the axes to mathematically differentiate them. Finally, the complete image is represented as a single global hypervector by bundling all of these spatially permuted pixel vectors together using majority rule.

This WPPE approach can result in a massive number of bound features in the high-dimensional space. For larger images, the majority of the data might be considered background, leading to significant redundancy or recognition errors. The positional encoding of each pixel makes image recognition extremely sensitive to small spacial shifts of the image. Thus, the encoding fails to capture and compress meaningful semantic data. Instead, the accumulated noise dilutes the actual structural information relevant for classification and adds to the computational load. Consequently, pixel-based encoding encounters a performance ceiling, struggling significantly with classification tasks that require discernment of complex, localised features, necessitating the development of more sophisticated methods like feature enhanced and hybrid approaches. To unlock HDC's true potential for computer vision, the encoding strategy must shift from rigidly memorising exact pixel coordinates to

binding and bundling resilient structural segments that remain stable even when the perspective changes.

3 Methodology

To overcome the extreme spatial sensitivity of traditional Whole-image Pixel Position Encoding (WPPE), this paper introduces and evaluates three novel encoding techniques based on localised structural feature detection: Column Bundled Feature Encoding (CBFE), Row Bundled Feature Encoding (RBFE), and Block Bundled Feature Encoding (BBFE). Rather than strictly encoding absolute, single-pixel coordinates, these methods bind one- or two-dimensional segments of the image using vertical columns, horizontal rows, or discrete blocks, respectively, into recognisable structural features. These extracted features are then bundled to generate a global, spatially robust image representation. The primary hypothesis is that extracting specific directional structures provides targeted invariance to viewing angle shifts; for instance, capturing only vertical features (CBFE) should yield invariance to horizontal perspective shifts, whereas horizontal features (RBFE) may prove robust against vertical panning or camera pitch changes. This study first details the mechanics of these three approaches, followed by experimental evaluation of their performance, computational efficiency, and sensitivity limits using highly similar real-world image datasets.

Across all approaches, the process begins by generating a base dictionary of 256 pixel-intensity hypervectors based on selected flip probabilities. Every pixel in the image is then mapped to its corresponding intensity hypervector. For Column-Based Feature Encoding (CBFE), we extract spatial information using a vertical sliding window. As illustrated in Figure 2, this window isolates a specific sequence of pixels, such as a 5-pixel vertical column. To preserve the spatial order within this column, a circular shift (permutation) is applied to each pixel's hypervector based on its vertical position. The top pixel's hypervector remains unchanged (zero shift), the second is shifted by one position, the third by two, and so on. These permuted hypervectors are then bound together to create a single, unique hypervector that represents that specific column feature. This sliding window moves sequentially to capture features across the entire image. Thus when processing a 30 x 30 pixel image with a 5-pixel vertical window we require 30 x 25 column features. Once all column features are extracted, their respective hypervectors are bundled together to produce a single composite hypervector that represents the entire image.

The primary advantage of this approach is its robustness to perspective shifts. By exclusively capturing vertical features, the CBFE method achieves greater invariance to changes in the viewing angle. In contrast, horizontal features tend to skew or distort when the angle changes, which disrupts similarity matching in standard pixel-based encoding methods.

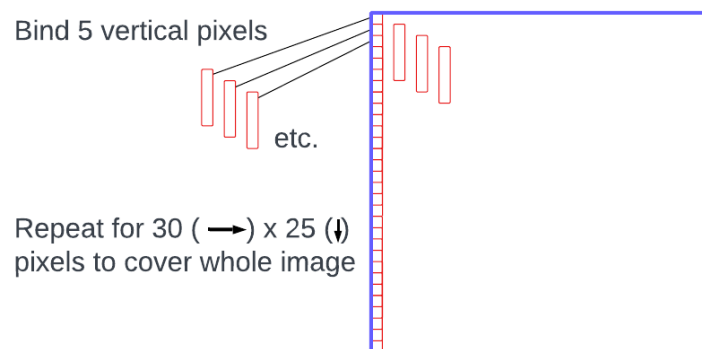


Figure 2: For CBFE, short columns of pixels are bound (5 pixel windows are illustrated in this image). This is repeated for the entire image. The full set is then bundled to produce the final image hypervector.

Row Bundled Feature Encoding (RBFE) directly adapts the principles of the CBFE technique by extracting spatial features along horizontal pixel rows rather than vertical columns. By prioritising horizontal spatial struc-

tures, RBE is specifically designed to achieve invariance against vertical movement and changes in camera pitch. Under such conditions, vertical features are prone to skewing and distortion, whereas horizontal features remain relatively stable, thereby preserving the structural integrity required for accurate similarity matching.

Block Bundled Feature Encoding (BBFE) expands upon the one-dimensional approaches of CBFE and RBE by extracting two-dimensional spatial features. This method utilises a sliding window to capture overlapping 2D blocks of pixels, such as an 8×8 pixel region, as individual features. To preserve the precise 2D spatial arrangement within each block, both the horizontal (x) and vertical (y) positions of every pixel must be explicitly encoded. Similar to Whole Picture Pixel Encoding (WPPE), this is achieved using position-dependent permutation shifts, incorporating a prime number multiplier to mathematically distinguish the x -coordinates from the y -coordinates during the shifting process. Once the pixels within a block are positionally encoded, their respective hypervectors are bound together to generate a single feature hypervector representing the structural composition of that specific 8×8 region. As the window traverses the image, the resulting feature hypervectors for all overlapping blocks are subsequently bundled to form the final composite hypervector for the entire image. This 2D block approach presents a clear theoretical trade-off in hyperdimensional system design. On one hand, it significantly enhances the representation of complex, two-dimensional structural information, theoretically making it more robust to simultaneous shifts in multiple directions. On the other hand, this high-fidelity structural capture comes at a substantially higher computational cost, as permuting, binding and bundling overlapping multi-dimensional arrays requires exponentially more operations than the 1D column or row methods.

To mitigate computational overhead during the feature extraction process, the stride of the sliding window can be adjusted. In the standard implementation, a baseline stride of 1 is utilised, meaning the window advances by a single pixel each time to capture a highly dense, overlapping set of spatial features. However, to systematically reduce the required number of hyperdimensional binding and bundling operations, larger stride intervals, for example strides of 2 and 5, can be introduced. By increasing the stride, the window skips intermediate pixels between extractions, which significantly decreases the total volume of feature hypervectors generated per image.

4 Results and Evaluation

To evaluate the encoding and recognition capabilities of CBFE, RBE, and BBFE, we utilised a subset of the Multi-Domain Dataset for Robots (MDDR) [Wozniak et al., 2025], samples of which are shown in Figure 3. This subset comprises an 800-frame video sequence captured by a robot-mounted Pi camera navigating a looping corridor. The dataset provides an excellent testbed for these encoding methods due to its subtle inter-frame movements and realistic sensor noise, such as auto-exposure adjustments and white balance drift.

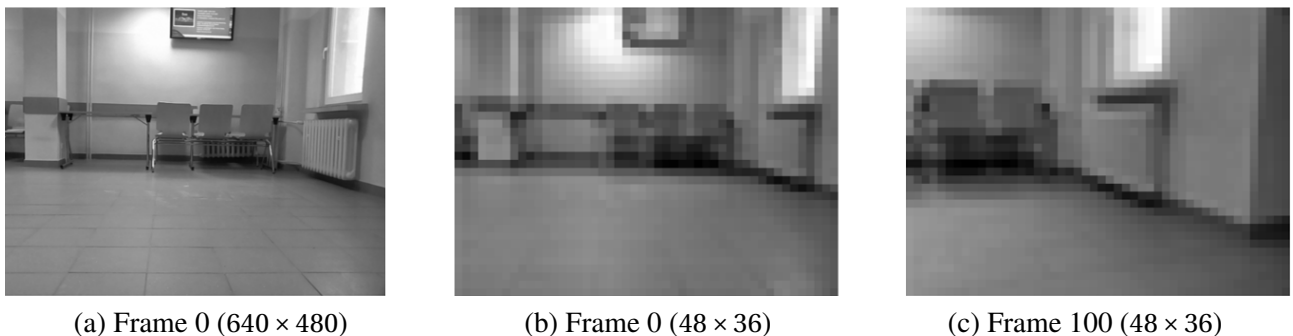


Figure 3: Sample frames from the MDDR sequence illustrating the resolution downsampling and corridor movement. (a) The initial reference frame at full 640×480 pixel resolution. (b) The same initial frame downsampled to 48×36 pixels to reduce computational overhead. (c) Frame 100 downsampled to 48×36 pixels, demonstrating the visual change after robot movement.

During the first 20 frames, the robot remains entirely stationary. Although these initial images appear visually identical to the human eye, yielding a Structural Similarity Measure (SSIM) of approximately 0.99, direct pixel-to-pixel comparisons at the full 640×480 resolution reveal that only 3.4% to 22% of the pixels actually match. This underlying variance makes this stationary period highly useful for benchmarking the similarity of the encoded hypervectors. Finally, to mitigate the significant computational overhead that high image resolutions impose on hyperdimensional processing, all images were downsampled from their native 640×480 resolution to 48×36 pixels using the LANCZOS resampling filter from the Python Pillow (PIL) library prior to encoding.

The performance of the encoding techniques (CBFE, RBFE, and BBFE) is heavily influenced by the size of the spatial features extracted by the sliding window. Using the 800-image sequence from the MDDR dataset, we analysed the effects of varying feature dimensions, specifically column length, row length, and block size. For all evaluations discussed in this section, the greyscale flip probability was fixed at 0.01. This value was established to provide an optimal balance: it ensures neighbouring greyscale intensities share enough similarity to mitigate realistic camera noise (such as auto-exposure drift), while preventing the hypervectors from blending into complete uniformity. In these evaluations, each frame is encoded into a composite hypervector and compared against the reference hypervector of the initial frame (Image 0). The sequence consists of a stationary period for the first 20 frames, where high Hamming similarity (> 0.8) is expected despite camera noise, followed by physical robot movement, where similarity should ideally drop to ≈ 0.5 (orthogonality) to indicate that the scene has changed.

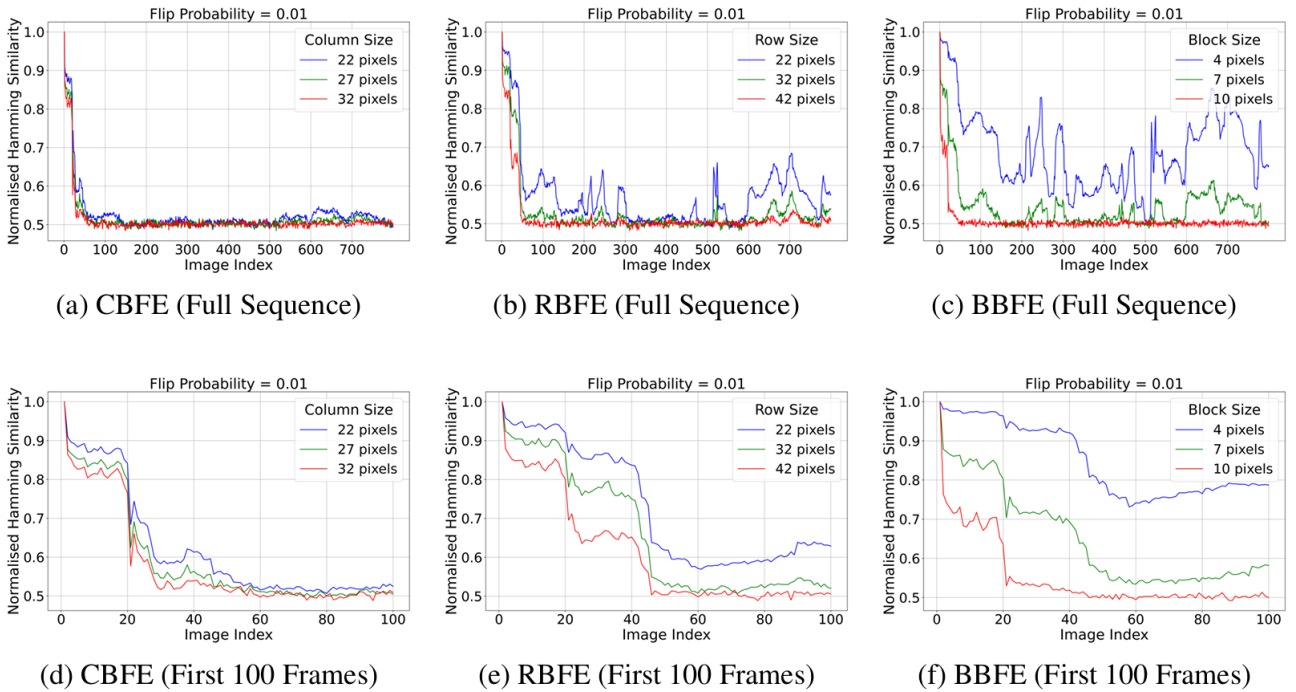


Figure 4: The effect of varying feature dimensions on normalized Hamming similarity. The top row (a, b, c) shows the full 800-image sequence for each encoding method, while the bottom row (d, e, f) provides a truncated view of the first 100 frames to highlight the system’s sensitivity during the transition from a stationary camera to physical movement. All plots maintain a constant greyscale flip probability of 0.01.

Column-Based Feature Encoding (CBFE): Figure 4(a) illustrates the impact of column size. When a shorter feature column of 22 pixels (blue line) is used, the system exhibits spurious recognition later in the sequence, with similarity frequently spiking above 0.5 when it should be orthogonal. Increasing the column size to 32 pixels (red line) successfully suppresses this noise. The truncated plot reveals that for the 32-pixel column, similarity remains high during the stationary phase (frames 0–20) before sharply and consistently

dropping to ≈ 0.5 once movement begins, indicating reliable differentiation of new scenes.

Row Bundled Feature Encoding (RBFE): RBFE demonstrates a higher susceptibility to false similarity than CBFE. As shown in Figure 4(b), horizontal feature extraction struggles to maintain orthogonality with shorter rows; both the 22-pixel and 32-pixel row sizes show significant false positive spikes throughout the 800-frame sequence. To achieve the stable baseline orthogonality (≈ 0.5) seen in the CBFE method, RBFE requires a significantly larger feature window, specifically a row size of 42 pixels (red line).

Block Bundled Feature Encoding (BBFE): The 2D spatial extraction of BBFE presents a distinct sensitivity profile, as seen in Figure 4(c). A small block size, indicated as 4 pixels (blue line), fails entirely to differentiate the shifting corridor, resulting in extreme false-positive recognition (similarity ranging from 0.6 to > 0.8) long after the robot has moved. Conversely, a block size of 10 pixels (red line) perfectly suppresses this late-sequence noise, remaining strictly orthogonal for non-matching images. However, the truncated plot highlights the theoretical trade-off of 2D block encoding: the high structural fidelity of a larger block makes it highly sensitive to the natural camera noise present even in the first 20 stationary frames, causing a much sharper initial drop in similarity compared to the 1D methods.

In the context of Visual Place Recognition (VPR), determining the spatial extent a robot can travel before the current view diverges from the reference image is critical for establishing an efficient keyframe capture strategy. To maintain continuous localisation, a new reference image must be captured just before the normalised Hamming similarity degrades to the orthogonal noise floor of approximately 0.5. Based on the optimal, noise-suppressing feature dimensions identified in Figure 4 (the red plots), the permissible travel distance varies significantly depending on the encoding technique used. As the robot is stationary for the first 20 frames, physical translation only commences at frame 21. Block Bundled Feature Encoding (BBFE) with a 10-pixel block exhibits extreme structural sensitivity; upon movement, its similarity drops to the 0.5 noise floor almost immediately (within 5 to 10 frames of motion). This indicates that while BBFE is highly precise, it requires dense, frequent keyframing. Column-Based Feature Encoding (CBFE) with a 32-pixel column provides a moderate tolerance to movement and maintains a similarity above the noise threshold for approximately 15 to 20 frames of movement before bottoming out near frame 40. Conversely, Row Bundled Feature Encoding (RBFE) with a 42-pixel row demonstrates the widest perceptual tolerance to the forward motion characteristic of this corridor dataset. It sustains a similarity above 0.6 for nearly 25 frames of continuous movement, only decaying to the noise floor around frame 45. Consequently, while BBFE enforces strict spatial bounds ideal for pinpoint accuracy, RBFE allows for a much sparser mapping strategy, significantly reducing the memory footprint required over extended travel trajectories.

Overall, the findings highlight a fundamental trade-off in hyperdimensional feature extraction. While expanding feature dimensions is necessary across all methods to reliably suppress false positives and ensure non-matching images are recognised as orthogonal, it simultaneously amplifies the system's sensitivity to inherent sensor noise during static scenes. This vulnerability is most pronounced in multi-dimensional block encoding (BBFE), which enforces strict structural fidelity at the cost of rapid similarity degradation. Consequently, the choice of encoding method directly dictates the operational strategy for Visual Place Recognition: highly sensitive methods like BBFE offer pinpoint precision but demand dense, frequent keyframing, whereas one-dimensional approaches like RBFE provide wider perceptual tolerance, enabling a more memory-efficient, sparse mapping strategy.

5 Conclusion

This work introduced Bundled Feature Encoding as a lightweight Hyperdimensional Computing method to improve robustness in visual recognition tasks. By extracting structural information via sliding windows, Column-Based (CBFE) and Row Bundled Feature Encoding (RBFE) successfully provide targeted spatial invariance, overcoming the strict pixel-to-pixel limitations of Whole Picture Pixel Encoding (WPPE). Notably, the one-dimensional extraction of CBFE and RBFE provides greater perceptual tolerance and spatial invariance compared to the rigid, two-dimensional Block Bundled Feature Encoding (BBFE), which suffers from significant computational overhead and extreme sensitivity to movement.

Theoretically and empirically, the orientation of the feature window dictates the system’s targeted invariance. CBFE demonstrates resilience to horizontal perspective shifts, performing well when the robot turns. Conversely, RBFE exhibits superior robustness to vertical camera panning and pitch changes. Crucially, in forward-translating environments, RBFE demonstrates the widest spatial tolerance, sustaining accurate image recognition significantly longer than CBFE or BBFE as the robot advances.

While hyperdimensional encoding remains sensitive to small variations in lighting and camera auto-exposure micro-adjustments, this was successfully mitigated by tuning the greyscale flip probability (e.g., 0.01) to increase intensity similarity, alongside expanding the spatial dimensions of the feature windows. However, our analysis reveals a fundamental trade-off: while larger feature dimensions successfully suppress false-positive recognition between differing scenes, they simultaneously amplify the system’s fragility to static sensor noise. This trade-off directly informs the operational strategy for real-world applications such as Visual Place Recognition (VPR). The strict structural fidelity of BBFE demands dense, frequent keyframing, whereas the wider tolerance of RBFE enables a highly memory-efficient, sparse mapping strategy. Overall, the results demonstrate that one-dimensional feature bundling provides the most balanced, efficient, and reliable approach for lightweight robotic vision systems.

Acknowledgement

This work was supported by the Northern Ireland Department for the Economy Postgraduate Studentship scheme.

References

- [Gayler, 1998] Gayler, R. W. (1998). Multiplicative binding, representation operators & analogy. In *Advances in Analogy Research: Integration of Theory and Data from the Cognitive, Computational, and Neural Sciences*, pages 1–4.
- [Kanerva, 1996] Kanerva, P. (1996). Binary spatter-coding of ordered k-tuples. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1112 LNCS:869–873.
- [Kleyko et al., 2019] Kleyko, D., Gayler, R. W., and Osipov, E. (2019). Learning sensorimotor control with neuromorphic sensors: Toward hyperdimensional active perception. *Science Robotics*, 4:1.
- [Kleyko et al., 2023a] Kleyko, D. V., Rachkovskij, D. A., Osipov, E., and Rahimi, A. (2023a). A survey on hyperdimensional computing aka vector symbolic architectures, part i: Models and data transformations. *ACM Computing Surveys*, 55(6).
- [Kleyko et al., 2023b] Kleyko, D. V., Rachkovskij, D. A., Osipov, E., and Rahimi, A. (2023b). A survey on hyperdimensional computing aka vector symbolic architectures, part ii: Applications, cognitive models, and challenges. *ACM Computing Surveys*, 55(9):175.
- [Mitrokhin et al., 2019] Mitrokhin, A., Sutor, P., Fermüller, C., and Aloimonos, Y. (2019). Learning sensorimotor control with neuromorphic sensors: Toward hyperdimensional active perception. *Science Robotics*.
- [Plate, 1995] Plate, T. A. (1995). Holographic reduced representations. *IEEE Transactions on Neural Networks*, 6:623–641.
- [Schlegel et al., 2022] Schlegel, K., Neubert, P., and Protzel, P. (2022). A comparison of vector symbolic architectures. *Artificial Intelligence Review*, 55:4523–4555.
- [Wozniak et al., 2025] Wozniak, P., Krzeszowski, T., and Kwolek, B. (2025). Multi-domain indoor dataset for visual place recognition and anomaly detection by mobile robots. *Scientific Data 2025 12:1*, 12:817–.

KeyJoint-STGCN: Keypoint-Based Spatio-Temporal Graph Learning for Monocular Trajectory Prediction

Patitapaban Palo, Satyajit Nayak, Kwanit Gupta, Satarupa Uttarkabat, Ganesh Sistu, Mohamed Tarek Ali

anSWer, Valeo India Private Limited, Chennai, India.

University of Limerick, Ireland

Abstract

Autonomous driving systems require robust motion understanding capabilities to ensure safe navigation in highly dynamic traffic environments. Accurate trajectory prediction of pedestrians and vehicles from monocular vision remains challenging due to viewpoint variations, occlusions, heterogeneous interactions, and complex scene dynamics. This paper presents KeyJoint-STGCN, a keypoint-driven spatio-temporal graph learning framework for monocular trajectory prediction in traffic scenes. The proposed approach first employs OpenPifPaf to estimate and track semantic keypoints of pedestrians and vehicles from monocular video sequences. These keypoints are then represented as graph nodes enriched with spatial position and temporal velocity features. A Spatio-Temporal Graph Convolutional Network (STGCN) is subsequently used to model motion dynamics and forecast future trajectories. To capture heterogeneous interactions among multiple road entities within the same frame, the framework constructs a multigraph representation with multiple edge connections encoding spatial and temporal dependencies at the keypoint level. The proposed method is evaluated on publicly available autonomous driving datasets, including JAAD, nuScenes, and BDD100K. Experimental results demonstrate that the proposed keypoint-centric graph representation improves trajectory forecasting performance and generalization capability compared with existing state-of-the-art trajectory prediction methods. The proposed framework highlights the effectiveness of combining semantic keypoint reasoning with spatio-temporal graph learning for intelligent transportation and computer vision applications.

Keywords: Keypoint-based representation, Monocular vision, Multi-agent interaction, Spatio-temporal graph convolution, Trajectory prediction.

1 Introduction

Trajectory prediction is an important research problem in computer vision and intelligent transportation systems due to its applications in autonomous driving, Advanced Driver Assistance Systems (ADAS), and traffic scene understanding [Ratre and Joshi, 2025]. Predicting the future motion of pedestrians and vehicles enables autonomous agents to better interpret dynamic environments and improve decision-making in complex urban traffic scenarios. However, accurate trajectory forecasting remains challenging because of behavioural uncertainty, occlusions, environmental variability, and heterogeneous interactions among multiple road users,

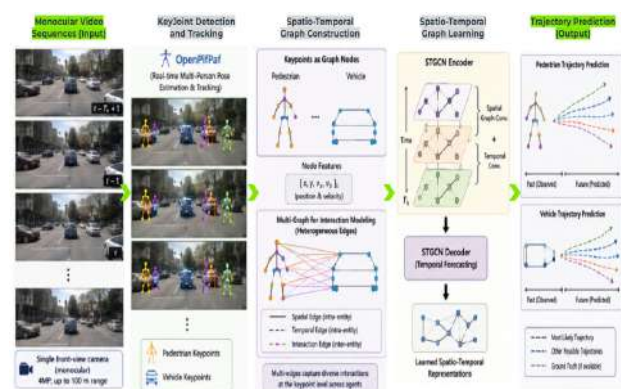


Figure 1: Overview of Proposed KeyJoint-STGCN

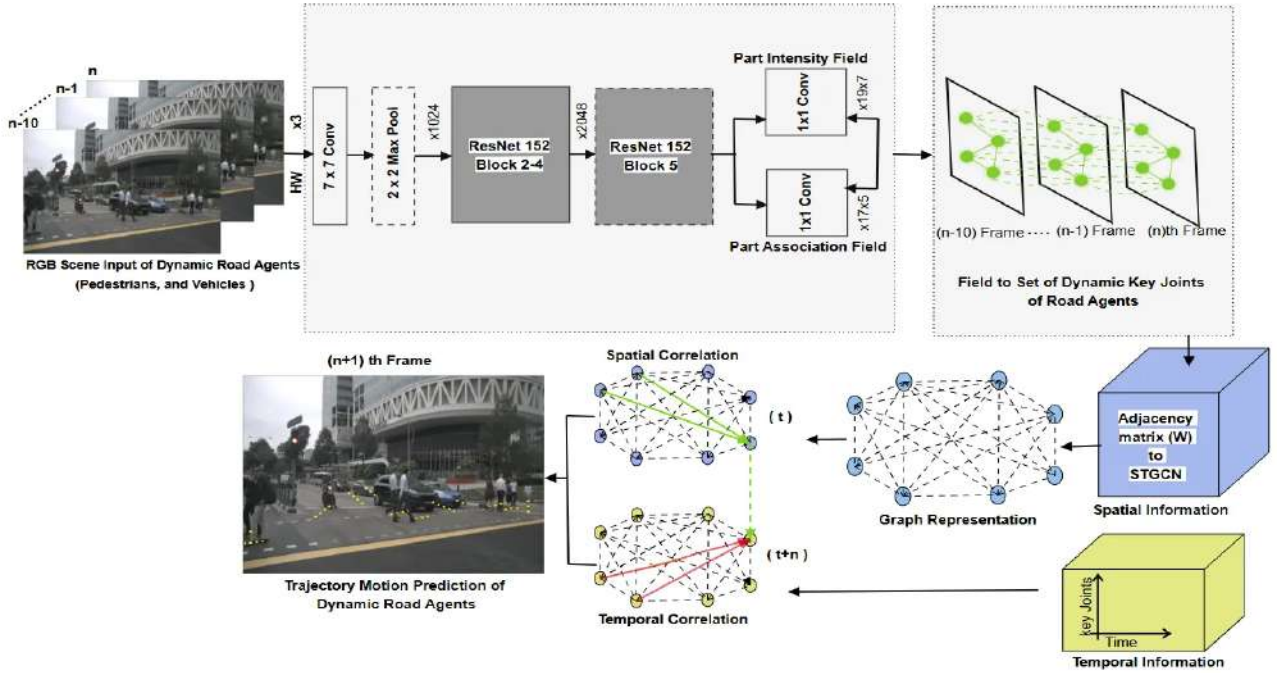


Figure 2: Proposed KeyJoint-STGCN Framework: The key joints detection process is done using the OpenPifPaf method, and the subsequent construction of a spatio-temporal graph for trajectory prediction. The graph is further enhanced using a multigraph approach. The STGCN is then employed to model temporal and spatial dependencies efficiently.

particularly in monocular vision settings briefed in Fig. 1 where depth perception is limited [Karle et al., 2022].

Recent deep learning approaches, including Social-GAN and Social-STGCNN [Xie et al., 2024], have demonstrated promising performance by modeling temporal dependencies and social interactions among agents. Similarly, graph-based methods such as GNNs and STGCNs have shown strong capability in learning spatial and temporal relationships in trajectory prediction tasks [Hsu and Chen, 2025]. Nevertheless, most existing approaches rely on coarse object-level representations, such as centroids or bounding boxes, and primarily focus on pedestrian prediction. As a result, they often fail to capture fine-grained articulated motion dynamics and semantic interaction cues that are critical in complex traffic environments involving heterogeneous road entities [Luo et al., 2024].

Pose and keypoint estimation methods, including OpenPose, and OpenPifPaf [Lambrecht et al., 2025], provide expressive representations of articulated motion and have shown effectiveness in action recognition and pedestrian intention analysis. However, semantic keypoint information remains underexplored in trajectory prediction frameworks, particularly for multi-class road-user interaction modeling in monocular driving scenarios.

To address these limitations, this paper proposes *KeyJoint-STGCN*, a keypoint-based spatio-temporal graph learning framework for monocular trajectory prediction. The proposed method employs OpenPifPaf [Kreiss et al., 2021] to estimate and track 17 human body keypoints and 24 vehicle semantic keypoints (via the pre-trained Apollo-Car3D model checkpoint) from monocular video sequences. These keypoints are represented as graph nodes with position and velocity features, while a multigraph interaction strategy is introduced to model heterogeneous spatial and temporal dependencies among road entities at the keypoint level. Subsequently, a Spatio-Temporal Graph Convolutional Network (STGCN) is utilized to learn motion dynamics and forecast future trajectories [Kahl et al., 2026].

An overview of the proposed framework is illustrated in Fig. 2. Unlike existing approaches that depend on multi-sensor fusion or bird’s-eye-view representations, the proposed framework operates entirely on monocular

front-view camera input, reducing computational complexity while maintaining efficient motion representation capability. The proposed method is evaluated on publicly available autonomous driving datasets, namely JAAD, nuScenes, and BDD100K [Kotseruba et al., 2016, Caesar et al., 2020, Yu et al., 2020]. Experimental results demonstrate improved trajectory prediction accuracy and generalization capability compared with existing state-of-the-art methods.

The pivotal research contributions of this work are summarized as follows:

- A novel keypoint-centric STGCN framework for monocular pedestrian and vehicle trajectory prediction.
- A multigraph interaction modeling strategy for capturing heterogeneous spatial and temporal dependencies at the semantic keypoint level.
- An efficient monocular vision-based trajectory prediction pipeline without relying on multi-sensor fusion architectures.
- Extensive validation on benchmark autonomous driving datasets demonstrating improved forecasting performance over state-of-the-art approaches.

2 Proposed Methodology: KeyJoint-STGCN

The proposed KeyJoint-STGCN framework consists of three major stages: semantic keypoint detection, spatio-temporal multigraph construction, and trajectory prediction using a Spatio-Temporal Graph Convolutional Network (STGCN). An overview of the framework is illustrated in Fig. 2.

2.1 Problem Formulation

Given M dynamic road entities observed over previous time steps $\{T_{n-10}, \dots, T_{n-1}\}$, the objective is to predict their future trajectories at time $T_{n+\Delta n}$. Each entity is represented using semantic key joints instead of coarse object-level centroids. For each key joint, the feature vector is defined as:

$$q_i = \{x_i, y_i, v_{x,i}, v_{y,i}\} \quad (1)$$

where (x_i, y_i) denote spatial coordinates and $(v_{x,i}, v_{y,i})$ represent velocity components.

2.2 Semantic Keypoint Detection

The proposed framework employs OpenPifPaf [Kreiss et al., 2022] for semantic keypoint tracking. Standard 17 COCO joints represent pedestrians, while a sparsified 24-keypoint vehicle skeleton (from ApolloCar3D) represents vehicle geometry. OpenPifPaf jointly estimates articulated key joints and establishes temporal associations across consecutive frames using Composite Fields and Temporal Composite Association Fields (TCAF). This enables robust tracking of pedestrians and vehicles without explicit data association.

2.3 Spatio-Temporal Multigraph Construction

For each frame, detected semantic key joints form multigraph nodes $G = (V, E)$. Horizontal, vertical, and velocity interactions are defined as

$$a_{ij}^d = \exp\left(-\alpha_d \frac{\|\mathbf{f}_i^d - \mathbf{f}_j^d\|^2}{N_d^2}\right), \quad d \in \{x, y, v\}, \quad (2)$$

where $\mathbf{f}^x = x$, $\mathbf{f}^y = y$, $\mathbf{f}^v = (v_x, v_y)$, and $N_x = W$, $N_y = H$, $N_v = V_{\max}$ denote the corresponding normalization factors. The fused adjacency matrix is

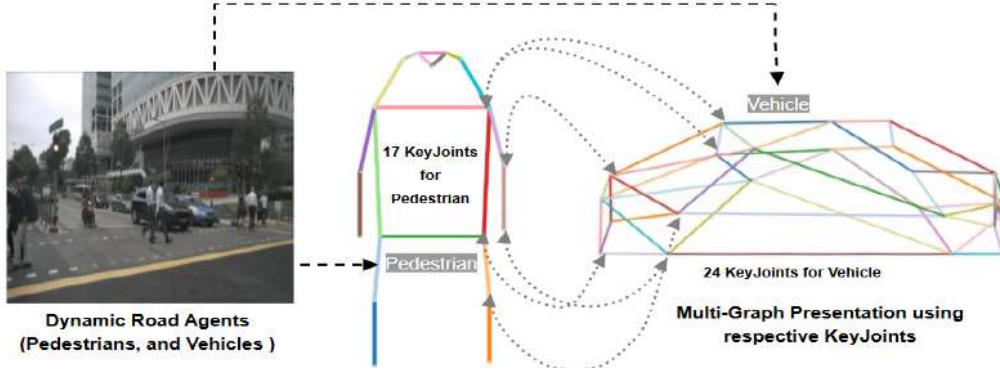


Figure 3: Graph construction process using KeyJoint-STGCN approach. Key joints are treated as individual nodes. Position and velocity data are incorporated into these nodes, forming the foundation of the graph. A multigraph is created to capture interactions between distinct entities, connecting one key joint (node) of one entity to another entity’s corresponding key joint (node) with multiple edges. As illustrated in Fig. 3, horizontal, vertical, and velocity edge relations are constructed via Eq. 2 and weighted by hyper-parameters $\lambda_x = 0.4, \lambda_y = 0.3, \lambda_v = 0.3$.

$$A = \lambda_x A_x + \lambda_y A_y + \lambda_v A_v, \quad \lambda_x + \lambda_y + \lambda_v = 1, \quad (3)$$

which is normalized before STGCN as

$$\hat{A} = D^{-1/2} (A + I) D^{-1/2}. \quad (4)$$

The resulting multigraph jointly models articulated keypoint structure and heterogeneous inter-agent interactions.

2.4 Trajectory Prediction using KeyJoint-STGCN

The constructed spatial graphs are fused into a unified interaction graph and processed using a Spatio-Temporal Graph Convolutional Network (STGCN) to jointly model spatial and temporal dependencies. The graph convolution operation is defined as:

$$H^{(l+1)} = \sigma(\bar{D}^{-1/2} \bar{A} \bar{D}^{-1/2} H^{(l)} W^{(l)}) \quad (5)$$

where $\bar{A}$ denotes the adjacency matrix with self-loops, $\bar{D}$ is the degree matrix, $H^{(l)}$ represents the node features, and $W^{(l)}$ denotes learnable weights.

Finally, a 3-layer Temporal Convolutional Network (TCN) with residual connections (kernel size $k = 3$, dilation factors $d \in \{1, 2, 4\}$, hidden channel size 64) forecasts future coordinates over the target horizon.

3 Experimental Results

3.1 Experimental Setup

The KeyJoint-STGCN framework is implemented in PyTorch and evaluated on a system with an Intel i7-13850HX CPU, 64 GB RAM, and a 4 GB GPU. Training uses the Adam optimizer with a learning rate of 1×10^{-4} and batch size of 128, converging in 50 epochs. Performance is assessed using ADE, FDE, and MSE for trajectory prediction accuracy.

Experiments are conducted on three benchmark datasets: JAAD (pedestrian behavior), nuScenes (multi-agent trajectory), and BDD100K (diverse urban driving) [Kotseruba et al., 2016, Caesar et al., 2020, Yu et al., 2020]. JAAD primarily focuses on pedestrian behaviour in urban driving scenarios, while nuScenes and BDD100K

provide large-scale multi-class traffic scenes with diverse environmental conditions and dynamic interactions. These datasets enable comprehensive evaluation of monocular trajectory prediction across pedestrians and vehicles.

3.2 Training and Validation Strategy

For all datasets, videos are divided into training, validation, and testing subsets. The proposed framework uses 10 observed frames to predict future trajectories over a 3-second horizon. OpenPifPaf is employed for semantic keypoint detection and tracking, and the extracted keypoints are converted into spatio-temporal multigraph representations before STGCN-based learning.

To validate model robustness, additional experiments are conducted using different pose trackers, including AlphaPose and PoseFlow [Wu et al., 2025]. Furthermore, ablation analysis is performed by varying the number of input frames to evaluate temporal dependency learning.



Figure 4: Examples of semantic keypoint detection for pedestrians and vehicles using OpenPifPaf on JAAD and BDD100K datasets.

3.3 Quantitative Performance Comparison

The proposed KeyJoint-STGCN is compared with state-of-the-art trajectory prediction methods, including DESIRE [Lee et al., 2017], Semantic-STGCNN [Rainbow et al., 2021], and GATraj [Cheng et al., 2023]. Tables 1 summarize the quantitative results on JAAD, nuScenes, and BDD100K datasets.

Table 1: Comprehensive performance comparison of trajectory prediction methods across JAAD, nuScenes, and BDD100K datasets.

Method	JAAD			nuScenes			BDD100K		
	ADE↓	FDE↓	MSE↓	ADE↓	FDE↓	MSE↓	ADE↓	FDE↓	MSE↓
DESIRE	0.8123	0.9347	703	2.4896	4.5923	1476	1.8901	2.9876	1075
Semantic-STGCNN	0.6975	0.7489	587	2.3128	4.2789	1201	1.5678	2.3789	823
GATraj	0.6450	0.7712	510	2.1050	4.2501	1035	1.4502	2.1405	755
KeyJoint-STGCN	0.6145	0.7214	462	2.1035	4.1107	932	1.2345	1.9456	741

Across all datasets, the proposed method consistently achieves lower ADE and FDE compared with existing methods. The improvements demonstrate the effectiveness of semantic keypoint-level motion representation and multigraph interaction modeling for monocular trajectory prediction.

3.4 Class-wise and Ablation Analysis

To further analyze performance across road-user categories, class-wise evaluation is conducted on the JAAD dataset. While pedestrian gains are substantial, vehicle trajectory forecasting is highly competitive with GATraj (1.9843 vs 1.9820 ADE on JAAD). This slight difference occurs because rigid vehicle dynamics are already well-approximated by bounding box centroids, unlike highly dynamic, articulated pedestrian body movements.

Table 2: Class-wise evaluation and pose tracker comparison on JAAD dataset.

Class-wise Evaluation				Pose Tracker Evaluation		
Class	Method	ADE↓	FDE↓	Pose Tracker	ADE↓	FDE↓
Vehicle	GATraj	1.9820	3.8862	PoseFlow	0.5446	0.6951
	KeyJoint-STGCN	1.9843	3.8945	AlphaPose	0.5223	0.6402
Pedestrian	GATraj	1.0340	1.8032	OpenPifPaf	0.5187	0.6287
	KeyJoint-STGCN	0.9215	1.6849	–	–	–

The impact of pose trackers is evaluated using PoseFlow, AlphaPose, and OpenPifPaf. As shown in Table 2, OpenPifPaf provides the best trajectory prediction accuracy due to its stable semantic keypoint tracking capability.

3.5 Effect of Input Frames and Empirical Complexity Metrics

The influence of temporal observation length is analyzed by varying the number of past input frames. Table ?? shows that 10 observed frames provide the best trade-off between trajectory prediction accuracy and computational complexity.

Table 3: Ablation study on graph topology.

Graph Topology	JAAD ADE↓	Observation
Single Spatial Graph	0.5912	Baseline
Proposed Multigraph	0.5187	Improved performance

Table 4: Model complexity.

Params (M)	FPS	Input
2.8	21.6	RGB (640 × 480)

The qualitative results demonstrated in Fig 5 that the proposed KeyJoint-STGCN effectively captures articulated motion dynamics and heterogeneous interactions, resulting in improved monocular trajectory prediction performance in complex traffic environments. Table 4 reports the model complexity of the proposed framework, consisting of 2.8M parameters and achieving a real-time inference speed of 21.6 FPS at an input resolution of 640 × 480. An ablation study isolating the graph construction strategy demonstrates that replacing the proposed multigraph with a single spatial graph increases the JAAD Average Displacement Error (ADE) from 0.5187 to 0.5912, highlighting the effectiveness of the multigraph topology in improving trajectory prediction performance. The lightweight monocular design enables real-time trajectory prediction without requiring computationally expensive multi-sensor fusion.

3.6 Discussion Insights

The results on JAAD, nuScenes, and BDD100K show that KeyJoint-STGCN consistently outperforms DESIRE, Semantic-STGCNN, and GATraj in ADE, FDE, and MSE, demonstrating strong cross-dataset general-

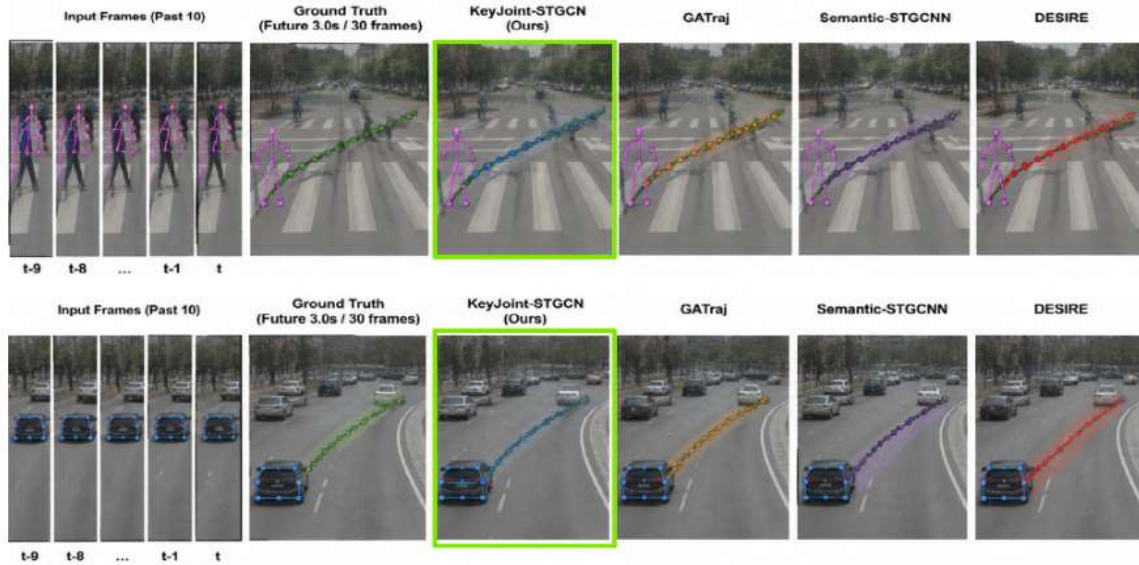


Figure 5: Predicted versus ground-truth (Row1 Fig: Pedestrian using JAAD and Row2 Fig: Vehicle using NuScenes Dataset) trajectory overlays compared to KeyJoint-STGCN marked in green.

ization. This gain is mainly due to keypoint-level motion modeling, which captures fine-grained articulated dynamics beyond bounding box representations.

Pose-aware modeling significantly improves pedestrian prediction by capturing subtle motion cues. Ablation results indicate that 10 input frames offer the best balance between accuracy and efficiency, while OpenPifPaf provides the most stable keypoint tracking, enhancing overall performance.

4 Conclusion and Future Scope

This paper introduced *KeyJoint-STGCN*, a keypoint-based spatio-temporal graph framework for monocular trajectory prediction of pedestrians and vehicles. By leveraging semantic keypoints from OpenPifPaf and a multigraph-STGCN formulation, the method captures fine-grained motion and inter-agent interactions more effectively than object-centric approaches, while operating efficiently on single-camera input. Experiments on JAAD, nuScenes, and BDD100K demonstrate consistent performance improvements and strong generalization across diverse traffic scenarios. However, performance depends on pose estimation quality, which may degrade under occlusion or poor visibility.

Future work will focus on improving robustness via uncertainty-aware or transformer-based pose modeling and incorporating scene context for enhanced prediction in complex environments.

References

- [Caesar et al., 2020] Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., and Beijbom, O. (2020). nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11621–11631.
- [Cheng et al., 2023] Cheng, H., Liu, M., Chen, L., Broszio, H., Sester, M., and Yang, M. Y. (2023). Gatraj: A graph- and attention-based multi-agent trajectory prediction model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 205:163–175.

- [Hsu and Chen, 2025] Hsu, B.-Y. and Chen, T.-S. (2025). Pedestrian trajectory prediction based on social pooling and spatio-temporal graph convolution network. In *2025 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, pages 1–5. IEEE.
- [Kahl et al., 2026] Kahl, F., Wegner, P., Kapsecker, M., Nissen, L., Faber, J., Jonas, S. M., and Reimer, L. M. (2026). Comparative assessment of accuracy in video-based monocular human pose estimation frameworks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- [Karle et al., 2022] Karle, P., Geisslinger, M., Betz, J., and Lienkamp, M. (2022). Scenario understanding and motion prediction for autonomous vehicles—review and comparison. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):16962–16982.
- [Kotseruba et al., 2016] Kotseruba, I., Rasouli, A., and Tsotsos, J. K. (2016). Joint attention in autonomous driving (jaad). *arXiv preprint arXiv:1609.04741*.
- [Kreiss et al., 2021] Kreiss, S., Bertoni, L., and Alahi, A. (2021). Openpipaf: Composite fields for semantic keypoint detection and spatio-temporal association. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):13498–13511.
- [Kreiss et al., 2022] Kreiss, S., Bertoni, L., and Alahi, A. (2022). Openpipaf: Composite fields for semantic keypoint detection and spatio-temporal association. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):13498–13511.
- [Lambricht et al., 2025] Lambricht, N., Englebert, A., Pitance, L., Fisette, P., and Detrembleur, C. (2025). Quantifying performance and joint kinematics in functional tasks crucial for anterior cruciate ligament rehabilitation using smartphone video and pose detection. *The Knee*, 52:171–178.
- [Lee et al., 2017] Lee, N., Choi, W., Vernaza, P., Choy, C. B., Torr, P. H. S., and Chandraker, M. (2017). Desire: Distant future prediction in dynamic scenes with interacting agents. pages 2165–2174.
- [Luo et al., 2024] Luo, S., Chen, W., Tian, W., Liu, R., Hou, L., Zhang, X., Shen, H., Wu, R., Geng, S., Zhou, Y., et al. (2024). Delving into multi-modal multi-task foundation models for road scene understanding: From learning paradigm perspectives. *IEEE Transactions on Intelligent Vehicles*.
- [Rainbow et al., 2021] Rainbow, B. A., Men, Q., and Shum, H. P. H. (2021). Semantics-stgcnn: A semantics-guided spatial-temporal graph convolutional network for multi-class trajectory prediction. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2959–2966.
- [Ratre and Joshi, 2025] Ratre, S. U. and Joshi, B. (2025). Trajectory and motion prediction of autonomous vehicles driving assistant system using distributed discriminator based bi-directional long short term memory model. *International Journal of Intelligent Transportation Systems Research*, 23(1):245–258.
- [Wu et al., 2025] Wu, R., Chen, Y., Xiao, J., Li, B., Fan, J., Dufaux, F., Zhu, C., and Liu, Y. (2025). Da-flow: Dual attention normalizing flow for skeleton-based video anomaly detection. *IEEE Transactions on Multimedia*.
- [Xie et al., 2024] Xie, J., Zhang, S., Xia, B., Xiao, Z., Jiang, H., Zhou, S., Qin, Z., and Chen, H. (2024). Pedestrian trajectory prediction based on social interactions learning with random weights. *IEEE Transactions on Multimedia*, 26:7503–7515.
- [Yu et al., 2020] Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., and Darrell, T. (2020). Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2636–2645.

Multi-session Online Open-vocabulary Object Mapping

Frank Gallagher¹, Marco Cagnetti², and John B. McDonald¹

¹*Maynooth University*

²*LAAS-CNRS, Université de Toulouse*

Abstract

In this paper, we present a multi-session online open-vocabulary object mapping system. Robots operating in previously unseen, complex real-world environments can benefit from semantic scene representations that can be constructed online and queried using natural language, enabling them to reason about tasks in the context of their surroundings. Multi-session mapping capabilities allow robots to reuse, refine and extend maps over repeated deployments while also providing the foundation for collaborative mapping by a team of robots. However, existing approaches typically address only subsets of these requirements, leaving the integration of persistent open-vocabulary mapping with an evolving multi-session SLAM estimate less explored. Our proposed system addresses this gap by integrating a robust sparse visual SLAM backend with an open-vocabulary object-mapping frontend. The resulting framework enables the incremental construction, refinement, and extension of open-vocabulary object maps over repeated deployments while maintaining consistency with the evolving SLAM estimate. We quantitatively evaluate geometric accuracy, object association, and runtime performance on seven Replica sequences, assess cross-session object consolidation on a split Replica sequence, and demonstrate persistent multi-session mapping on a real-world RGB-D sequence.

Keywords: SLAM, Multi-session, Open-vocabulary

1 Introduction

For robots to operate effectively in unknown real-world environments, they must be able to reason about their surroundings at a semantic level. This requires the ability to construct scene representations that capture not only the geometry of the environment, but also information describing the objects it contains and their spatial relationships. Robot perception systems have evolved over time from producing geometry-centric maps designed primarily for localisation and reconstruction toward increasingly semantic models that support higher-level reasoning and interaction. Recent approaches leverage vision foundation models to produce open-vocabulary scene representations that are not restricted to a predefined set of object categories, allowing natural language queries or instructions to be grounded directly in the map (Gu et al., 2024). This makes such representations particularly attractive for robots that must operate in previously unseen environments, where the relevant objects and task descriptions cannot be fully specified in advance.

For practical robot deployment, such scene representations should be constructed incrementally from incoming observations and updated as the underlying SLAM estimate changes, for example following loop closure and backend optimisation. Incremental construction allows a robot to build a map while exploring a previously unknown environment, rather than relying on a pre-existing reconstruction or accurately posed image sequences. Propagating subsequent pose corrections prevents the semantic representation from becoming inconsistent with the corrected spatial estimate. Recent systems have demonstrated online open-vocabulary mapping coupled with SLAM (Martins et al., 2025; Laina et al., 2026). Robots should also be able to retain and extend these representations when revisiting an environment. Multi-session operation allows observations acquired along separate trajectories to increase spatial coverage while preserving previously mapped information. This requires independently constructed geometric maps to be aligned and their associated semantic representations to be reconciled when overlap is discovered. Related work has demonstrated such reconciliation for collaborative semantic and open-vocabulary mapping with multiple robots (Chang et al., 2023; Gu et al., 2025).

However, the combination of persistent multi-session operation, open-vocabulary object mapping, and explicit coupling to the evolving state of a visual SLAM backend remains less explored.

In this paper, we address this gap by integrating the object-mapping component of ConceptGraphs (Gu et al., 2024) with ORB-SLAM3 (Campos et al., 2021), a mature sparse visual SLAM system that supports multi-session mapping. The resulting system incrementally constructs a multi-session, open-vocabulary object map from a single RGB-D input stream. For each input sequence, the system estimates camera poses, detects open-vocabulary object observations, associates them across views, and fuses them into a persistent 3D object map. By coupling the object map to the SLAM backend, the system can update object positions and geometry when optimised camera poses become available, and can extend the object map across multiple mapping sessions. This enables persistent language-grounded object mapping in environments that are explored incrementally over time. The code repository associated with this paper is available at <https://github.com/robotvisionmu/orb-cg>.

2 Related Work

Scene representations in robotics have progressively shifted from geometry-centric maps toward increasingly semantic and language-aware models of the environment. Sparse visual SLAM systems (Campos et al., 2021) represent the world using visually salient landmarks, providing an efficient structure for camera tracking and localisation. Dense mapping systems (Whelan et al., 2016) extend this representation by reconstructing detailed surface geometry of the environment. Semantic mapping (McCormac et al., 2018) further augments geometric maps with object labels, allowing robots to reason not only about where surfaces are, but also about what they correspond to.

3D scene graphs provide a more structured representation of indoor environments by combining metric, semantic, and topological information in a unified hierarchy. Armeni et al. (2019) proposed a seminal approach that uses a 3D mesh and registered panoramic images to construct a building-scale scene graph containing object, room, and camera semantics, as well as relationships between these entities. However, the representation is constructed offline from a precomputed reconstruction. Rosinol et al. (2021) extended this work by integrating visual-inertial SLAM to construct a metric-semantic mesh of the scene online. The system also tracks dynamic elements in the scene, such as humans and robots. The metric-semantic mesh is built online, but the higher-level layers of the dynamic scene graph are computed offline as a post-processing step after mapping. Hydra (Hughes et al., 2022) subsequently addressed this limitation by constructing the entire hierarchical structure online in real-time, but remains closed-vocabulary.

Other works have explored the interaction between scene graph representations and SLAM backends. In particular, vS-Graphs (Tourani et al., 2026) integrates a scene graph representation with ORB-SLAM3, using the semantics of structural scene elements (ground surfaces, walls, rooms) as constraints to improve SLAM performance. This demonstrates the potential benefits of coupling higher-level scene representations with visual SLAM. However, the output representation is limited in semantic scope to serve the primary goal of improving localisation and mapping performance through the integration of semantic constraints.

Recent work has extended scene representations toward open-vocabulary and language-grounded mapping. HOV-SG (Werby et al., 2024) constructs hierarchical open-vocabulary scene graphs that support language-based queries over objects and regions. ConceptGraphs (Gu et al., 2024) constructs object-centric 3D scene graphs from RGB-D observations by associating object detections across views and fusing them into a global 3D map. These systems demonstrate the value of open-vocabulary representations for querying and reasoning about 3D environments beyond a fixed semantic label set. However, they assume that accurate camera poses are already available and operate primarily offline or in a batch-processing setting. As a result, they do not directly address the problem of maintaining an open-vocabulary object map while a robot is actively exploring an unknown environment.

Recent systems such as ReasoningGraph (Gassol Puigjaner et al., 2026) and Describe Anything Anywhere at Any Moment (DAAAM) (Gorlo et al., 2026) address the online open-vocabulary mapping problem through extension of the Hydra framework, mentioned previously. Similar to our work, OVO (Martins et al., 2025) and FindAnything (Laina et al., 2026) integrate open-vocabulary semantics with online SLAM to maintain an open-vocabulary 3D map. These systems focus on single-session operation.

Recent research has also considered multi-robot semantic mapping. Hydra-Multi (Chang et al., 2023)

aligns, jointly optimises, and reconciles 3D scene graphs constructed by multiple agents, but retains Hydra’s closed-vocabulary representation. MR-COGraphs (Gu et al., 2025) supports distributed multi-robot open-vocabulary mapping by exchanging compact object graphs, although it is loosely coupled to an external localisation system and uses object correspondences for inter-map alignment. In contrast, our work couples open-vocabulary object mapping directly to the multi-map capabilities of ORB-SLAM3, allowing pose optimisation and map merging to be propagated to a persistent object representation across repeated deployments.

3 Methodology

The proposed system consists of a ConceptGraphs-based (Gu et al., 2024) open-vocabulary object-mapping frontend and an ORB-SLAM3 (Campos et al., 2021) visual SLAM backend. An overview of the system architecture is shown in Figure 1. The system maintains a separate object-centric, multi-map scene representation that mirrors the map and keyframe structure of the ORB-SLAM3 *Atlas*. This Atlas-aligned representation is denoted by $A = \{M_i\}_{i=1}^N$, where each map $M_i = (K_i, O_i)$ corresponds to an ORB-SLAM3 map and contains a set of keyframes K_i and a corresponding set of mapped objects O_i . Each keyframe $k = (\mathbf{T}_k, \mathbf{I}_k, \mathbf{D}_k, Z_k)$ comprises its pose, RGB image, depth image, and a set of object observations Z_k . A mapped object $o = (P_o, \mathbf{f}_o)$ is formed by fusing a set of supporting observations Z_o , where P_o is its 3D point-cloud representation and $\mathbf{f}_o$ is its aggregated CLIP feature.

Each input RGB-D frame is first processed by ORB-SLAM3, which selects a subset of frames as keyframes to provide spatial coverage of the environment for efficient mapping and localisation. Rather than processing every input frame, the frontend processes only keyframes, which form the interface between the SLAM backend and the object-mapping frontend. As ORB-SLAM3 performs loop closure, relocalisation, map merging, and pose optimisation, keyframe poses may be updated and keyframes may be reassigned to another map or removed. These changes are propagated to the frontend through keyframe events, allowing the corresponding object observations and mapped objects to remain consistent with the current SLAM estimate.

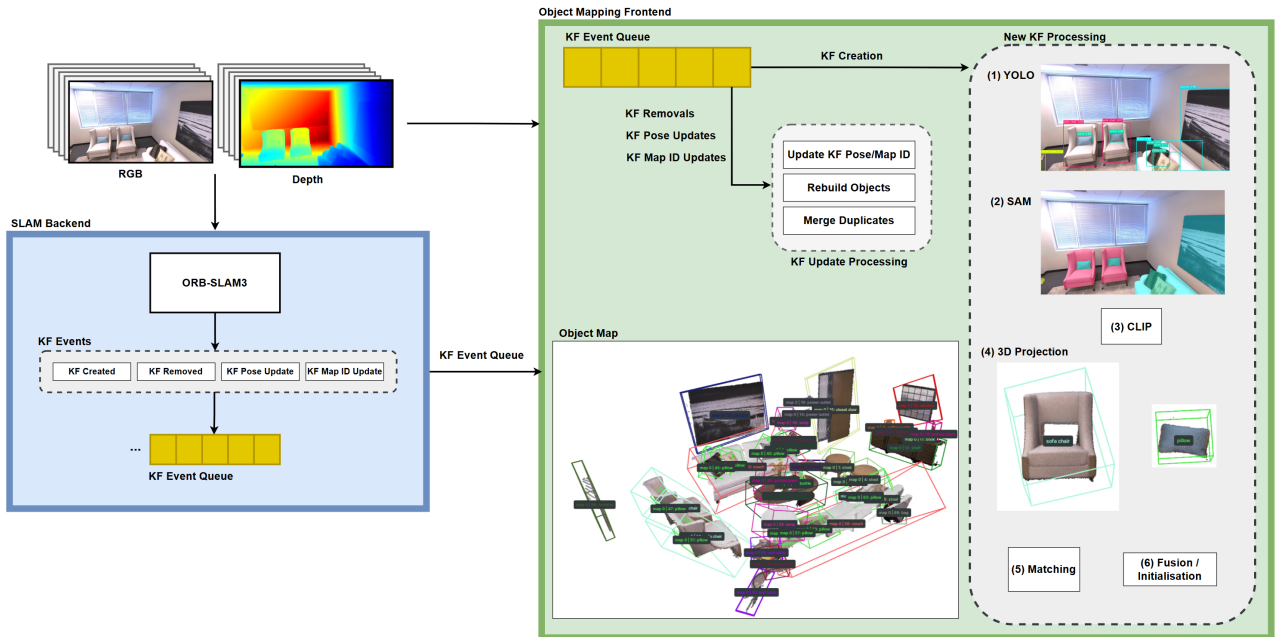


Figure 1: System architecture of the proposed ORB-SLAM3 and ConceptGraphs-based object-mapping pipeline. RGB-D frames are processed by ORB-SLAM3, which provides keyframe selection, camera tracking, pose optimisation, relocalisation, loop closure, and multi-session map management. Selected keyframes form the interface to the object-mapping frontend. A keyframe event queue propagates asynchronous SLAM state changes, including keyframe creation, removal, pose updates, and map reassignment. The frontend processes keyframes using open-vocabulary detection, segmentation, vision-language feature extraction, 3D projection, and object association to maintain a persistent open-vocabulary object map consistent with the current SLAM estimate.

3.1 SLAM Backend (ORB-SLAM3)

For each incoming RGB-D frame, ORB-SLAM3 estimates the camera pose relative to the active map and selects keyframes for mapping. Keyframe poses and map points are refined through local bundle adjustment, while place recognition enables loop closure, relocalisation, and map merging.

Of particular relevance to this work is the ORB-SLAM3 *Atlas*, which maintains multiple maps and a shared place-recognition database. An *Atlas* comprising all previous sessions can be loaded during initialisation, while a new map can be created when tracking cannot be recovered in an existing map. ORB-SLAM3 continues to search for correspondences with previously stored maps and merges them into a common coordinate frame when sufficient overlap is detected. This mechanism supports incremental multi-session operation, since maps constructed at different times, or from partially disconnected trajectories, can be reused and merged into a single persistent spatial representation of the environment.

3.2 Keyframe Event Interface

Since ORB-SLAM3 is a real-time, multithreaded system, its state can change asynchronously relative to the object-mapping frontend. To propagate these changes, we introduce a keyframe event queue. Events are emitted when a keyframe k is created or removed, when its pose $\mathbf{T}_k$ is updated, or when it is reassigned to another map following relocalisation or map merging. These events allow the frontend to update the corresponding observations Z_k and their map associations. While an event is waiting to be consumed, subsequent events for the same keyframe are coalesced, avoiding unnecessary reprocessing while ensuring that the frontend receives the most recent SLAM state.

3.3 Open-Vocabulary Object Mapping

The frontend is based on the open-vocabulary object-mapping component of ConceptGraphs. For each keyframe k , its RGB-D images, camera intrinsics, and estimated pose $\mathbf{T}_k$ are retrieved. YOLO-World (Cheng et al., 2024) extracts open-vocabulary object detections from the RGB image, and SAM (Kirillov et al., 2023) generates corresponding object masks. Detections are filtered to remove low-confidence, very small, background, or otherwise invalid regions. CLIP (Radford et al., 2021) vision-language features are computed for the remaining object proposals.

The resulting detections form the object observations Z_k . Each valid observation is projected into 3D using the depth image and camera intrinsics and transformed into the coordinate frame of its associated map using $\mathbf{T}_k$. The observation is then compared with existing objects $o \in O_i$ using spatial overlap and visual-semantic similarity. If a suitable match is found, the observation is added to the object’s supporting observations Z_o and fused into its geometry P_o and CLIP feature $\mathbf{f}_o$. Otherwise, a new mapped object is created.

Unlike the full ConceptGraphs pipeline, the proposed system maintains an object-level semantic representation rather than constructing a complete relationship graph. Relationship extraction is omitted because it is task-dependent and unnecessary for maintaining the persistent object map. The resulting representation therefore consists of open-vocabulary object instances with 3D geometry and aggregated vision-language features.

3.4 Updating the Object Map After SLAM Optimisation

When ORB-SLAM3 updates the pose $\mathbf{T}_k$ of a keyframe, the change is propagated to the frontend through a keyframe event. Any mapped object o supported by observations from the affected keyframe is marked for reconstruction. Its geometry P_o is then rebuilt from the supporting observations Z_o using the latest keyframe poses, ensuring consistency with the optimised SLAM estimate. If a keyframe is removed, its observations are removed from the corresponding supporting sets Z_o . Objects with no remaining valid observations are deleted. Duplicate objects within the same map are subsequently identified and merged.

3.5 Multi-Session Mapping

The proposed system leverages the multi-map capabilities of the ORB-SLAM3 *Atlas*. When ORB-SLAM3 creates a new map, for example following tracking loss or at the start of a new session, the frontend maintains a corresponding map $M_j = (K_j, O_j)$ while preserving the object sets associated with previous maps. Object

observations are therefore accumulated only within the map to which their originating keyframes currently belong.

If ORB-SLAM3 subsequently detects overlap and merges two maps, the resulting keyframe pose and map-assignment changes are propagated to the frontend through the event interface. The corresponding mapped objects are transferred into the resulting map and compared using the same spatial and visual-semantic association criteria applied to new observations. Duplicate representations of the same physical object can therefore be consolidated after the underlying SLAM maps have been aligned and merged.

This design allows object maps to be extended over repeated deployments in overlapping regions of the same environment. It also provides a basis for collaborative mapping, where observations collected during different sessions, or potentially by different robots, can be integrated into a shared persistent representation once their underlying SLAM maps are aligned or merged.

4 Evaluation

The proposed system is evaluated in both single-session and multi-session settings. Single-session evaluation measures the geometric accuracy of the constructed object map and the consistency of object association, using seven RGB-D sequences of the 3D Replica dataset (Straub et al., 2019) provided by iMAP (Sucar et al., 2021), for which ground-truth camera poses and semantic instance annotations are available. The `room 2` sequence was excluded because tracking loss caused ORB-SLAM3 to initialise a second map, preventing a single rigid trajectory alignment. Multi-session evaluation measures the system’s ability to reconcile and extend object maps across sessions. Cross-session object consolidation is evaluated on the Replica room 0 sequence split into two sessions, while persistent multi-session mapping is demonstrated on two real-world RealSense RGB-D sessions captured along partially overlapping trajectories. Finally, runtime performance is evaluated on the Replica sequences. The evaluation focuses on the performance of the integration of ORB-SLAM3 and the ConceptGraphs-based frontend, not the individual performance of each subsystem.

4.1 Single-Session

To prevent geometric accuracy metrics from being confounded by errors in object fusion, geometric performance is evaluated separately from object association. Geometric evaluation measures whether the object observations correctly represent the position and spatial extent of the corresponding physical object. Object association evaluation measures whether detections from the same physical object are grouped together without merging distinct objects or fragmenting one object across multiple map objects. For each Replica scene, visible ground-truth (GT) object geometry was reconstructed using semantic-instance images, depth images and GT camera poses corresponding to the keyframes selected by the system. Structural scene elements (walls, floors and ceilings) were excluded as these were not mapped by the system. Each object observation was assigned to the GT instance occupying the largest fraction of its predicted mask, provided this fraction was at least 0.5. GT masks were used only to establish identity and not to construct the predicted geometry. The ORB-SLAM3 trajectory was aligned to the Replica frame using an SE(3) transformation fitted between corresponding camera centres. The mean camera-centre residual after alignment was 0.90 cm, providing context for the contribution of trajectory error to the reported geometry metrics.

For geometric evaluation, observation point clouds were grouped by assigned GT instance and transformed into the Replica frame. Centroid error measures 3D localisation, while AABB (Axis Aligned Bounding Box) IoU measures agreement in object position and spatial extent. Table 1 reports pooled results over the seven Replica sequences. Of the 372 evaluable GT instances, 262 (70.4%) were assigned to object observations. Median centroid error was 3.92 cm, compared with a mean of 10.82 cm. Removing the worst 5% of instances reduced the mean to 7.13 cm, showing that a small number of large errors strongly affected the average. These errors were primarily associated with incomplete or oversized masks spanning neighbouring instances. Mean 3D AABB IoU was 0.499, indicating additional errors in reconstructed object extent and overlap.

Object association was evaluated from the GT assignments of each mapped object’s supporting observations. Of the 258 mapped objects, 33 had no observations assigned to a GT instance. Of the remaining 225 mapped objects, 154 (68.4%) were strictly clean, with every supporting detection assigned to the same GT

Table 1: Pooled single-session evaluation results on the 7 Replica sequences

<i>Geometry</i>		<i>Object Association</i>	
Evaluable GT Instances	372	Mapped Objects	258
Assigned GT Instances	262	Unassigned	33
Alignment Residual (mean)	0.9 cm	Clean	154 / 225 (68.4%)
Centroid (mean median)	10.82 cm 3.92 cm	Mixed	55 / 225 (24.4%)
AABB IoU (mean median)	0.499 0.486	Partially Assigned	16 / 225 (7.1%)
		Fragmented GT	48 / 262 (18.3%)

instance. A further 55 (24.4%) contained confirmed support from multiple GT instances, while 16 (7.1%) combined single-GT support with detections for which no GT identity could be established. Additionally, 48 of the 262 (18.3%) GT instances were assigned to observations distributed across multiple mapped objects. The association results therefore show both over-merging and fragmentation, despite most mapped objects comprising observations of a single physical instance.

4.2 Multi-Session

Cross-session object association was evaluated on the Replica `room 0` iMAP sequence, split into two sessions. The incoming-object association recall is reported, which measures how effectively objects from the second session are incorporated into the merged object map. Observations supporting each pre-merge mapped object were assigned to GT instances using the same procedure as in the single-session evaluation. A second-session object was eligible for consolidation if at least one of its observations shared a GT instance with an observation supporting a first-session object. Consolidation was successful when the second-session object and a corresponding first-session object were merged into the same final mapped object. Of the 22 eligible second-session objects, 19 were successfully consolidated (86.4%). As this metric operates on object predictions produced independently in each session, it cannot completely isolate cross-session consolidation from errors introduced during single-session mapping. Fragmentation or incomplete object reconstruction can alter both the number of eligible second-session objects and the evidence available for their subsequent association.

Multi-session persistence and map expansion were demonstrated using two real-world sessions captured, using a RealSense 435i RGB-D sensor, along partially overlapping trajectories in the Eolas Building at Maynooth University. Figure 2 shows the first session map is retained and extended by the second session following the ORB-SLAM3 map merge. This demonstrates the construction of a persistent object map over repeated deployments.

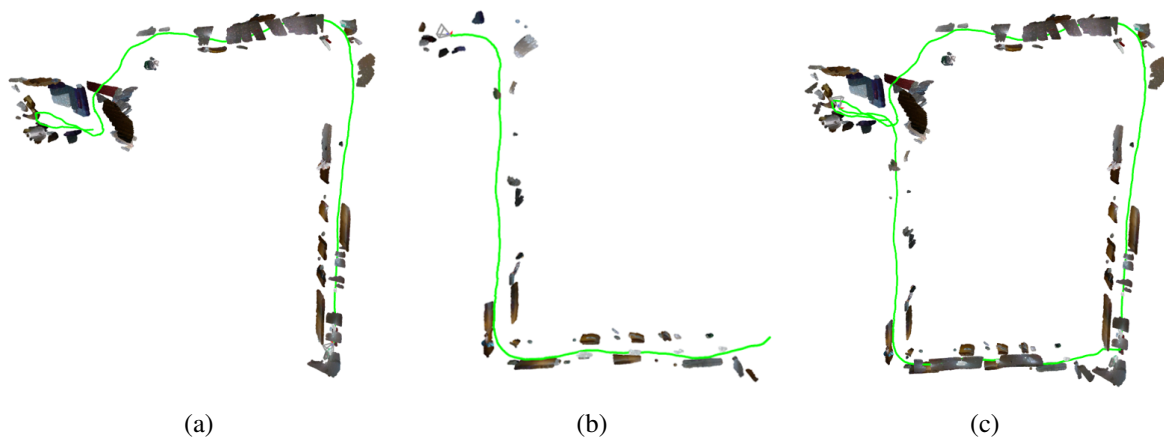


Figure 2: Multi-session object mapping on real-world scene captured by RealSense. Two partial maps (a) and (b) are constructed from separate sessions with overlapping trajectories and are subsequently fused into a single persistent object map (c).

4.3 Runtime

Runtime was measured over the seven Replica sequences on an Intel Core i7-13700KF CPU, 32 GB RAM, and an NVIDIA RTX 4090 GPU. The sequences comprise 14,000 RGB-D frames at 1200×680 resolution, from which ORB-SLAM3 selected 1,294 keyframes. The timings are reported in Table 2. ORB-SLAM3 tracking, which is executed for every input frame, averaged 12.48 ms per frame. Synchronous object-mapping processing (YOLO-World, SAM, CLIP, 3D projection and object matching) averaged 493.4 ms per keyframe. The dominant costs were SAM segmentation (158.9 ms) and CLIP embedding (224.6 ms). Object-map maintenance, including geometry rebuilding and duplicate-object merging, required an additional 204.1 ms of computation when normalized by the number of processed keyframes. This value is an amortized computational cost rather than synchronous keyframe latency, since maintenance operations are triggered separately as the map is updated. Based on the total processing time, including pipeline overheads, the system achieved an end-to-end throughput of 10.8 FPS across the seven Replica sequences. Per-scene throughput ranged from 6.5 to 32.8 FPS, reflecting differences in the number of keyframes generated and the resulting object-processing and map-maintenance workload.

Table 2: Mean runtime on the 7 Replica sequences

<i>Component</i>	<i>Time</i>
ORB-SLAM3 tracking	12.48 ms/frame
Synchronous KF processing	493.42 ms/KF
Map maintenance	204.12 ms/KF

5 Conclusion

In this paper, we presented a system for multi-session online open-vocabulary object mapping that integrates ORB-SLAM3 with a ConceptGraphs-based object-mapping frontend. By coupling object observations to SLAM keyframes, the system incrementally constructs an object map, updates its geometry as the SLAM estimate changes, and retains and reconciles object maps across mapping sessions.

The evaluation demonstrates that the system generally achieves accurate object localisation, although reconstruction of object spatial extent remains less consistent. Object association also remains a source of error, with evidence of both over-merging and fragmentation in the results. These errors can propagate to cross-session consolidation, since the reconciliation process operates on the object predictions produced independently in each session. Nevertheless, the multi-session experiments demonstrate that previously mapped objects can be retained, corresponding objects consolidated, and the object map extended over repeated deployments. Runtime analysis shows that object mapping introduces a significant computational cost relative to SLAM tracking.

Future work will initially address object reconstruction and association (spatial extent, fragmentation and overmerging). Further work will focus on a multithreaded architecture to reduce mapping overhead before extending the system towards simultaneous collaborative mapping, environmental change detection, application-specific relationships and hierarchies, ROS integration and support for the monocular and stereo configurations provided by ORB-SLAM3.

Acknowledgements

This paper has emanated from research conducted with the financial support of Taighde Éireann - Research Ireland under grant number GOIPG/2023/3321.

References

- Armeni, I., He, Z.-Y., Zamir, A., Gwak, J., Malik, J., Fischer, M., and Savarese, S. (2019). 3D scene graph: A structure for unified semantics, 3D space, and camera. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5663–5672.

- Campos, C., Elvira, R., Gómez, J. J., Montiel, J. M. M., and Tardós, J. D. (2021). ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map SLAM. *IEEE Transactions on Robotics*, 37(6):1874–1890.
- Chang, Y., Hughes, N., Ray, A., and Carlone, L. (2023). Hydra-Multi: Collaborative online construction of 3D scene graphs with multi-robot teams. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10995–11002.
- Cheng, T., Song, L., Ge, Y., Liu, W., Wang, X., and Shan, Y. (2024). YOLO-World: Real-time open-vocabulary object detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16901–16911.
- Gassol Puigjaner, A., Zacharia, A., and Alexis, K. (2026). Relationship-aware hierarchical 3D scene graph. In *2026 IEEE International Conference on Robotics and Automation (ICRA)*.
- Gorlo, N., Schmid, L., and Carlone, L. (2026). Describe anything anywhere at any moment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 35002–35013.
- Gu, Q., Kuwajerwala, A., Morin, S., Jatavallabhula, K. M., Sen, B., Agarwal, A., Rivera, C., Paul, W., Ellis, K., Chellappa, R., Gan, C., de Melo, C. M., Tenenbaum, J. B., Torralba, A., Shkurti, F., and Paull, L. (2024). ConceptGraphs: Open-vocabulary 3D scene graphs for perception and planning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5021–5028.
- Gu, Q., Ye, Z., Yu, J., Tang, J., Yi, T., Dong, Y., Wang, J., Cui, J., Chen, X., and Wang, Y. (2025). MR-COGraphs: Communication-efficient multi-robot open-vocabulary mapping system via 3D scene graphs. *IEEE Robotics and Automation Letters (RA-L)*, 10(6):5713–5720.
- Hughes, N., Chang, Y., and Carlone, L. (2022). Hydra: A real-time spatial perception system for 3D scene graph construction and optimization. In *Robotics: Science and Systems (RSS)*.
- Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. (2023). Segment anything. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3992–4003.
- Laina, S. B., Boche, S., Papatheodorou, S., Schaefer, S., Jung, J., and Leutenegger, S. (2026). FindAnything: Open-vocabulary and object-centric mapping for robot exploration in any environment. In *IEEE International Conference on Robotics and Automation (ICRA)*.
- Martins, T. B., Oswald, M. R., and Civera, J. (2025). Open-vocabulary online semantic mapping for SLAM. *IEEE Robotics and Automation Letters (RA-L)*, 10(11):11745–11752.
- McCormac, J., Clark, R., Bloesch, M., Davison, A., and Leutenegger, S. (2018). Fusion++: Volumetric object-level SLAM. In *International Conference on 3D Vision (3DV)*, pages 32–41.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*.
- Rosinol, A., Violette, A., Abate, M., Hughes, N., Chang, Y., Shi, J., Gupta, A., and Carlone, L. (2021). Kimera: from SLAM to spatial perception with 3D dynamic scene graphs. *The International Journal of Robotics Research (IJRR)*, 40(12-14):1510–1546.
- Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J. J., Mur-Artal, R., Ren, C., Verma, S., Clarkson, A., Yan, M., Budge, B., Yan, Y., Pan, X., Yon, J., Zou, Y., Leon, K., Carter, N., Briales, J., Gillingham, T., Mueggler, E., Pesqueira, L., Savva, M., Batra, D., Strasdat, H. M., Nardi, R. D., Goesele, M., Lovegrove, S., and Newcombe, R. (2019). The Replica dataset: A digital replica of indoor spaces.
- Sucar, E., Liu, S., Ortiz, J., and Davison, A. (2021). iMAP: Implicit mapping and positioning in real-time. In *Proceedings of the International Conference on Computer Vision (ICCV)*.
- Tourani, A., Ejaz, S., Bavle, H., Fernandez-Cortizas, M., Morilla-Cabello, D., Sanchez-Lopez, J. L., and Voos, H. (2026). vS-Graphs: Tightly coupling visual SLAM and 3D scene graphs exploiting hierarchical scene understanding. *IEEE Robotics and Automation Letters (RA-L)*, 11(8):9239–9246.
- Werby, A., Huang, C., Büchner, M., Valada, A., and Burgard, W. (2024). Hierarchical open-vocabulary 3D scene graphs for language-grounded robot navigation. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands.
- Whelan, T., Salas-Moreno, R. F., Glocker, B., Davison, A. J., and Leutenegger, S. (2016). ElasticFusion: Real-time dense SLAM and light source estimation. *The International Journal of Robotics Research (IJRR)*, 35(14):1697–1716.

Towards ISP-Free Fisheye Camera Perception: An Evaluation in Image and BEV Space

Krishna Prasath Kizhpada Vasudevan¹, Ganesh Sistu², Ciaran Eising¹, Venkatesh Satagopan³, Mihai Ilie², William O’Grady², Praveen Narasappareddygar³, and Hariharan Rangunathan³

¹*University of Limerick, Ireland*

²*Valeo Ireland*

³*Valeo India*

Abstract

Conventional autonomous perception pipelines rely heavily on ISP-processed RGB imagery, which is optimized for human visual aesthetics but frequently discards radiometric data critical for machine cognition. While prior RAW-domain research has demonstrated the benefits of sensor-domain learning, these studies predominantly address 2D image-space detection utilizing narrow-field rectilinear cameras. Modern autonomous driving systems, however, increasingly demand 3D Bird’s-Eye-View (BEV) spatial aggregation derived from complex surround-view fisheye cameras. This work systematically investigates ISP-free perception directly in the sensor domain using un-demosaiced 16-bit Bayer RAW automotive fisheye imagery. To support this investigation, we utilize the RAW-Det-Fisheye dataset, comprising 168,000 synchronized, multi-camera frames containing aligned 16-bit sensor RAW and ISP-processed RGB representations. To comprehensively evaluate sensor-domain resilience, we analyze performance across both 2D image-space detection using YOLOv8 and 3D BEV-space reasoning using BEVFormer equipped with multi-view feature aggregation. Experimental evaluations reveal that despite the severe non-linear radial distortion of fisheye lenses and the inherent sparsity of Bayer patterns, sensor-domain RAW representations achieve semantic detection and spatial localization performance comparable to conventional ISP-processed RGB baselines. These findings conclusively demonstrate that robust geometric consistency and semantic features are preserved prior to ISP intervention, motivating the development of highly efficient, sensor-native perception architectures tailored explicitly for machine vision.

Keywords: Fisheye Perception, Image Signal Processor Free Detection, Sensor-Native Learning, BEV Perception, Autonomous Driving, Raw Fisheye Dataset

1 Introduction

Modern autonomous driving systems increasingly rely on automotive fisheye cameras due to their ultra-wide field of view, enabling surround-view perception, near-field obstacle detection, and automated parking functionalities [Kumar et al., 2022]. However, fisheye imaging introduces severe radial distortion and spatially varying resolution, creating significant challenges for downstream perception algorithms. The measured RAW signal is

$$R_c(x, y) = g \cdot t \cdot I_c(x, y) + n_{shot} + n_{read} \quad (1)$$

where g is sensor gain, t is exposure time, and n_{shot} and n_{read} denote shot and read noise. Most automotive cameras employ a Bayer Color Filter Array (CFA), producing a sparse single-channel RAW representation. Conventional pipelines convert this sensor-domain data into RGB imagery through an Image Signal Processor (ISP):

$$I_{RGB} = f_{ISP}(R) \quad (2)$$

where f_{ISP} includes demosaicing, white balance, gamma correction, tone mapping, and denoising.

Although ISP processing improves visual quality for human observers, it may alter the radiometric fidelity of sensor measurements [Brooks et al., 2018, Li et al., 2024a]. Consequently, recent work has explored RAW-domain learning for image restoration and image-space perception [Chen et al., 2018, Xu et al., 2023, Cui and Harada, 2024, Morawski et al., 2022, Guo et al., 2025, Li et al., 2024a].

However, prior studies have largely focused on 2D image-space reasoning using conventional cameras. The effectiveness of sensor-domain RAW representations for Bird’s-Eye-View (BEV) perception under fisheye imaging remains largely unexplored. To address this gap, we investigate ISP-free autonomous perception using 16-bit Bayer RAW automotive fisheye imagery. Using the proposed RAW-Det-Fisheye dataset, YOLOv8 and BEVFormer, we evaluate whether semantic and geometric representations can be learned directly from sensor measurements.

2 Contributions

The primary contributions of this work are summarized as follows:

1. **ISP-Free Fisheye Perception Paradigm:** We pioneer an ISP-free perception pipeline tailored specifically for complex automotive fisheye cameras. We demonstrate that machine cognition can operate directly on mosaiced, 16-bit Bayer RAW sensor measurements, effectively bypassing the radiometric information loss and hallucination artifacts inherent in human-centric ISP pipelines.
2. **Dual-Space Architectural Evaluation:** We provide the first systematic evaluation of sensor-domain robustness across two distinct spatial paradigms: 2D image-space reasoning (utilizing YOLOv8) and 3D multi-view Bird’s-Eye-View (BEV) perception (utilizing BEVFormer). This effectively addresses both the classical camera-based and modern BEV-based perception applications. To our knowledge, this is the first work to evaluate the ISP influence on BEV tasks.
3. **RAW-Det-Fisheye Dataset:** We evaluated the models on a comprehensive, large-scale industry grade automotive dataset comprising 168,000 synchronized frames of aligned 16-bit sensor RAW and ISP-processed RGB representations. By maintaining identical ground-truth bounding box annotations across domains, this dataset enables tightly controlled, rigorous evaluation of sensor-domain learning against traditional imaging pipelines.
4. **Demonstration of Geometric Resilience:** Our empirical findings establish that multi-view spatial aggregation and 3D geometric projection can successfully occur prior to ISP intervention. We prove that sensor-domain RAW representations preserve sufficient semantic structure and geometric consistency to achieve detection and localization performance comparable to conventionally optimized RGB baselines.

3 State of the Art

3.1 RAW-Domain Learning and Neural ISPs

Traditional Image Signal Processors (ISPs) rely on handcrafted, sequential modules designed to produce visually appealing RGB images for human consumption. However, this human-centric optimization often destroys low-level radiometric details essential for machine cognition. Early RAW-domain research addressed this by developing deep learning models for extreme low-light imaging [Chen et al., 2018] and introducing inverse ISP modeling to recover sensor data from sRGB images [Brooks et al., 2018]. Subsequent works explored mapping RAW signals to sRGB domains using neural networks [Zhang et al., 2021] or explicitly optimizing

Neural ISPs for downstream machine perception tasks [Morawski et al., 2022]. More recently, approaches like Dark-ISP [Guo et al., 2025] have further enhanced RAW image processing specifically to boost low-light object detection. While these methods demonstrate the value of preserving RAW data, they generally still rely on an intermediate processing step to reconstruct a dense representation before applying perception networks, adding unnecessary computational overhead.

3.2 RAW-Domain Perception in Image Space

To eliminate the computational bottleneck of explicit ISP reconstruction, recent literature has shifted toward direct ISP-free perception. Foundational works have established new benchmarks and models specifically for 2D object detection directly on raw sensor measurements [Xu et al., 2023, Li et al., 2024b]. To leverage the vast knowledge in models pre-trained on RGB data, techniques like RAW-Adapter [Cui and Harada, 2024] have been introduced to efficiently adapt standard visual models to camera RAW inputs without full retraining. Furthermore, direct processing of un-demosaiced snapshots has been shown to significantly improve overall visual computing efficiency [Li et al., 2024a]. These studies collectively prove that neural networks can extract robust semantic features directly from sparse Bayer-patterned data.

3.3 Bridging the Gap: From 2D to BEV-Space Fisheye Perception

Despite the success of RAW-domain learning in 2D image space, modern autonomous driving pipelines increasingly depend on 3D Bird’s-Eye-View (BEV) perception architectures. BEV perception requires rigorous geometric feature projection, multi-view spatial aggregation, and 3D reasoning. Performing these complex operations directly on sparse, un-demosaiced Bayer data presents a fundamentally different challenge than standard 2D detection, particularly when utilizing automotive fisheye cameras characterized by extreme non-linear radial distortion.

Paradigm	Processing Pipeline	Limitations Addressed by Our Work
Traditional HW ISP	RAW $\rightarrow$ HW ISP $\rightarrow$ RGB $\rightarrow$ Detect	Discards linear radiometric data; demosaicing hallucinates edges; optimized for humans, not machines.
Neural / Task-ISP	RAW $\rightarrow$ Neural ISP $\rightarrow$ RGB $\rightarrow$ Detect	Retains dependency on dense intermediate reconstruction, leading to high latency.
2D RAW Detection	RAW $\rightarrow$ 2D Object Detection	Lacks 3D geometric reasoning; primarily evaluated on rectilinear, narrow-field lenses.
Proposed Framework	Fisheye RAW $\rightarrow$ BEV/2D Detect	Investigates direct 3D BEV spatial aggregation under extreme fisheye distortion without demosaicing.

Table 1: Comparison of traditional, neural ISP, and sensor-direct perception pipelines.

As summarized in Table 1, prior research has predominantly focused on rectilinear 2D tasks. This work explicitly targets the unexplored intersection of ISP-free perception, fisheye geometry, and 3D BEV spatial aggregation. By demonstrating that sensor-domain RAW representations remain geometrically stable during complex BEV transformations, our research bridges the critical gap between raw sensor physics and state-of-the-art autonomous driving architectures.

4 The RAW-Det-Fisheye Dataset

To rigorously evaluate the efficacy of ISP-free perception in autonomous driving, we evaluate using the **RAW-Det-Fisheye** dataset. Existing autonomous driving benchmarks predominantly provide heavily compressed, ISP-processed RGB imagery, while established RAW-domain datasets are largely restricted to narrow-field,

rectilinear cameras. This dataset bridges that critical gap by providing large-scale, synchronized multi-camera fisheye imagery specifically curated for sensor-domain machine perception.

4.1 Dataset Composition and Sensor Characteristics

The dataset comprises a total of 168,000 frames, carefully partitioned into 160,000 training frames and 8,000 validation frames. The imagery is captured using automotive-grade surround-view fisheye cameras characterized by extreme non-linear radial distortion Figure 1. The dataset was collected directly from automotive image sensors in RAW Bayer format prior to ISP processing. The Operational Design Domain includes urban, suburban, parking and highway driving environments containing vehicles, pedestrians, cyclists, road infrastructure, and traffic control devices. Data were recorded under diverse illumination conditions ranging from bright daylight to low-light nighttime scenes to evaluate perception performance across a broad spectrum of sensor operating conditions. The dataset consists of 16-bit Bayer RAW images in BGGR format captured using a Dual Conversion Gain (DCG) HDR automotive fisheye sensor. Native sensor frames are acquired at a resolution of 1632×1290 pixels and subsequently downsampled to 1024×864 while preserving the Bayer mosaic structure. The downsampling operation is performed directly on the RAW data without demosaicing or color interpolation, ensuring that the original CFA sampling pattern is maintained.

Data was collected at a frequency of 10 Hz over 50 hours of recordings. The data is geographically diverse, ensuring the model’s robustness across different regional infrastructures, with the distribution: **Czech Republic: 44%, UK & Republic of Ireland: 22%, USA: 17%, Germany: 13%, Denmark: 1.5%, Others: 2.5%.**

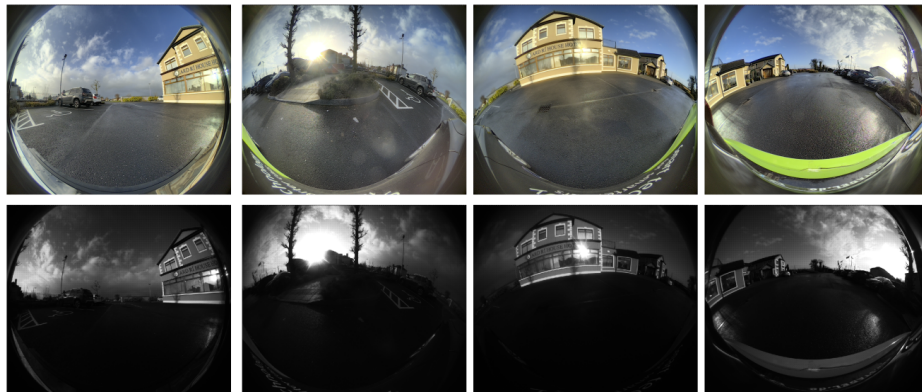


Figure 1: ISP Processed and RAW Bayer Fisheye images from our Dataset. Raw images are enhanced for visualization purposes.

More samples and quality results will be shared as a Supplementary material and will be uploaded to the arxiv

4.2 Multi-Domain Aligned Representations

A core feature of the RAW-Det-Fisheye dataset is the strict temporal and spatial alignment of multiple imaging domains for every captured frame. This enables perfectly controlled comparisons between sensor-native processing and conventional pipelines. Each frame is provided in two distinct representations: (i) The direct, 16-bit linear radiometric output from the image sensor, preserving the original dynamic range and containing inherent physical noise (e.g., shot and read noise) prior to any ISP intervention (ii) The conventionally processed 32-bit, 3-channel sRGB output generated via the camera’s hardware ISP. This serves as the primary baseline representing human-optimized perception.

4.3 Annotation Consistency

To ensure that empirical performance variations are exclusively attributable to the input data representation rather than label discrepancies, ground-truth bounding box annotations are rigidly identical across both do-

mains. This strict alignment guarantees that subsequent evaluations in both 2D image-space and 3D BEV-space accurately reflect the semantic and geometric resilience of the underlying pixel representations.

5 Proposed Methodology

To rigorously evaluate the viability of ISP-free perception, we deploy two state-of-the-art architectures: YOLOv8 [Jocher et al., 2023] for 2D image-space reasoning and BEVFormer [Li et al., 2022] for 3D multi-view spatial aggregation. By isolating the perception models from the ISP, we can directly assess the neural network’s ability to interpret raw sensor physics.

5.1 Image-Space Perception Using YOLOv8

Image-space perception evaluates semantic understanding directly in 2D sensor coordinates. We utilize YOLOv8 [Jocher et al., 2023], a highly efficient single-stage object detector featuring a modified CSPDarknet backbone, a Path Aggregation Network (PANet) neck, and an anchor-free detection head. For our ISP-free baseline, YOLOv8 is trained directly on the single-channel Bayer mosaiced RAW input without demosaicing or any ISP preprocessing:

$$X_{\text{RAW}} \in \mathbb{R}^{H \times W \times 1} \quad (3)$$

To prevent the model from inheriting biases optimized for dense sRGB imagery, and to preserve the physical properties of the sensor data, we applied the following specific training configurations. The training was started from scratch with no pretrained RGB weights. Mosaic augmentation is disabled to preserve contiguous Bayer mosaicing patterns and localized physical noise distributions. Automatic Mixed Precision (AMP) is disabled to maintain the full 16-bit radiometric depth of the sensor measurements. Input resolution is 1024x864 pixels with a single-channel RAW input.

5.2 BEV-Space Perception Using BEVFormer

Unlike 2D detection, BEV-space perception demands rigorous geometric feature consistency across multi-view spatial projections. We utilize BEVFormer [Li et al., 2022], a transformer-based architecture that aggregates spatiotemporal features from multiple cameras into a unified Bird’s-Eye-View grid using spatial cross-attention.

A critical challenge in adapting BEVFormer for our dataset is that standard pinhole camera projection models fail under extreme fisheye distortion. To resolve this, we modified the spatial cross-attention (deformable attention) module to incorporate a 4th-order polynomial camera model, inspired by the WoodScape dataset geometry [Yogamani et al., 2019]. This custom geometric projection ensures that 3D reference points in the BEV space are accurately mapped to their corresponding highly distorted 2D pixel coordinates on the image plane.

Aside from adapting the spatial projection geometry, BEVFormer was modified to directly process single-channel 16-bit Bayer RAW TIFF images by changing only the first convolution layer of the ResNet-101 backbone from three to one input channel. Both RGB and RAW models were trained from scratch without pretrained weights using identical optimizer settings, training schedules, loss functions, and network architecture, comprising a ResNet-101 with DCNv2, an FPN, a BEV Encoder, and a DETR Decoder for predicting a 78×78 BEV grid of Vehicle and Pedestrian classes. Evaluation was performed using only the current frame, without temporal history, thereby isolating the influence of the sensor-domain representation.

6 Experimental Results

6.1 Image-Space Detection Resilience

Table 2 presents a quantitative comparison of YOLOv8 detection performance across ISP-processed RGB and real sensor-domain RAW representations and qualitative results of RAW and ISP processed images are shown in figure 2 and figure 3 respectively

Metric	ISP-Processed RGB	Real Sensor RAW
Vehicle Precision	0.84	0.79
Vehicle Recall	0.44	0.37
Vehicle F1-score	0.58	0.503
False Positives / Frame	0.95	1.08
mAP@0.50	0.525	0.513

Table 2: Quantitative comparison at conf=0.25 of YOLOv8 2D image-space detection across imaging domains.

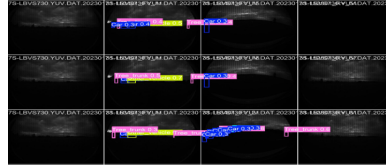


Figure 2: Image-space detection inference using YOLOv8 in real 16-bit Bayer RAW automotive fisheye imagery.

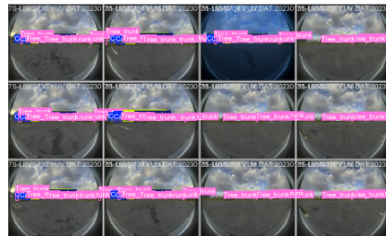


Figure 3: Image-space detection inference using YOLOv8 in ISP processed RGB automotive fisheye imagery.

The empirical evaluations demonstrate that sensor-domain RAW representations achieve semantic detection parity with conventional ISP pipelines. Specifically, RAW input yields a Vehicle F1-score of 0.503, close to the 0.58 achieved by the RGB baseline. Also the precision and recall for the RAW and RGB are close to each other. The RAW domain exhibits a robust $mAP@0.50=0.513$ which is remarkably closer to RGB $mAP@0.50=0.525$ and the resilient recall rates confirm that the fundamental semantic identity of objects is preserved prior to ISP intervention. The relatively low recall observed for both RGB and RAW models reflects the challenging nature of automotive fisheye object detection, characterized by severe radial distortion and large object-scale variation. Since both models were trained and evaluated under identical conditions, the comparison fairly isolates the effect of the input representation.

6.2 BEV-Space Geometric Stability

The most significant findings of this study emerge in the 3D BEV-space perception tasks. Table 3 summarizes the performance of the modified BEVFormer architecture operating under severe fisheye distortion and qualitative results of RAW and ISP processed images are shown in Figure 4



Figure 4: BEV-space detection inference using BevFormer in real 16-bit Bayer RAW and ISP processed Image in automotive fisheye imagery dataset. The left side shows inference on RAW image and right side shows inference on ISP processed image

Metric	ISP-Processed RGB	Real Sensor RAW
Vehicle Precision	0.726	0.726
Vehicle Recall	0.867	0.874
Pedestrian Precision	0.244	0.235
Pedestrian Recall	0.276	0.276
Average Orientation Error (AOE) Vehicle	0.040	0.041
Average Orientation Error (AOE) Pedestrian	0.394	0.417
Direct Corner Error Vehicle	0.599	0.581
Closest Corner Error Vehicle	0.583	0.566

Table 3: Comparison of BEVFormer @conf=0.3 multi-view 3D perception performance.

Despite the necessity for complex spatial projection and multi-view temporal feature aggregation, un-demosaiced RAW inputs achieve identical precision (0.726) and slightly superior recall (0.874) for vehicle detection compared to the RGB baseline. Furthermore, the geometric localization capabilities—measured via Average Orientation Error (AOE) and 3D corner errors—remain phenomenally stable. In fact, RAW representations marginally outperformed RGB in Direct Corner Error (0.581 vs. 0.599), indicating that the preservation of linear radiometric data and the absence of demosaicing artifacts actually aid the spatial cross-attention mechanisms in triangulating precise 3D geometric coordinates.

7 Discussion and Analysis of ISP Effects

The comparable, and in some metrics superior, performance of sensor-domain representations fundamentally challenges the necessity of hardware ISPs in autonomous perception pipelines. Traditional ISP operations—such as gamma compression and tone mapping—are mathematically non-linear transformations designed to map wide-dynamic-range scenes into a restricted, human-pleasing color space. While aesthetically beneficial, these operations structurally distort the linear relationship between scene radiance and pixel intensity.

Our findings demonstrate that state-of-the-art architectures equipped with deformable attention mechanisms can directly decode sparse, 16-bit physical sensor measurements. The profound robustness of RAW representations in 3D BEV space proves that the non-linear projection algorithms required to untangle severe radial fisheye distortion are mathematically compatible with raw Bayer sampling. By entirely bypassing the hardware ISP, autonomous driving systems can effectively eliminate the latency, power consumption, and hallucination artifacts associated with demosaicing, directly translating sensor-native measurements into reliable semantic and geometric intelligence.

8 Conclusion

This paper systematically investigates the paradigm of ISP-free autonomous perception, extending RAW-domain learning beyond rectilinear 2D detection into the rigorous domain of 3D Bird’s-Eye-View (BEV) spatial aggregation utilizing automotive fisheye cameras. By introducing the RAW-Det-Fisheye dataset, we facilitated a strictly controlled evaluation between 16-bit un-demosaiced sensor measurements and conventionally optimized RGB imagery.

Our comprehensive experiments using YOLOv8 and BEVFormer empirically establish that linear, sensor-domain representations preserve critical semantic structure and yield highly accurate 3D geometric localization, successfully navigating the compounded challenges of Bayer sparsity and severe radial distortion. This work strongly motivates a paradigm shift toward highly efficient, low-latency, sensor-native perception architectures jointly optimized for raw physics and machine intelligence, rather than human visual aesthetics. While our

current evaluation primarily focuses on the major dynamic object categories—such as vehicles and pedestrians—that are most critical for immediate crash prevention and automated braking systems, these promising baseline results open exciting avenues for future research. Moving forward, we plan to extend this ISP-free paradigm to traffic control elements, including traffic signs and traffic lights, as well as smaller, color-pattern-dependent objects like traffic cones and bollards. By demonstrating that core spatial and semantic intelligence is already robustly preserved in the RAW domain, this work lays a strong foundation for optimizing sensor-native networks across all edge cases, ultimately opening up the possibility of utilizing raw sensor data for full-scale, comprehensive autonomous driving applications.

References

- [Brooks et al., 2018] Brooks, T., Mildenhall, B., Xue, T., Chen, J., Sharlet, D., and Barron, J. T. (2018). Un-processing images for learned raw denoising.
- [Chen et al., 2018] Chen, C., Chen, Q., Xu, J., and Koltun, V. (2018). Learning to see in the dark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Cui and Harada, 2024] Cui, Z. and Harada, T. (2024). Raw-adapter: Adapting pre-trained visual model to camera raw images. *arXiv*.
- [Guo et al., 2025] Guo, J., Gao, X., Yan, Y., Li, G., and Pu, J. (2025). Dark-isp: Enhancing raw image processing for low-light object detection. *ICCV 2025*.
- [Jocher et al., 2023] Jocher, G., Chaurasia, A., and Qiu, J. (2023). Yolo by ultralytics.
- [Kumar et al., 2022] Kumar, V. et al. (2022). Fisheye cameras for autonomous driving: Survey, architecture and challenges. *arXiv preprint arXiv:2205.13281*.
- [Li et al., 2024a] Li, Z., Lu, M., Zhang, X., Feng, X., Asif, M. S., and Ma, Z. (2024a). Efficient visual computing with camera raw snapshots. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46:4684–4701.
- [Li et al., 2022] Li, Z., Wang, W., Li, H., Xie, E., Sima, C., Lu, T., Qiao, Y., and Dai, J. (2022). Bevformer: Spatial-temporal transformer for 3d object detection from multi-camera images. In *European conference on computer vision*, pages 1–18. Springer.
- [Li et al., 2024b] Li, Z.-Y., Jin, X., Sun, B., Guo, C.-L., and Cheng, M.-M. (2024b). Towards raw object detection in diverse conditions. *arXiv*.
- [Morawski et al., 2022] Morawski, I., Chen, Y.-A., Lin, Y.-S., Dangi, S., He, K., and Hsu, W. H. (2022). Genisp: Neural isp for low-light machine cognition. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 629–638.
- [Xu et al., 2023] Xu, R., Chen, C., Peng, J., Li, C., Huang, Y., Song, F., Yan, Y., and Xiong, Z. (2023). Toward raw object detection: A new benchmark and a new model. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13384–13393.
- [Yogamani et al., 2019] Yogamani, S., Hughes, C., Horgan, J., et al. (2019). Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9308–9318.
- [Zhang et al., 2021] Zhang, Z., Wang, H., Liu, M., Wang, R., Zhang, J., and Zuo, W. (2021). Learning raw-to-srgb mappings with inaccurately aligned supervision.

A Domain and Sensor-Adaptive Metric Road Width Estimation from Dual Rear Cameras via Transfer Learning

Zuha Tassawar Hussain ¹, Waqar Shahid Qureshi ¹, Nazre Batool ¹, Syed Ibrahim Hassan ²,
David Power ², and Kieran Feighan ²

¹*University of Galway, Ireland*

²*Pavement Management Services Ltd., Ireland*

Abstract

An accurate measurement of road geometry is important for road infrastructure and maintenance, safety analysis and autonomous systems. While image-based approaches offer an economical substitute to LiDAR, these approaches struggle on roads with degraded or absent lane markings, and rely on forward facing cameras which are highly receptive to occlusions. This paper introduces a novel, domain-, and sensor-adaptive framework for metric road width estimation using dual rear-view cameras on operational road (pavement) survey vehicles. We propose a system based on a fine-tuned road segmentation model with trajectory-aware temporal smoothing and multi-view image fusion via calibrated projections. Extensive validation indicates that our system produces dense, stable, and metrically accurate road width profiles that outperform the marking dependent baselines, especially on unmarked roads. Our system bridges vision-based analysis and infrastructure-grade monitoring offering a practical and high-throughput solution for road network-scale pavement assessment without any additional hardware.

Keywords: Road Width Estimation, Multi-view Fusion, Geometric Reconstruction

1 Introduction

A precise road geometry measurement, particularly metric width, is essential for the maintenance and safety of modern transportation systems. While vision-based sensing is a scalable and cost-effective alternative to LiDAR, perspective distortions, occlusions and degraded (or absent) lane markings make road width estimation challenging. Most common approaches to infer road width through lane marking separation fail on unmarked or non-standard roads[Mahlberg et al., 2022]. Approaches using forward-facing systems (e.g dash cams) are also prone to occlusion and are poorly suited for pavement inspection. In contrast, dual rear-view cameras mounted on survey vehicles offer a stable and unobstructed view right behind the vehicle. However, road segmentation deep learning models trained on public benchmarks suffer from a severe domain shift on rear-view images and poorly marked lanes. Furthermore, per-frame processing of camera video streams leads to temporally unstable predictions.

Road boundary detection has evolved from classical gradient-based edge detection and geometric fitting (e.g., Hough Transform) [Zou et al., 2019] to deep learning approaches. While classical methods were enhanced with probabilistic and temporal filtering, they remained brittle in real-world conditions. Learning-based models such as SegNet [Paszke et al., 2016] improved semantic understanding but are predominantly trained on forward-facing urban datasets and degrade on rear-view imagery of marking-deficient rural roads.

Surround-view systems fuse multiple cameras for 360° road-edge detection [Dahal et al., 2021], but are designed for obstacle avoidance rather than metric infrastructure assessment [Yuan et al., 2023, Hecker et al., 2018]. Vision Transformers have recently advanced segmentation, edge detection [Pu et al., 2022], and line extraction

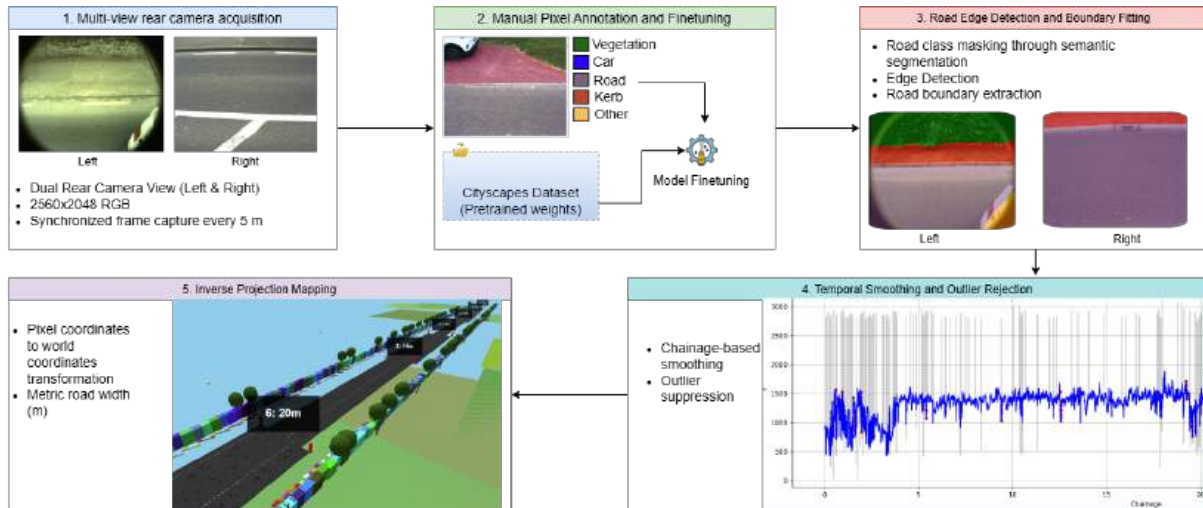


Figure 1: Overview of the proposed methodology for metric road width estimation using dual rear-view cameras. The framework integrates semantic segmentation, geometric boundary estimation, inverse perspective projection, and trajectory-aware temporal refinement.

[Xu et al., 2021], yet models pre-trained on urban data struggle with the appearance and perspective variations of rear-view pavement imagery. Metric width estimation commonly relies on inverse perspective mapping (IPM) with camera calibration and RANSAC fitting [Zou et al., 2019, Ali et al., 2020, Dahal et al., 2021], but assumes clear lane markings [Lu et al., 2025], while temporal models such as SCNNs and ConvLSTM [Dong et al., 2023, Zhang et al., 2021] have been validated primarily on forward-facing video streams.

Our work addresses these limitations through an integrated pipeline that (1) adapts transformer-based segmentation to the rear-view pavement domain, (2) fuses dual rear-camera observations for marking-independent width estimation, and (3) applies trajectory-aware temporal smoothing to produce stable metric profiles. To the best of our knowledge, ours is the first system combining domain-adaptive transformer-based segmentation, dual rear-view fusion, and trajectory-aware estimation for network-level pavement assessment.

2 Methodology

The proposed system operates on an operational pavement survey vehicle equipped with a calibrated dual rear-facing RGB camera rig mounted on a rigid gantry structure, as illustrated in Fig. 2. We propose an end-to-end system for metric road width estimation using synchronized dual rear-view cameras mounted on an operational pavement survey vehicle. The proposed pipeline integrates domain-adaptive semantic segmentation, geometric boundary modeling, metric reconstruction through camera calibration, and trajectory-aware temporal refinement. An overview of our system is illustrated in Fig. 1.

2.1 Sensor Configuration and Data Acquisition

Two cameras are symmetrically positioned on the left and right sides of the vehicle at a fixed height above the road surface and oriented outward with a slight downward pitch, enabling unobstructed observation of pavement edges immediately behind the vehicle. Compared to conventional forward-facing perception systems, this rear-view configuration reduces occlusions from traffic and headlight glare while providing stable post-passage visibility of pavement boundaries. This geometry aligns naturally with *chainage*-

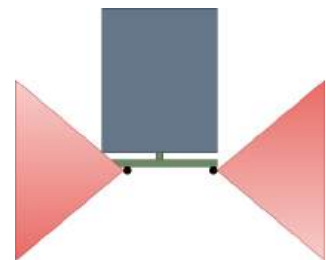


Figure 2: Schematic of the dual rear-view camera configuration mounted on the pavement survey vehicle.

based pavement survey workflows, where measurements are referenced to physical location along the traveled trajectory.

Each camera captures synchronized high-resolution imagery at 2560×2048 pixels, ensuring temporally aligned left–right image pairs for accurate lateral reconstruction. Images are acquired at fixed spatial intervals along the vehicle trajectory, providing consistent coverage for downstream geometric processing. To address the domain gap between publicly available urban datasets and rear-view pavement imagery, a task-specific dataset was constructed from real-world survey dataset covering diverse roads, including urban, rural, and maintenance scenarios. Pixel-level annotations were generated for five semantic categories: road, kerb, vegetation, vehicle, and background.

2.2 Domain-Adaptive Semantic Segmentation

Let $I_t^{(l)}$ and $I_t^{(r)}$ denote synchronized left and right camera images captured at time index t . A semantic segmentation network $f_\theta(\cdot)$ maps each image to dense per-pixel class likelihoods:

$$S_t^{(i)} = f_\theta(I_t^{(i)}), \quad i \in \{l, r\}, \quad (1)$$

$$S_t^{(i)} \in \mathbb{R}^{H \times W \times |\mathcal{C}|} \quad (2)$$

where $|\mathcal{C}|$ represent the total number of semantic classes and the final semantic prediction is obtained as:

$$M_t^{(i)}(x, y) = \arg\max_{c \in \mathcal{C}} S_t^{(i)}(x, y, c). \quad (3)$$

Pixels corresponding to the pavement boundary are extracted by identifying road-class pixels adjacent to a differently-labelled pixel within a local neighbourhood $\mathcal{N}$:

$$E_t^{(i)} = \{(x, y) \mid M_t^{(i)}(x, y) = \text{road}, \exists (x', y') \in \mathcal{N}(x, y) : M_t^{(i)}(x', y') \neq \text{road}\},$$

where jointly considering transitions from road to kerb, vegetation, or other classes keeps the method robust in scenarios where explicit kerb structures are absent, degraded, or partially occluded, while still preserving strong geometric cues for boundary localization.

2.3 Geometric Road Boundary Estimation

For each camera view, the extracted pavement boundary pixels $\mathcal{E}_t^{(i)}$ are approximated using a single straight line in image space. Rather than relying on dense contour extraction, a line-based representation provides a compact and geometrically stable description of road edge, which is suitable for metric projection. A candidate road boundary is represented as

$$\ell_t^{(i)} : a_t^{(i)}x + b_t^{(i)}y + c_t^{(i)} = 0, \quad (4)$$

where x and y denote image coordinates. Due to incorrect decisions caused by shadow, vegetation, or other occlusions, the line parameters are estimated using a robust fitting procedure. This is done by minimizing a robust geometric error over the extracted boundary pixels:

$$\ell_t^{(i)} = \arg\min_{\ell} \sum_{(x_j, y_j) \in \mathcal{E}_t^{(i)}} \rho(|ax_j + by_j + c|), \quad (5)$$

where $\rho(\cdot)$ is a truncated squared loss implemented via RANSAC: residuals below a threshold $\tau = 5$ pixels are treated as inliers and refit by least squares, while larger residuals are excluded, over 3000 trial hypotheses, suppressing outliers arising from shadows, debris, vegetation intrusion, and transient occlusions. After estimating the boundary line, a single boundary observation is obtained by traversing the fitted line from the bottom of the

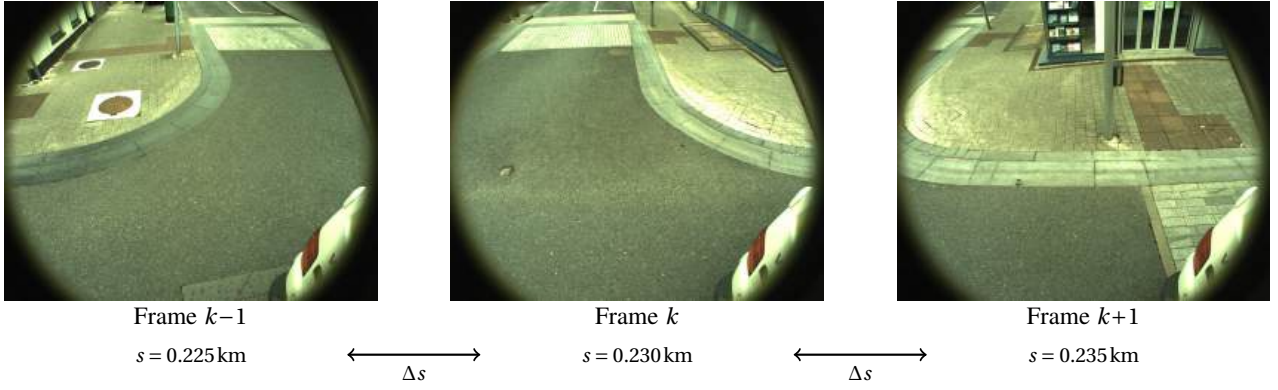


Figure 3: Illustration of trajectory-aware temporal refinement. Rear-view frames are captured at regular spatial intervals (every ~ 5 m) along the vehicle trajectory. For a reference chainage location s_t , width measurements within a spatial neighborhood of length $2\Delta s$ are aggregated, with $\Delta s = 5$ m matching the inter-frame spacing. This chainage-based formulation enables physically meaningful smoothing that is invariant to vehicle speed and frame sampling rate.

image upward and selecting the first row with sufficient local inlier support. Let H denote the image height, and let $x(y)$ denote the fitted line's x -coordinate at row y :

$$x(y) = -\frac{b_t^{(i)}y + c_t^{(i)}}{a_t^{(i)}}. \quad (6)$$

Starting from $y = H$ and stepping upward, the fitted line is evaluated row by row until a row with sufficient nearby boundary support is found, yielding a single point $p_t^{(i)}$ as the image-space input for metric projection. The search begins at $y = H$ and moves upward because pixels farther from the camera correspond to larger, less-resolved ground patches under inverse perspective projection, so rows closest to the camera are preferred whenever reliable; requiring local inlier support before accepting a row makes the selection robust to localized occlusion or ambiguity, deferring upward only when necessary.

2.4 Metric Reconstruction via Inverse Perspective Projection

Using intrinsic and extrinsic camera calibration, the image-space boundary points $p_t^{(i)}$ obtained in Section 2.3 are projected onto the road surface under a locally planar assumption, with $p_t^{(i)} = (u_t^{(i)}, v_t^{(i)})$. Let $\mathbf{H} = \mathbf{K}[\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{t}]$ denote the ground-plane homography, where $\mathbf{K}$ is the intrinsic matrix and $(\mathbf{r}_1, \mathbf{r}_2, \mathbf{t})$ are obtained from extrinsic calibration.

For each trajectory index t , the left and right image-space boundary observations, $(u_t^{(l)}, v_t^{(l)})$ and $(u_t^{(r)}, v_t^{(r)})$, are mapped to the ground plane using the inverse homography. The metric road width is then computed as the Euclidean distance between the projected boundary points:

$$W_t = \left\| \pi\left(\mathbf{H}^{-1}[u_t^{(l)}, v_t^{(l)}, 1]^\top\right) - \pi\left(\mathbf{H}^{-1}[u_t^{(r)}, v_t^{(r)}, 1]^\top\right) \right\|_2. \quad (7)$$

where $\pi(\cdot)$ denotes normalization of homogeneous coordinates. This produces a sequence of raw road-width estimates along the vehicle trajectory.

2.5 Trajectory-Aware Temporal Refinement

During pavement surveys, images are acquired at fixed spatial intervals along the trajectory. In this configuration, synchronized rear-view frames are captured approximately every 5 m of travel, so consecutive frames provide overlapping views of the same road segment. This introduces a key prior: pavement width varies smoothly

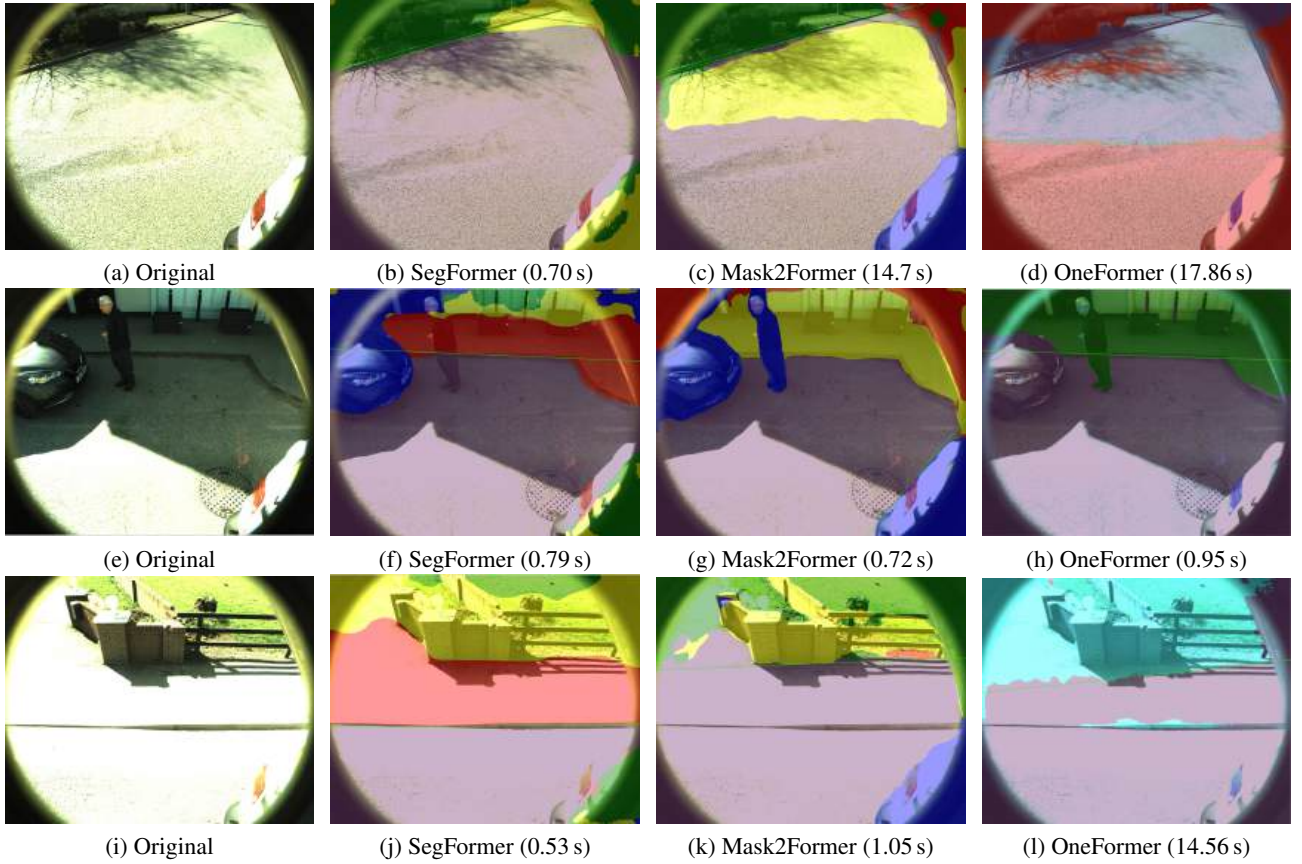


Figure 4: Qualitative comparison of semantic segmentation models with their inference time in seconds. Rows correspond to left rear-view pavement scenes, while columns show the original image and predictions from SegFormer, Mask2Former, and OneFormer. Green lines indicate RANSAC-fitted pavement boundaries fitted to road edge pixels, and red circles denote observation points for geometric evaluation. Quantitatively, SegFormer achieves the highest performance on the rear-view pavement dataset (mIoU: 0.957, mDice: 0.978, Road IoU: 0.995), substantially outperforming Mask2Former (mIoU: 0.465, mDice: 0.581, Road IoU: 0.872) and OneFormer (mIoU: 0.013, mDice: 0.025, Road IoU: 0.051).

along distance and does not change abruptly over short spatial extents under normal conditions. Therefore, neighboring frames contain overlapping geometric information that can improve robustness.

Let s_t denote the chainage (cumulative travel distance) for frame t . Since measurements are referenced to physical location rather than time, refinement is performed in the trajectory domain, ensuring invariance to vehicle speed, frame rate variations, and stop–start motion.

Instantaneous width estimates are susceptible to segmentation uncertainty, geometric amplification during inverse projection, and localized disturbances (e.g., junctions, occlusions). To enforce spatial consistency, width measurements are aggregated within a chainage-aligned neighborhood.

$$\mathcal{W}_t = \{W_k \mid |s_k - s_t| \leq \Delta s\}, \quad (8)$$

where Δs defines the spatial window length. This formulation is illustrated in Fig. 3. Unlike conventional temporal filtering based on frame order, this method aggregates measurements over a fixed physical distance, making refinement directly interpretable for infrastructure analysis. Since neighboring frames are captured every 5 m, $\mathcal{W}_t$ represents multiple observations of the same pavement geometry.

A robust width estimate is obtained by minimizing the ℓ_1 deviation within the local neighborhood.

$$\hat{W}_t = \underset{w}{\operatorname{argmin}} \sum_{W_k \in \mathcal{W}_t} |W_k - w|, \quad (9)$$

The final width is computed by aggregating width estimates within a local chainage neighborhood. This trajectory-aware refinement suppresses isolated outliers while preserving genuine geometric variations, producing smooth, physically consistent width profiles suitable for chainage-based pavement assessment.

3 Experiments and Results

3.1 Segmentation Model Evaluation

As the first and most critical stage of our pipeline, semantic segmentation accuracy directly impacts downstream geometric fitting and temporal aggregation. Therefore, we evaluated different models not only on pixel-level metrics, but also on boundary continuity, geometric stability, and computational efficiency, all essential for operational deployment. We compared three state-of-the-art models: SegFormer [Xie et al., 2021], Mask2Former [Zhang et al., 2022], and OneFormer [Jain et al., 2023], on 201 manually annotated rear-view images (102 left, 99 right) spanning five classes (road, vegetation, kerb, vehicle, background), split 80/20 into training and test sets. Fine-tuned from Cityscapes-pretrained weights for 50 epochs; all models were evaluated on the test set under identical conditions on the same GPU without any test-time augmentation.

Fig. 4 shows qualitative results with RANSAC-fitted boundaries (green lines). SegFormer consistently produces smooth, coherent road masks with well-defined edges, giving a stable boundary even under shadows, occlusions, and perspective distortion. In contrast, Mask2Former gives fragmented masks near boundaries, leading to discontinuous edge points and unstable RANSAC fits. OneFormer suffers from severe class confusion, preventing meaningful boundary extraction quite often. Based on these model evaluations, we selected SegFormer for our segmentation.

3.2 Road-Width Estimation and Results

Following model selection, the performance of our road width estimation pipeline is evaluated through a large-scale comparison between historical data collected in 2018 and recent measurements in 2025. In 2018, a single road width was recorded for each 100 m sample unit, whereas in 2025, twenty estimates of rear-view image pairs are acquired per 100 m interval at regular spacing along the vehicle trajectory. Due to this substantial difference in acquisition protocol, direct point-to-point comparison is not meaningful. Instead, we observe geometric consistency, spatial behavior, and network-level agreement using aggregated statistics for each section.

Fig. 5a shows a comparison of historical road widths (2018) vs. estimated widths by our system. It can be observed that the estimated widths are consistently lower than historical measurements while preserving the relative ordering between road sections, indicating that the framework captures meaningful geometric variation rather than converging to a nominal width.

This systematic offset likely reflects a combination of factors: the 2018 and 2025 measurements were captured using different camera systems with independent orientations and calibrations, so any bias in mounting height, pitch angle, or intrinsic parameters propagates directly into the projected width without a shared reference to cross-check consistency; and the seven-year interval between surveys leaves open the possibility of genuine physical change in pavement geometry over time. The consistency and directionality of the offset across sections suggests a systematic rather than random cause, which we intend to isolate through cross-validation against high-accuracy ground truth in future work.

Beyond aggregated averages, the dense measurements enable continuous longitudinal analysis along chainage. As illustrated in Fig. 5b, the estimated width profile closely follows historical trend while exhibiting smooth and stable behavior, which is essential for pavement management systems that rely on consistent chainage based indicators for segmentation and maintenance planning.

This strong alignment empirically validates the trajectory-aware temporal refinement strategy, which aggregates measurements over fixed physical distances rather than frame indices to produce geometrically consistent width estimates. Overall, our system delivers stable, infrastructure-relevant road-width measurements across a

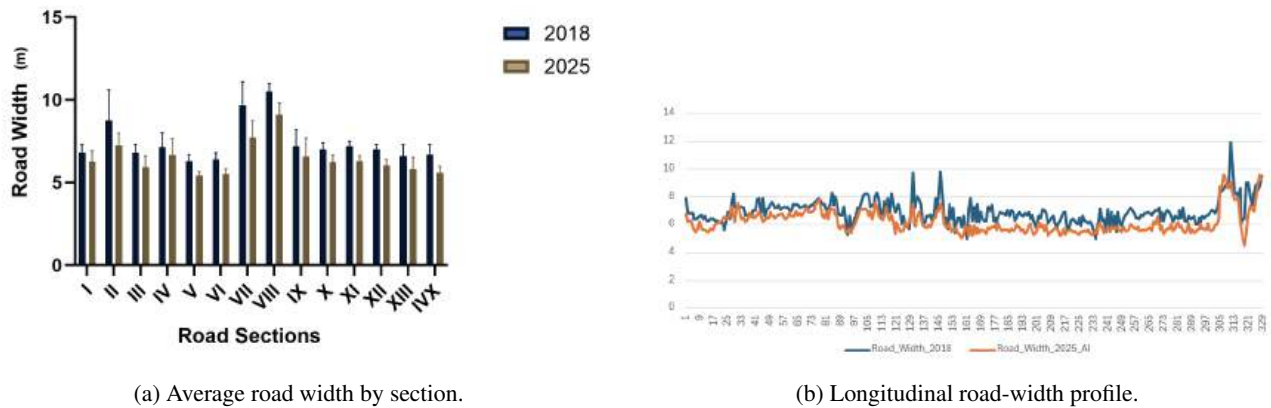


Figure 5: Comparison of 2018 pavement survey widths and 2025 system estimates. (a) Average road widths by representative section, with error bars denoting the standard deviation of chainage-level measurements within each section. (b) Longitudinal width profile along a representative road section, showing the estimated measurements closely following the historical trend while exhibiting smooth and stable behaviour.

diverse regional road network. Although absolute differences relative to historical surveys are observed, these remain systematic and consistent.

4 Conclusion and Future Work

This paper presented a domain-adaptive framework for metric road width estimation using dual rear-view cameras mounted on operational pavement survey vehicles. The proposed framework overcomes key limitations of marking-dependent and forward-facing vision approaches, by combining fine-tuned transformer-based semantic segmentation, calibrated multi-view geometric reconstruction, and trajectory-aware temporal smoothing. Extensive evaluation on real visual survey data demonstrates that the framework produces dense, stable, and metrically consistent width profiles, including on challenging unmarked and rural roads. Our approach requires no additional hardware, integrates seamlessly with existing visual survey workflows, and delivers infrastructure-grade geometric measurements at network scale.

While the proposed framework demonstrates robust performance, it operates under certain practical constraints. First, the 2018 baseline measurements were derived from manually segmented road boundaries, followed by automated metric reconstruction to estimate road width. Consequently, our system’s validation assesses consistency between a semi-automated baseline and a fully automated system, rather than direct manual ground-truth verification. Second, the system assumes the availability of a calibrated dual rear-camera rig, which may not always be available. Third, although trajectory-aware smoothing improves temporal stability, the use of a fixed spatial window (Δs) may not adapt optimally to roads with highly variable geometry or frequent transitions, such as urban intersections or work zones.

Future work will focus on cross-validating vision-based width estimates against high-accuracy manual surveys and mobile LiDAR measurements on selected road sections to establish absolute accuracy bounds. We also plan to incorporate explicit junction and road-transition detection, as well as shadow-aware and illumination-invariant segmentation to improve performance under challenging conditions. Finally, we will investigate adaptive smoothing strategies in which the spatial window (Δs) varies according to road type, curvature, and contextual cues.

References

- [Ali et al., 2020] Ali, A., Hassan, A., Ali, A. R., Khan, H. U., Kazmi, W., and Zaheer, A. (2020). Real-time vehicle distance estimation using single view geometry. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1111–1120.
- [Dahal et al., 2021] Dahal, A., Golab, E., Garlapati, R., Kumar, V. R., and Yogamani, S. (2021). Roadedgenet: Road edge detection system using surround view camera images. *Electronic Imaging*, 33:1–8.
- [Dong et al., 2023] Dong, Y., Patil, S., Van Arem, B., and Farah, H. (2023). A hybrid spatial–temporal deep learning architecture for lane detection. *Computer-Aided Civil and Infrastructure Engineering*, 38(1):67–86.
- [Hecker et al., 2018] Hecker, S., Dai, D., and Van Gool, L. (2018). End-to-end learning of driving models with surround-view cameras and route planners. In *Proceedings of the european conference on computer vision (eccv)*, pages 435–453.
- [Jain et al., 2023] Jain, J., Li, J., Chiu, M. T., Hassani, A., Orlov, N., and Shi, H. (2023). Oneformer: One transformer to rule universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2989–2998.
- [Lu et al., 2025] Lu, Z., Pang, J., and Zhou, Z. (2025). Geometry-aware cross-modal translation with temporal consistency for robust multi-sensor fusion in autonomous driving. *Electronics*, 14(23):4663.
- [Mahlberg et al., 2022] Mahlberg, J. A., Li, H., Cheng, Y.-T., Habib, A., and Bullock, D. M. (2022). Measuring roadway lane widths using connected vehicle sensor data. *Sensors*, 22(19):7187.
- [Paszke et al., 2016] Paszke, A., Chaurasia, A., Kim, S., and Culurciello, E. (2016). Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*.
- [Pu et al., 2022] Pu, M., Huang, Y., Liu, Y., Guan, Q., and Ling, H. (2022). Edter: Edge detection with transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1402–1412.
- [Xie et al., 2021] Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., and Luo, P. (2021). Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090.
- [Xu et al., 2021] Xu, Y., Xu, W., Cheung, D., and Tu, Z. (2021). Line segment detection using transformers without edges. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4257–4266.
- [Yuan et al., 2023] Yuan, T., Cao, W., Zhang, S., Yang, K., Schoen, M., and Duraisamy, B. (2023). Lane detection and estimation from surround view camera sensing systems. In *2023 IEEE SENSORS*, pages 1–4. IEEE.
- [Zhang et al., 2022] Zhang, G., Navasardyan, S., Chen, L., Zhao, Y., Wei, Y., Shi, H., et al. (2022). Mask matching transformer for few-shot segmentation. *Advances in Neural Information Processing Systems*, 35:823–836.
- [Zhang et al., 2021] Zhang, J., Deng, T., Yan, F., and Liu, W. (2021). Lane detection model based on spatio-temporal network with double convolutional gated recurrent units. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):6666–6678.
- [Zou et al., 2019] Zou, Q., Jiang, H., Dai, Q., Yue, Y., Chen, L., and Wang, Q. (2019). Robust lane detection from continuous driving scenes using deep neural networks. *IEEE transactions on vehicular technology*, 69(1):41–54.

SCULPT: Training Edge Vision Models for Post-Training Quantization Readiness

Bharadwaj Kavuri, Sourav Babu-PK, Varadhraj Ellapan, Pullarao Maddu, Prasad Deshpande

Valeo Vision Systems, Tuam, Ireland

Valeo India Pvt. Ltd, Chennai, India

Abstract

Edge vision models are difficult to deploy on resource-constrained hardware, making low-bit post-training quantization (PTQ) attractive. In practice, standard FP32 training often produces heavy-tailed activation distributions whose outliers destabilize activation quantization: preserving the full range wastes quantization bins on rare extremes, while aggressive clipping causes information loss. Existing solutions typically rely on quantization-aware training (QAT), which adds training complexity and bit-width coupling, or advanced PTQ procedures that repair the model after training.

We present SCULPT (Statistical Clipping and Uniform Loss for Post-Training), a training-time method that improves PTQ readiness during ordinary FP32 fine-tuning. SCULPT combines a topology-aware activation regularizer that suppresses quantization-hostile skewness and kurtosis with a stable percentile-based clipping mechanism that learns deployment-ready activation bounds. Unlike QAT, SCULPT does not simulate quantization during optimization; unlike post hoc outlier-repair PTQ methods, it does not require runtime activation transformations. The learned clipping bounds can be exported directly into a standard PTQ workflow for low-bit deployment, including INT8 and lower-bit settings such as W4A8.

Keywords: Post-training quantization, Activation outliers, Edge vision, Model compression

1 Introduction

Edge vision and perception models remain difficult to deploy on resource-constrained hardware due to tight latency, memory, and power budgets. Low-bit quantization is attractive because it enables efficient integer inference while preserving existing model architectures and deployment pipelines. In practice, however, post-training quantization (PTQ) often degrades sharply at low precision when activations exhibit heavy-tailed or highly asymmetric distributions. If the quantizer preserves rare outliers, the effective resolution for the dense bulk of the distribution becomes too coarse; if it clips aggressively, those rare extremes are truncated [Banner et al., 2019].

Figure 1 illustrates this failure mode on a representative hard layer. Most activations remain concentrated in a narrow operating region, but a small number of extreme values drive the maximum far beyond the normal scale of the layer. This mismatch makes min/max calibration brittle and is a major source of PTQ error.

Existing methods typically address this problem either during quantized training or after the full-precision model has already been trained. Quantization-aware training (QAT) and related approaches learn clipping thresholds or quantizer parameters during optimization, but they add training complexity and are often tied to a specific quantization setting [Choi et al., 2018, Jain et al., 2020, Esser et al., 2020]. On the other hand, advanced PTQ methods improve low-bit accuracy through post hoc calibration, adaptive rounding, reconstruction, or activation smoothing after training [Nagel et al., 2020, Li et al., 2021, Xiao et al., 2023]. These methods can be effective, but they also add workflow complexity or transformations that are unattractive in deployment-constrained settings.

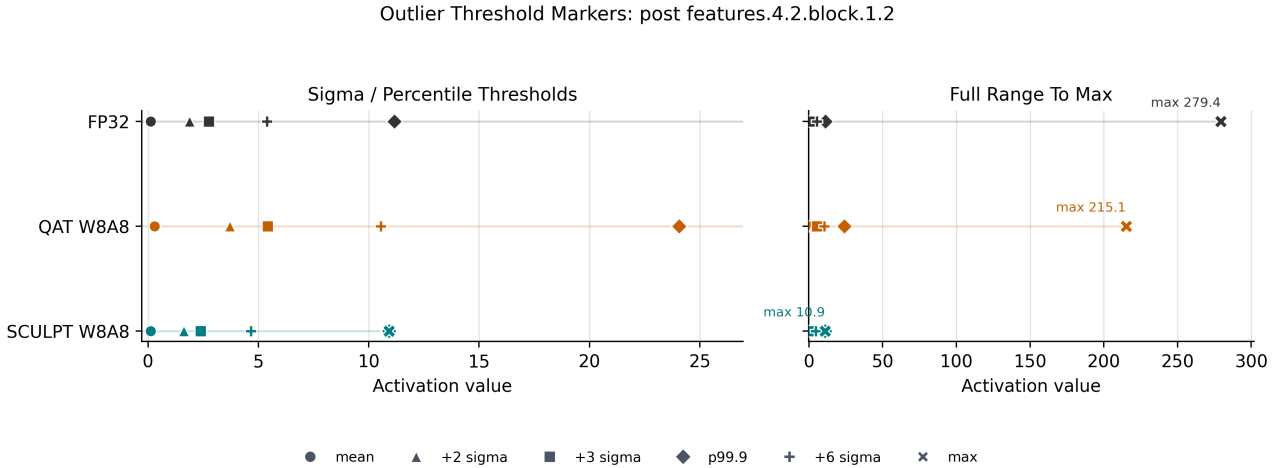


Figure 1: Outlier-threshold marker comparison for a representative hard layer (`post.features.4.2.block.1.2`). Rare extremes drive the calibration maximum far outward in standard FP32 PTQ and, to a lesser extent, in QAT. SCULPT keeps the maximum much closer to the normal layer scale, reducing outlier sensitivity during PTQ calibration.

In this paper, we investigate a different point in the design space: rather than repairing quantization failure only after training, can a model be trained in full precision so that it is already easier to quantize later? Our central hypothesis is that a substantial part of PTQ brittleness originates from activation distributions produced by ordinary FP32 optimization, especially the emergence of structured activation outliers that are poorly matched to low-bit uniform quantizers. To test this idea, we propose **SCULPT** (Statistical Clipping and Uniform Loss for Post-Training), a training-time method for improving PTQ readiness during ordinary FP32 fine-tuning.

SCULPT combines two components. First, *Uniform Activation Distribution Regularization* (UADR) penalizes excess skewness and kurtosis in selected intermediate activations. Second, *StablePercentileClip* introduces stable percentile-based clipping bounds that are adaptive during training and frozen by the end of optimization, yielding activation limits that can be exported directly to a standard PTQ workflow. In contrast to QAT, SCULPT does not simulate quantization during training; in contrast to post hoc outlier-repair methods, it does not require runtime activation transformations.

Our contributions are:

- We identify structured activation outliers from standard FP32 training as a major obstacle to low-bit PTQ.
- We propose SCULPT, combining topology-aware activation regularization with stable percentile clipping to improve PTQ readiness.
- We evaluate SCULPT under a matched quantization contract and analyze accuracy, efficiency, and activation outlier behavior.

2 Related Work

PTQ converts a trained full-precision network into a lower-precision representation, typically using affine or symmetric quantizers [Jacob et al., 2018]. Its accuracy depends heavily on how well the learned activation distributions match the target quantizer. For a uniform quantizer with clipping range $[-\alpha, \alpha]$ or $[0, \alpha]$, increasing α preserves outliers but coarsens the quantization grid, while decreasing α improves resolution for the dense central mass but clips the tails [Banner et al., 2019]. This tradeoff becomes especially severe at low bit-widths.

There is a large body of prior work on improving quantization robustness during training. PACT learns activation clipping parameters [Choi et al., 2018], TQT learns quantization thresholds through backpropagation [Jain et al., 2020], LSQ learns step sizes for low-bit training [Esser et al., 2020], and Robust Quantization studies training models resilient across quantization settings [Shkolnik et al., 2020]. These methods typically optimize under simulated quantization and are therefore naturally viewed as QAT-style approaches.

Another line of work improves PTQ after training through post hoc correction. AdaRound adapts rounding decisions [Nagel et al., 2020], BRECC performs block-wise reconstruction [Li et al., 2021], and SmoothQuant reduces activation outliers by shifting difficulty from activations to weights through an equivalent offline transformation [Xiao et al., 2023]. Recent vision PTQ work has also shown that activation distributions after nonlinearities can be poorly matched to standard uniform quantizers, motivating specialized activation quantization schemes such as PTQ4ViT [Yuan et al., 2022]. SCULPT targets a different operating point: instead of simulating quantization during training or repairing the model after training, it modifies the FP32 fine-tuning trajectory so that the resulting model is more compatible with a later PTQ step.

3 Method

We propose **SCULPT**, a training-time method for improving the post-training quantization (PTQ) readiness of neural networks during ordinary FP32 fine-tuning. SCULPT combines two complementary components: (1) a topology-aware activation regularizer that suppresses quantization-hostile skewness and heavy tails, and (2) a stable percentile-based clipping layer that learns deployment-ready activation bounds. The regularizer provides soft distribution shaping, while the clipping layer provides hard boundary enforcement.

Figure 2 summarizes the full SCULPT pipeline. During fine-tuning, Adaptive Stable Percentile Clipping (Adaptive-SPC) wraps standard activations with percentile-based clamped activations whose bounds are stabilized over time and frozen for deployment. In parallel, UADR tracks activation statistics from selected intermediate tensors and penalizes excess skewness and kurtosis using topology-aware one-sided targets.

3.1 Problem Setup

Let f_θ denote a model trained in full precision on task loss $\mathcal{L}_{\text{task}}$. Our goal is not to quantize the model during training, but to modify the FP32 training trajectory so that the final model is more robust to later PTQ at low precision. We focus on intermediate activations, which are often a dominant source of PTQ error when their dynamic ranges are governed by rare but large outliers. SCULPT seeks to reduce this tradeoff at its source by shaping activation statistics during FP32 fine-tuning.

3.2 StablePercentileClip

StablePercentileClip learns stable activation bounds during training and exports them directly for later PTQ. Unlike fixed clipping functions such as ReLU6, the clipping range is estimated from observed activation statistics and refined throughout fine-tuning. Given an input tensor $X^{(t)}$ at step t , we first apply the wrapped activation function:

$$A^{(t)} = \phi(X^{(t)}), \quad (1)$$

where $\phi(\cdot)$ denotes the original activation (e.g., SiLU, ReLU, ReLU6, or Hardswish). Adaptive-SPC estimates percentile bounds on the output of the wrapped activation, not on the pre-activation tensor. StablePercentileClip therefore wraps the original activation rather than replacing it, preserving the underlying nonlinearity while adding learned post-activation bounds.

Let $V^{(t)} \in \mathbb{R}^N$ denote the flattened version of $A^{(t)}$. To control sorting cost, we form a strided subsample $\tilde{V}^{(t)}$ using

$$s = \max\left(1, \left\lfloor \frac{N}{N_{\max}} \right\rfloor\right), \quad (2)$$

and

$$\tilde{V}^{(t)} = \left\{ V_{i:s}^{(t)} \right\}_{i=0}^{\lfloor N/s \rfloor - 1}. \quad (3)$$

We then compute detached lower and upper percentile estimates:

$$Q_{\text{lower}}^{(t)} = \text{Quantile}(\tilde{V}^{(t)}, p_{\text{lower}}), \quad Q_{\text{upper}}^{(t)} = \text{Quantile}(\tilde{V}^{(t)}, p_{\text{upper}}), \quad (4)$$

$$\hat{Q}_{\text{lower}}^{(t)} = \text{sg}(Q_{\text{lower}}^{(t)}), \quad \hat{Q}_{\text{upper}}^{(t)} = \text{sg}(Q_{\text{upper}}^{(t)}). \quad (5)$$

We maintain running lower and upper bounds updated with inverse-time momentum:

$$\alpha_t = \frac{\alpha_0}{1 + \gamma t}, \quad (6)$$

$$M_{\text{lower}}^{(t)} = (1 - \alpha_t) M_{\text{lower}}^{(t-1)} + \alpha_t \hat{Q}_{\text{lower}}^{(t)}, \quad (7)$$

$$M_{\text{upper}}^{(t)} = (1 - \alpha_t) M_{\text{upper}}^{(t-1)} + \alpha_t \hat{Q}_{\text{upper}}^{(t)}. \quad (8)$$

The clipped activation is then

$$Y^{(t)} = \text{clip}(A^{(t)}, M_{\text{lower}}^{(t)}, M_{\text{upper}}^{(t)}). \quad (9)$$

Because the running bounds are updated through detached statistics, gradients do not propagate through the quantile computation itself. During backpropagation, the layer behaves like a hard clamp applied after the wrapped activation.

3.3 UADR

Stable clipping constrains activation range but does not explicitly discourage skewed or heavy-tailed activation shapes. To complement it, we apply **UADR** to selected learnable-layer outputs. Here, “topology-aware” refers to the functional position of an activation within a computation block, such as pre-activation linear outputs versus post-activation tensors. Although we retain the acronym UADR, the regularizer does not force activations toward a strictly uniform distribution; instead, it suppresses excess skewness and heavy tails using topology-aware one-sided targets.

Let $X^{(l)} \in \mathbb{R}^N$ denote the flattened output tensor of learnable layer l . We compute the empirical mean and standard deviation:

$$\mu_l = \frac{1}{N} \sum_{i=1}^N X_i^{(l)}, \quad \sigma_l = \sqrt{\frac{1}{N} \sum_{i=1}^N (X_i^{(l)} - \mu_l)^2 + \epsilon}, \quad (10)$$

and standardized activations

$$Z_i^{(l)} = \frac{X_i^{(l)} - \mu_l}{\sigma_l}. \quad (11)$$

Skewness and kurtosis are then

$$\gamma(X^{(l)}) = \frac{1}{N} \sum_{i=1}^N (Z_i^{(l)})^3, \quad \kappa(X^{(l)}) = \frac{1}{N} \sum_{i=1}^N (Z_i^{(l)})^4. \quad (12)$$

In the experiments of this paper, we use the simpler setting

$$\gamma_T^{(l)} = 0 \quad (13)$$

for all regularized layers, and a shared upper target $\kappa_T = 3$.

The per-layer UADR loss is

$$\mathcal{L}_{\text{UADR}}^{(l)} = \alpha \max(0, \gamma(X^{(l)}) - \gamma_T^{(l)})^2 + \beta \max(0, \kappa(X^{(l)}) - \kappa_T)^2. \quad (14)$$

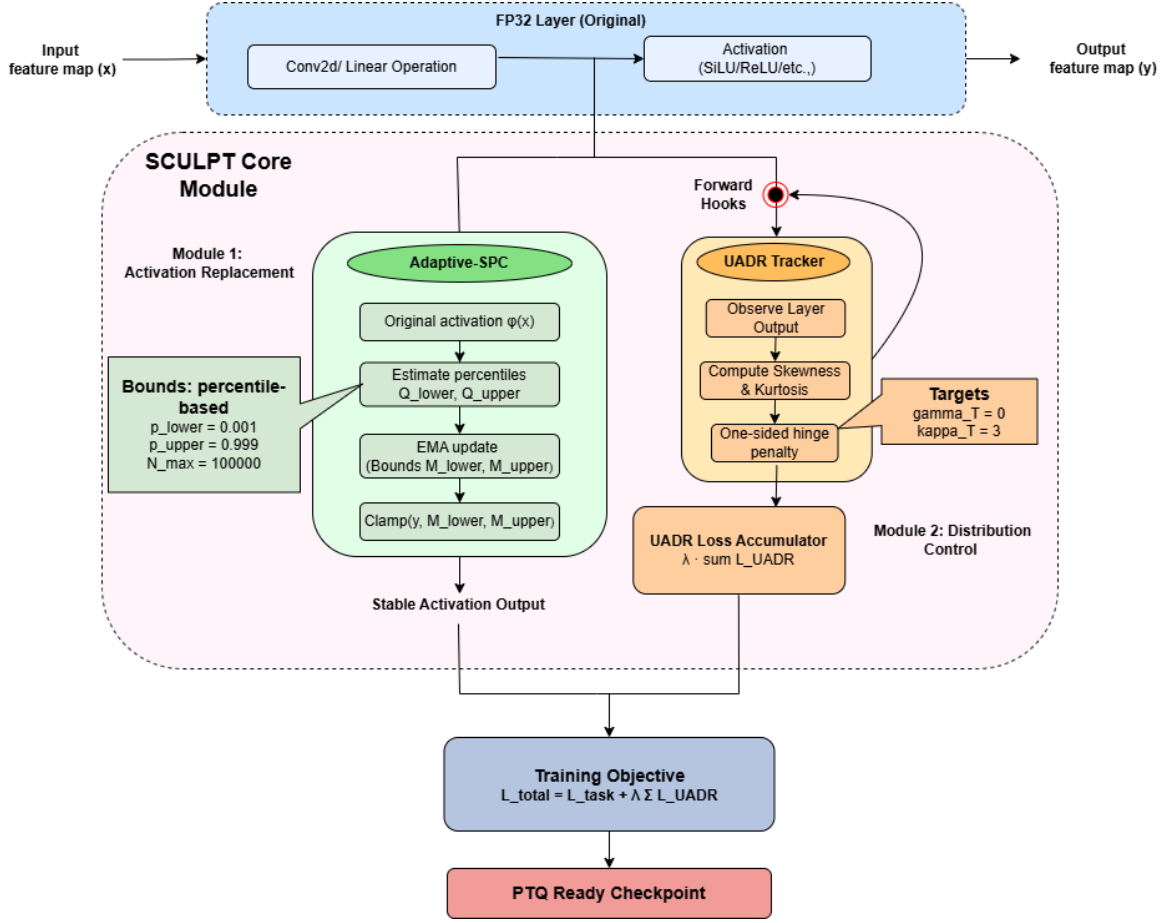


Figure 2: Overview of SCULPT. Adaptive-SPC learns stable activation bounds for PTQ export, while UADR penalizes excess skewness and kurtosis.

Let $\mathcal{S}$ denote the set of layers to which UADR is applied. In our implementation, UADR is applied to Conv2d and Linear outputs, while Adaptive-SPC is inserted after SiLU, ReLU, ReLU6, and Hardswish activations. The total training objective is

$$\mathcal{L}_{total} = \mathcal{L}_{task} + \lambda \sum_{l \in \mathcal{S}} \mathcal{L}_{UADR}^{(l)}. \quad (15)$$

UADR and StablePercentileClip play complementary roles: UADR reduces the tendency of intermediate activations to develop excessive skewness and heavy tails, while StablePercentileClip enforces explicit activation bounds that stabilize into deployment-time clipping limits. After fine-tuning, the learned bounds are frozen and exported as layerwise activation limits for deployment, so the resulting model can be quantized using a standard PTQ workflow without simulated quantization during training and without runtime outlier-repair transforms.

4 Experimental Setup

We evaluate SCULPT on EfficientNet-B0 [Tan and Le, 2019] with ImageNette [Howard, 2019, Deng et al., 2009] as a controlled low-bit quantization testbed. The baseline model is initialized from Torchvision ImageNet-1K EfficientNet-B0 weights, adapted to 10 classes, and trained in FP32 for 80 epochs using AdamW, label smoothing of 0.1, RandAugment, automatic mixed precision (AMP), batch size 128, and a cosine OneCycle learning-rate schedule.

Starting from the trained FP32 checkpoint, SCULPT fine-tunes the model with Adaptive Stable Percentile Clipping inserted after all SiLU, ReLU, ReLU6, and Hardswish activations, and with UADR regularization applied to all Conv2d and Linear outputs. Adaptive-SPC estimates lower and upper activation percentile bounds at 0.001 and 0.999, respectively, using an inverse-time exponential moving average. We set $N_{\max} = 100000$ for percentile estimation in all experiments. These bounds are frozen after epoch 2 and converted into static activation-plus-clamp modules for deployment-time PTQ.

For UADR, we use a one-sided hinge penalty with target skewness $\gamma_T = 0$, target kurtosis $\kappa_T = 3$, $\alpha = 0.01$, $\beta = 0.01$, and global regularization weight $\lambda = 5 \times 10^{-4}$, which is reduced by a factor of 0.1 after SPC freezing.

We compare standard FP32 PTQ, SCULPT PTQ, and QAT under an identical supergroup QDQ contract. The PTQ setup uses per-channel symmetric weight quantization, per-tensor asymmetric activation quantization, and min/max calibration over 32 calibration batches. QDQ nodes are placed before Conv/Linear modules and after activation or SPC clamp modules, with no QDQ inserted between Conv and SiLU/ReLU/SPC. The QAT baseline uses native PyTorch fake quantizers under the same supergroup boundaries.

Checkpoint selection is based on quantized validation accuracy under the matched supergroup PTQ contract rather than FP32 validation accuracy. For FP32, SCULPT, and QAT runs, epoch-level PTQ tracking is used to identify the best deployment-relevant checkpoint.

5 Experimental Results

5.1 Main Quantization Results

Table 1 compares standard FP32 PTQ, SCULPT-conditioned PTQ, and QAT baselines under the same supergroup QDQ contract. SCULPT substantially improves low-bit PTQ relative to naive FP32 PTQ. In the primary W8A8 setting, standard FP32 PTQ achieves 90.5478% top-1 accuracy, whereas SCULPT PTQ reaches 99.3631%, an absolute gain of 8.8153 percentage points. Under the same W8A8 setting, the tested QAT baseline reaches 98.2420%, which remains below SCULPT PTQ by 1.1211 points.

The same trend holds at W4A8. Standard FP32 PTQ drops sharply to 40.5605%, whereas SCULPT PTQ reaches 80.2803%, slightly exceeding the tested QAT W8A8 PTQ baseline at 79.4395%. These results suggest that conditioning activation statistics during FP32 fine-tuning can substantially improve downstream PTQ robustness, even without simulated quantization during training.

Method	FP32	W8A8 PTQ	W4A8 PTQ	Best PTQ Step
FP32	99.2866	90.5478	40.5605	3577
SCULPT	99.5159	99.3631	80.2803	584
QAT W8A8	97.7325	98.2420	79.4395	2044
QAT W4A8	93.1465	89.9618	98.2675	2044

Table 1: Main results under the shared QDQ contract.

SCULPT reaches its best W8A8 PTQ result after 584 optimizer steps, compared with 2044 for the tested W8A8 QAT baseline and 3577 for naive FP32 PTQ. Its full run is also shorter: 16 epochs / 1168 steps, versus 30 epochs / 2190 steps for both QAT baselines and 80 epochs / 5840 steps for the original FP32 run.

Although dedicated low-bit QAT remains stronger in its target setting, SCULPT is designed to improve PTQ readiness without simulated quantization during training, and is therefore most directly compared against naive PTQ and the matched W8A8 QAT baseline in the primary deployment setting.

5.2 Activation Distribution Analysis

To understand why SCULPT improves PTQ accuracy, we analyze the activation distributions observed after training. Across the 131 encoded activation boundaries used in the quantized model, our layer-wise analysis shows that SCULPT reduces the mean PTQ NRMSE/std from 0.04818 for standard FP32 PTQ to 0.01519,

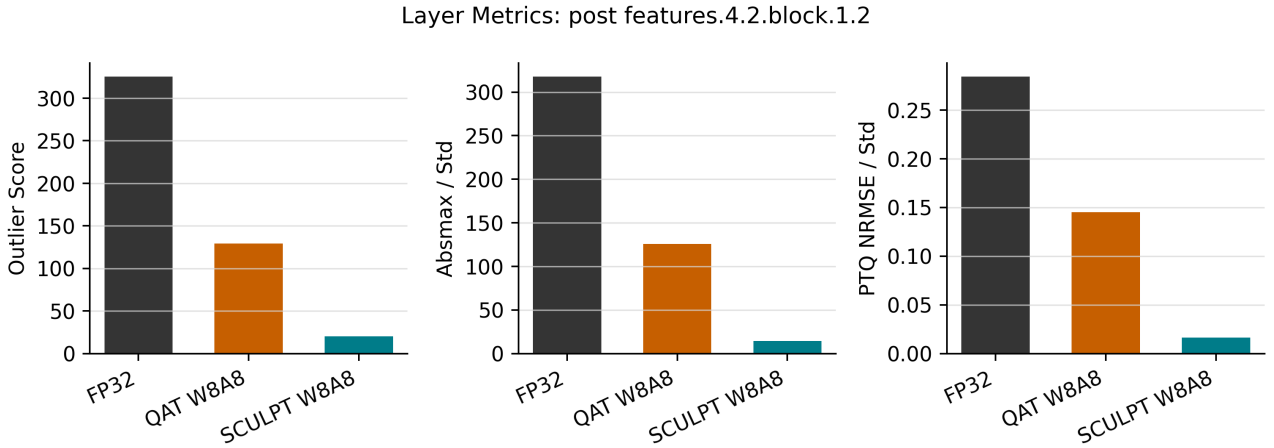


Figure 3: Scalar metrics for a representative hard layer. Lower is better in all three cases. SCULPT sharply reduces outlier score, absmax/std, and PTQ NRMSE/std relative to both standard FP32 PTQ and the tested W8A8 QAT baseline.

and reduces the mean absmax/std from 42.04 to 11.56. These reductions are consistent with the intended role of SCULPT: suppressing the rare extreme values that otherwise dominate min/max calibration and stretch the quantization range.

Figure 3 summarizes a representative hard layer (`post.features.4.2.block.1.2`) using scalar metrics. SCULPT reduces the outlier score from 325.08 in FP32 PTQ and 129.12 in QAT to 19.99, reduces absmax/std from 317.6 and 125.8 to 14.4, and reduces normalized PTQ error from 0.2841 and 0.1452 to 0.0163, respectively. These results indicate that SCULPT does not merely shift calibration slightly; it materially changes the tail behavior that makes PTQ brittle in the first place.

5.3 Discussion

These results suggest that a substantial portion of PTQ brittleness can be mitigated before quantization itself, by modifying the activation statistics learned during FP32 fine-tuning. In this controlled benchmark, SCULPT yields a substantially stronger PTQ checkpoint than naive FP32 PTQ and also compares favorably to the tested W8A8 QAT baseline while requiring fewer optimization steps to reach its best quantized result.

6 Conclusion

We presented **SCULPT**, a training-time method for improving post-training quantization (PTQ) readiness during ordinary FP32 fine-tuning. SCULPT combines a topology-aware activation regularizer that suppresses quantization-hostile skewness and heavy tails with a stable percentile-based clipping mechanism that learns deployment-ready activation bounds. Together, these components condition the full-precision model so that later low-bit PTQ becomes substantially more robust without simulated quantization during training or runtime outlier-repair transforms.

On EfficientNet-B0/ImageNet under an identical supergroup QDQ contract, SCULPT substantially outperforms naive FP32 PTQ and compares favorably to the tested W8A8 QAT baseline. In the primary W8A8 setting, SCULPT PTQ reaches 99.3631% top-1 accuracy, compared with 90.5478% for standard FP32 PTQ and 98.2420% for the tested W8A8 QAT baseline. These results support the hypothesis that a substantial portion of PTQ brittleness can be mitigated before quantization itself. Extending the same analysis to more complex perception models remains an important next step.

Acknowledgments

Generative AI assisted with language refinement and structural editing; the authors verified all technical claims and remain responsible for the manuscript. The authors gratefully acknowledge Valeo Vision Systems, Tuam, Ireland, for providing support and resources for this work.

References

- [Banner et al., 2019] Banner, R., Nahshan, Y., Hoffer, E., and Soudry, D. (2019). ACIQ: Analytical clipping for integer quantization of neural networks. In *International Conference on Learning Representations (ICLR)*.
- [Choi et al., 2018] Choi, J., Wang, Z., Venkataramani, S., Chuang, P. I.-J., Srinivasan, V., and Gopalakrishnan, K. (2018). PACT: Parameterized clipping activation for quantized neural networks. *arXiv preprint arXiv:1805.06085*.
- [Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE.
- [Esser et al., 2020] Esser, S. K., McKinstry, J. L., Bablani, D., Appuswamy, R., and Modha, D. S. (2020). Learned Step Size Quantization. In *International Conference on Learning Representations (ICLR)*.
- [Howard, 2019] Howard, J. (2019). Imagenette: A smaller subset of 10 easily classified classes from imagenet. <https://github.com/fastai/imagenette>.
- [Jacob et al., 2018] Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H., and Kalenichenko, D. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Jain et al., 2020] Jain, S., Gural, A., Wu, M., and Dick, C. (2020). Trained quantization thresholds for accurate and efficient fixed-point inference of deep neural networks. In *Proceedings of Machine Learning and Systems (MLSys)*.
- [Li et al., 2021] Li, Y., Gong, R., Tan, X., Yang, Y., Hu, P., Zhang, Q., Yu, F., Wang, W., and Gu, S. (2021). BRECC: Pushing the limit of post-training quantization by block reconstruction. In *International Conference on Learning Representations (ICLR)*.
- [Nagel et al., 2020] Nagel, M., Amjad, R. A., Van Baalen, M., Louizos, C., and Blankevoort, T. (2020). Up or down? adaptive rounding for post-training quantization. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*.
- [Shkolnik et al., 2020] Shkolnik, M., Chmiel, B., Banner, R., Shomron, G., Nahshan, Y., Bronstein, A., and Weiser, U. (2020). Robust quantization: One model to rule them all. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- [Tan and Le, 2019] Tan, M. and Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR.
- [Xiao et al., 2023] Xiao, G., Lin, J., Seznec, M., Wu, H., Demouth, J., and Han, S. (2023). Smoothquant: Accurate and efficient post-training quantization for large language models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*.
- [Yuan et al., 2022] Yuan, Z., Xue, C., Chen, Y., Wu, Q., and Sun, G. (2022). PTQ4ViT: Post-training quantization for vision transformers with twin uniform quantization. In *European Conference on Computer Vision (ECCV)*.

Pix2Repair: Implicit Shape Restoration from Images

Nikolas Lamb¹, Xinchao Song², Natasha Kholgade Banerjee³, Sean Banerjee³

¹*Clarkson University, Potsdam, NY, USA*, ²*Rochester Institute of Technology, Rochester, NY, USA*,

³*Wright State University, Dayton, OH, USA*

nikolas.lamb@gmail.com, xxsics@rit.edu,
{natasha.banerjee, sean.banerjee} @wright.edu

Abstract

We present Pix2Repair, an automated shape repair approach that generates restoration shapes from images to repair fractured objects. Prior repair approaches require a high-resolution watertight 3D mesh of the fractured object as input. Such input 3D meshes must be obtained using expensive 3D scanners and require manual cleanup, limiting accessibility and scalability. Pix2Repair takes an image of the fractured object as input and automatically generates a 3D printable restoration. We contribute a novel shape function that deconstructs a latent code representing the fractured object into a complete shape and a break surface. We collect a dataset of fractured everyday objects and show restorations for real fractures from our dataset and for synthetic fractures from the Geometric Breaks and Breaking Bad datasets. Our approach outperforms shape completion approaches adapted for shape repair in terms of chamfer distance, normal consistency, and percent restorations generated.

1 Introduction

We all know the frustration of trying to repair a treasured glass, vase, or mug that has been accidentally fractured, e.g., Figure 1(a). Sometimes fractured objects can be repaired via reassembly. However, in many cases parts are destroyed or lost during the fracturing process leaving gaps in the repaired object. Automated shape repair approaches seek to fill this gap by automatically generating restoration shapes for fractured objects. In addition to preserving heirlooms, object repair algorithms have the potential to reduce household waste, restore cultural heritage objects, and automate industrial recycling. A recent body of work has emerged to automate the repair of fractured objects by predicting 3D printable restoration shapes using machine learning [Hermoza and Sipiran, 2018, Lamb et al., 2022c, Lamb et al., 2022b, Lamb et al., 2022a, Jaramillo and Sipiran, 2024]. However, existing approaches operate in low-resolution voxel space [Hermoza and Sipiran, 2018], or require high resolution 3D scans of the fractured object [Lamb et al., 2022c, Lamb et al., 2022b, Lamb et al., 2022a, Jaramillo and Sipiran, 2024] which must be obtained using expensive scanners, e.g., the \$2,000 (USD) Einscan-SP, making them inaccessible to consumers.

We present Pix2Repair, an approach to automatically generate repair parts for fractured objects from images. Our approach builds on prior approaches that perform shape completion from images using implicit representations [Murez et al., 2020, Lin et al., 2020, Mescheder et al., 2019, Wang et al., 2022, Xu et al., 2020, Chen and Zhang, 2019, Li and Zhang, 2021, Hui et al., 2022, Cheng et al., 2023]. However, existing shape completion approaches generate a complete shape from a partial observation, e.g., an image that observes the complete shape from a fixed perspective. Unlike traditional shape completion, our approach generates a restoration shape from an image of a fractured object, such as the image shown in Figure 1(a). Generating restoration shapes is a harder task than generating complete shapes as restorations are often much smaller than complete shapes, as is the case for the restoration shape in Figure 1(f). Restorations also contain high-frequency geometry at the point of fracture that is difficult to predict directly.

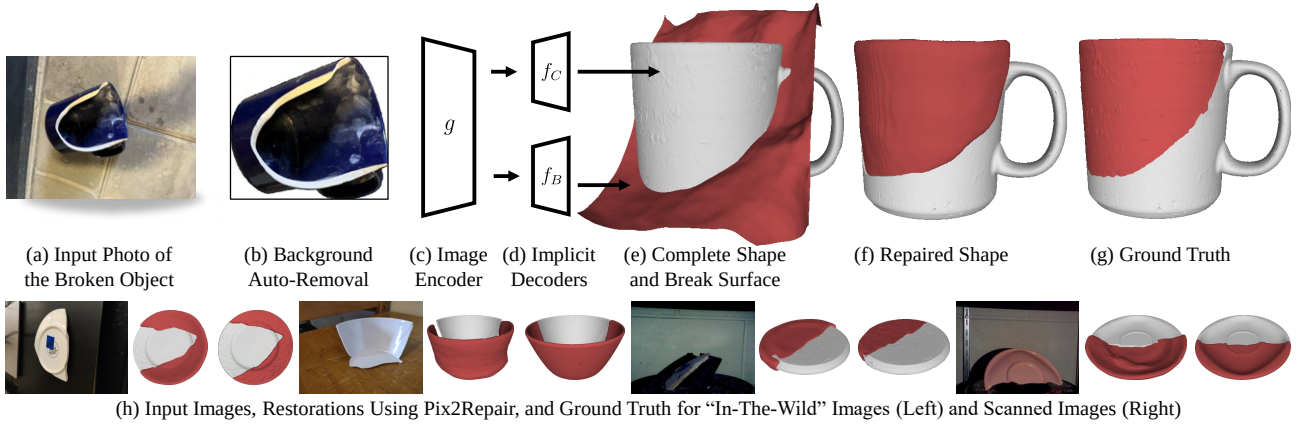


Figure 1: We present Pix2Repair, an automated shape repair approach. Pix2Repair takes an image of a fractured object as input, automatically removes the background, and predicts a complete shape and break surface, which combine to generate a restoration shape.

To overcome the challenge of representing shapes with small size and high-frequency surface detail, we compute the repair shape as the Boolean intersection of a predicted complete shape and break shape, where the break shape is given by a break surface, shown in Figures 1(e). Our approach draws inspiration from existing shape repair approaches [Lamb et al., 2022b, Lamb et al., 2022a], which similarly represent the restoration shape as a Boolean intersection. However, our approach learns to predict a complete shape and break surface from a single shape code representing the fractured object, in contrast to prior approaches that obtain two separate complete and break codes by performing iterative optimization during inference. Learning a mapping from the fractured shape code to the complete shape and break surface allows our network to predict restoration shapes directly from color images of the fractured shape. Accepting images as input makes Pix2Repair more accessible to consumers and faster during inference than prior approaches.

2 Related Work

Approaches to perform repair of damaged objects use voxels [Hermoza and Sipiran, 2018], point clouds [Jaramillo and Sipiran, 2024], or leverage implicit representations [Lamb et al., 2022b, Lamb et al., 2022c, Lamb et al., 2022a]. Approaches based on implicit representations learn a mapping from fractured shapes to restoration shapes [Lamb et al., 2022c], or learn occupancy [Lamb et al., 2022b], or combined signed distance, occupancy, and normal field functions [Lamb et al., 2022a] for a complete shape and break surface that combine to generate the restoration shape. Unlike these approaches that use 3D shape as input, our work uses images, enabling repair from a single image. Work exists in generative prediction of brittle fractures [Huang and Kanai, 2025] and in understanding images of cultural heritage objects using vision language models [Zhang et al., 2026]. Approaches have also investigated re-assembly [Chen et al., 2022, Tsesmelis et al., 2024], where the goal is assembling existing pieces rather than inferring new missing ones. A related field of study is shape completion, which focuses on generating a complete shape from single and multi-view images [Xie et al., 2020, Wu et al., 2023, Yi et al., 2021, Liu et al., 2023, Chen et al., 2026, Wang et al., 2021, Cheng et al., 2023]. A large body of recent work has focused on learning to represent shapes implicitly using functions [Murez et al., 2020, Lin et al., 2020, Chen and Zhang, 2019, Wang et al., 2022, Li and Zhang, 2021, Xu et al., 2020, Hui et al., 2022, Mescheder et al., 2019, Wang et al., 2021], including methods to learn multi-scale implicit fields [Wang et al., 2022], disentangle object geometry and texture [Li and Zhang, 2021], impose symmetry constraints [Xu et al., 2020], or deforming shape templates [Hui et al., 2022]. Unlike shape completion approaches, our approach learns to predict a restoration shape as the Boolean intersection of two

other shape functions, i.e., the complete shape and break surface.

3 Pix2Repair

We use the approach from DeepMend [Lamb et al., 2022b] to represent a restoration shape R as arithmetic relaxations to logical operations between learnable occupancy function representations o_C and o_B of a complete shape C and a break surface B , as $o_R(\mathbf{x}) = o_C(\mathbf{x})o_B(\mathbf{x})$, where $\mathbf{x} \in \mathbb{R}^3$ represents a 3D point. We learn two functions, f_C and f_B , to represent occupancy in the complete shape and occupancy in the break shape respectively. During training, the function f_C is tasked with learning a one-to-many mapping from the fractured shape observation to the complete shape. The function f_B is tasked with learning a one-to-one mapping from the fractured shape observation to the break shape. Each function takes point $\mathbf{x} \in \mathbb{R}^3$ as input and is conditioned on a latent code corresponding to an observation of the fractured shape $\mathbf{z}$, such that $o_C(\mathbf{x}) = f_C(\mathbf{z}, \mathbf{x})$ and $o_B(\mathbf{x}) = f_B(\mathbf{z}, \mathbf{x})$. During training and testing we obtain $\mathbf{z}$ by passing an input image through an image encoder g to obtain a fractured shape latent code. We compute occupancy in the restoration shape by substituting the predicted values into its occupancy representation, i.e., as $o_R(\mathbf{x}) = f_B(\mathbf{z}, \mathbf{x})f_C(\mathbf{z}, \mathbf{x})$.

We train our network on complete, break, and restoration shape tuples (C, B, R) . We use $\hat{o}_C(\mathbf{x})$, $\hat{o}_B(\mathbf{x})$, and $\hat{o}_R(\mathbf{x})$ to denote the ground truth occupancy values for shapes C , B , and R respectively at a point $\mathbf{x}$. During training, for each shape tuple we optimize with respect to the loss

$$\mathcal{L} = \sum_{\mathbf{x} \in \mathcal{X}} (BCE(o_C(\mathbf{x}), \hat{o}_C(\mathbf{x})) + \lambda_B BCE(o_B(\mathbf{x}), \hat{o}_B(\mathbf{x})) + \lambda_R BCE(o_R(\mathbf{x}), \hat{o}_R(\mathbf{x}))). \quad (1)$$

In Equation (1), BCE is the binary cross-entropy loss, and $\mathcal{X}$ is a set of probing points. The individual loss terms encourage the predicted values for the complete, break, and restoration shapes to converge to the ground truth values during training. λ_B and λ_R are coefficients that allow us to balance the relative importance of the complete, break, and restoration shape loss. During training we optimize with respect to the parameters of g , f_C , and f_B . We use $\lambda_B = 1.0$ and $\lambda_R = 1.0$. To perform optimization we use the Adam optimizer with a learning rate of $lr = 2e^{-5}$. For all experiments we train our approach for 10 epochs. Unless otherwise specified we use ResNet34 as our image encoder g . We concatenate ResNet34 with a fully connected layer with 512 output nodes to obtain a 512 dimensional fractured shape code $\mathbf{z}$. We represent the occupancy functions f_C and f_B using two neural networks with 6 ResNet blocks and batch normalization. Each ResNet block has 256 nodes with leaky ReLU activation functions. The latent code and points are input to f_C and f_B separately using two fully connected layers with 256 output nodes and added together after the first layer. The final layer has a single output node with a sigmoid activation function.

To obtain sample points for training, for each training shape tuple we pre-compute n sample points uniformly in a cube with side lengths of 1.1 units surrounding the shapes. We also pre-compute n sample points on the surface of each shape in the tuple, and offset the surface points randomly according to a Gaussian distribution with $\sigma = 0.01$. We combine the uniform and surface sampled points into a single set $\mathcal{P}$. We use $n = 50,000$. On each training iteration we randomly select a subset $\mathcal{X} \subset \mathcal{P}$ of m sample points from the pre-computed uniform and surface sampled points. We select the points $\mathcal{X}$ such that, for each shape $S \in \{C, B, R\}$, at least $\frac{m}{6}$ points lie inside of S and at least $\frac{m}{6}$ points lie outside of S . During training, we use $m = 2,048$ sample points. After training our network, to obtain the restoration mesh we query our network with a grid of k^3 points and extract the 0.5 level isosurface using marching cubes. For evaluation we use $k = 128$. For the visualizations shown in the paper we use $k = 256$.

We input images of fractured objects with backgrounds removed to our network. We separately train a background auto-remover to automatically detect the fractured object and remove the background using the images and ground truth masks from our dataset. We use Segment Anything Model (SAM) as the base model and fine-tune its mask decoder to generate favorable masks based on the approach proposed by MedSAM [Ma et al., 2024]. A bounding box of the same size as the input image is used as the prompt for the decoder. To further remove the noise, only the largest connected component in the generated mask is returned as the final mask. We use Dice Similarity Coefficient (DSC) to evaluate the removal performance. Our auto-remover achieves an average DSC score of 0.9608, while the original SAM achieves a score of 0.0001.

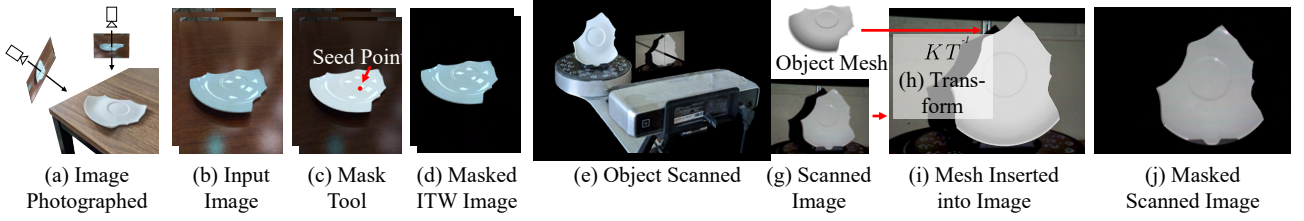


Figure 2: Data collection. For ITW images, we (a-b) photograph objects, and (c) use SAM to (d) obtain masked images. For scanned images, we use the model and .xml files obtained from Einscan scanner 3D scan process in *Fantastic Breaks* (scanning process shown in (e)), to (f-g) obtain a mesh and image, and use the transform in (h) to (i) insert the mesh into the image and (j) obtain the masked image.

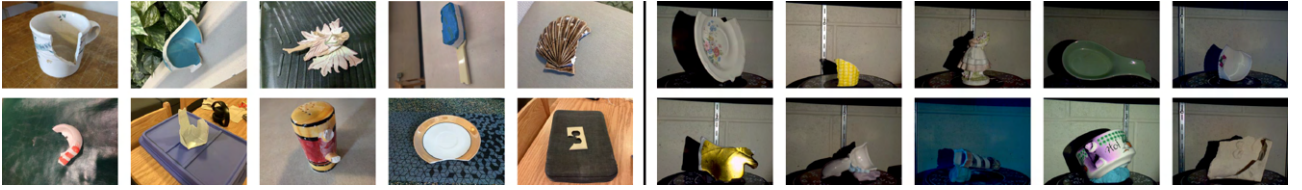


Figure 3: Example ITW photographs (left set) and scanned images (right set) from our dataset.

4 Experiments

We train Pix2Repair on different combinations of two synthetic datasets, as well as a dataset of real images that we collected, and evaluate it on our in-house dataset. Since no image-based shape repair approaches exist, for comparison, we adapt two existing shape completion approaches, Pix2Vox++ [Xie et al., 2020] and Occupancy Networks (ONet) [Mescheder et al., 2019], to take images as input to perform shape repairs.

In-house Dataset of Real Images. We create an in-house dataset of 11,637 real-world images of damaged objects from *Fantastic Breaks* [Lamb et al., 2023], allowing shape repair approaches to use 3D models from *Fantastic Breaks* for training and evaluation. 9,528 images are taken “in-the-wild”, i.e., photographed in a variety of residential, office, and classroom environments, and 2,109 scanned images from the imaging process used to generate 3D scans for *Fantastic Break*. As shown in the left set of Figure 3, the 9,528 in-the-wild images are obtained using consumer grade phones and cameras. We recruited 14 experimenters to obtain photographs. Each object was photographed by at least two experimenters, from multiple viewpoints in at least three locations, as shown in Figure 2(a). Photographs were taken in diverse locations and against a variety of backgrounds within university premises. Two experimenters manually selected a seed point as input to SAM, and selected a segmentation level (out of three possible levels) that best segmented the object, as shown in Figure 2(c). Figure 2(d) shows a masked image example. We obtain the 2,109 scan images by parsing scan files of the objects provided by *Fantastic Breaks* through scanning the object using the Einscan-SP 3D scanner. The scan process is shown in Figure 2(e). We collected images from the scan files obtained by the authors of *Fantastic Breaks*. During scanning, the scanner software produces a single .xml project file that contains pointers to the color images and depth scans, as well as the scanner intrinsic matrix. Color images are saved as .bmp files. Depth scans are saved as binary .rge files, which also store the camera frame, distance from the camera to the scanner turntable, depth scan points, colors, and normals, and the transformation matrix T that aligns the depth scan to the final high-resolution 3D model. We implement a parser for the .rge files that enables extraction of the depth scan points, colors, and normals, and the object pose in the image T . We used our parser to extract colored point clouds with normals that correspond to the depth scans obtained by the scanner. We show example scanned images at the bottom of Figure 3. We obtained the mask of the object in each .bmp image using the transformation matrix T stored in the .rge file. We invert T to obtain a transform that inserts the 3D model in the .bmp image, i.e., T^{-1} , apply T^{-1} to the 3D model, and project the transformed model onto the image using the camera intrinsic matrix K to obtain the image mask, as shown in Figure 2(i).

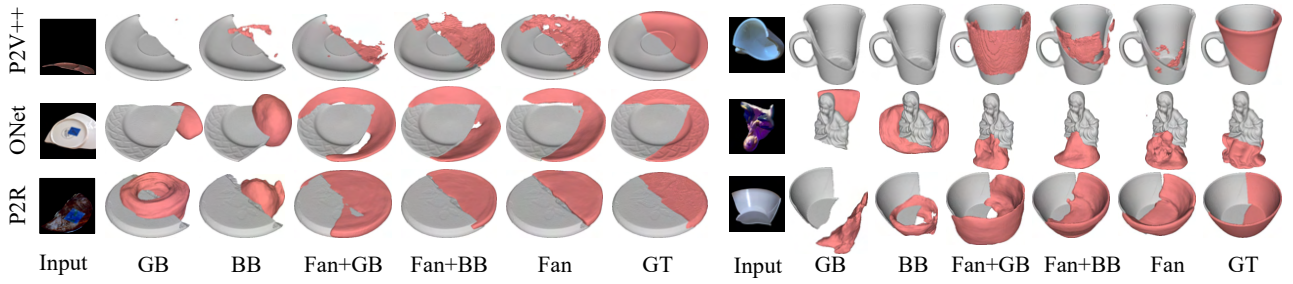


Figure 4: Repairs for images from our dataset for Pix2Vox++ (P2V++), ONet, and Pix2Repair (P2R) over various train sets and ground truth (GT). Fractured object in grey, repair in red.

Synthetic Datasets. We use two publicly available fractured object datasets that contain synthetically fractured objects: Geometric Breaks (GB) [Lamb et al., 2022a] and Breaking Bad (BB) [Sellán et al., 2022]. GB contains 25,250 objects, and each object is fractured between 1 and 10 times. Objects are fractured by subtracting a randomized geometric primitive from each object. We use objects from the jars, bottles, mugs, and GSO subsets of the dataset, totaling 5,252 objects. BB contains 10,474 publicly available 3D models. The dataset is divided into everyday objects, artifact objects, and other objects based on application. Fractures are synthesized by computing a set of the most likely fracture modes. Each object is fractured 80 times. We use 405 objects from the everyday subset, totaling 32,400 objects. All objects from the GB and BB datasets are normalized to sit inside a unit cube. Though all objects from GB are oriented upright, some objects from BB are not. We manually find and remove objects from BB that are oriented perpendicular to canonical orientations from ShapeNet. To enable computing the occupancy, we also discard BB objects that are not watertight. After removal, we have 31,661 BB objects. We generate renders that look directly at the fractured region. We follow prior work that trains on synthetic renders [Hui et al., 2022, Chen and Zhang, 2019, Huang et al., 2022, Mescheder et al., 2019]. To create renders, we compute the volumetric center of all vertices on the fractured region. We place a camera looking at the center of the fractured region vertices. We fit a plane to the fractured region vertices, and offset the camera in the direction of the plane normal. The normal may be pointing in the incorrect direction, i.e., pointing into the fractured object. To prevent incorrect normals, we compute the average dot product between the normal and the set of vectors pointing from the center of the fractured region vertices to the vertices in the fractured object, and flip the normal if the dot product is positive. We offset the camera in the direction of the normal such that it observes the entire object. We light the object using ambient lighting and using a camera-centered spotlight. We render objects at 224×224 pixel resolution using Pyrender. We split each dataset into 70%/30% train/test sets within classes. As GB contains a high number of diverse objects (5,252) and a low number of fractures per object (1-10), we randomly select five renders from each fractured object in the dataset, giving us 18,170 training and 8,090 testing renders. As BB contains a low number of diverse objects (405) and a high number of fractures per object (80), we randomly select one render from each fractured object, giving us 22,219 training and 10,181 testing renders. We use the same split as *Fantastic Breaks* for our dataset, giving us 8,598 training and 3,039 testing images. We do not allow train/test object sets to overlap.

Adapted Shape Repair Approaches. We adapt ONet and Pix2Vox++ to perform shape repair using images as our baselines. ONet represents shapes implicitly using the occupancy function. Pix2Vox++ represents shapes as voxels. To adapt shape completion approaches for shape repair, we feed images of fractured objects as input and train each approach to predict the restoration shape. For Pix2Vox++ we implement their 128^3 voxel resolution network, and train the network on 128^3 resolution voxelized restoration shapes. For ONet, we use our point sampling approach to obtain 100,000 uniform and surface sampled points.

Train Set	Pix2Vox++			ONet			Pix2Repair		
	NZ % $\uparrow$	CD $\downarrow$	NC $\uparrow$	NZ % $\uparrow$	CD $\downarrow$	NC $\uparrow$	NZ % $\uparrow$	CD $\downarrow$	NC $\uparrow$
GB	8.2%	0.230	0.518	83.9%	0.165	0.520	95.5%	0.125	0.497
BB	84.9%	0.147	0.530	87.0%	0.095	0.545	99.7%	0.068	0.577
BB + GB	36.49%	0.193	0.497	87.4%	0.121	0.528	99.2%	0.092	0.548
Ours + GB	83.1%	0.103	0.532	99.4%	0.062	0.635	99.9%	0.062	0.599
Ours + BB	86.3%	0.108	0.533	99.5%	0.051	0.641	100.0%	0.051	0.646
Ours + BB + GB	76.0%	0.134	0.538	99.7%	0.056	0.642	99.9%	0.055	0.637
Ours	90.4%	0.087	0.532	99.9%	0.049	0.638	100.0%	0.055	0.617

Table 1: Non-zero percentage (NZ%), chamfer distance (CD), and normal consistency (NC) for repair approaches. GB=Geometric Breaks, BB=Breaking Bad, Ours=Our in-house dataset. Best values are in bold.

5 Results

We show the quantitative results on our dataset in Table 1 and example restorations for objects from our dataset generated by Pix2Repair in Figure 1(h) and 4. The results demonstrate that Pix2Repair has the best performance in all but one case. They also illustrate that training repair approaches on the images of real fractured objects or synthetic datasets augmented with such real images is necessary to generate accurate restorations of real objects. To evaluate predicted restoration shapes, we use the chamfer distance (CD). CD is an indicator of overall shape similarity, but may fail to capture out-of-distribution predictions [Wu et al., 2021]. A challenge of predicting canonically oriented restoration shapes is that the pose of the restoration shape cannot be determined for objects with rotational symmetry. As the location of the fractured region is independent of the object’s canonical orientation, the network cannot accurately estimate where the restoration shape should be placed relative to the canonically oriented fractured shape. To evaluate symmetric shapes, we rotate the restoration shape around the up-axis P times, and take the rotation that minimizes the CD between the predicted restoration shape and the ground truth shape to be the pose of the predicted restoration shape. We use $P = 36$ for all classes in the dataset. We also evaluate shapes using the normal consistency (NC) [Mescheder et al., 2019]. NC captures the alignment of surface normals between the predicted and ground truth shapes. We also show the non-zero percentage (NZ%), that denotes the percentage of fractured shapes for which a restoration shape is generated, i.e., for which the occupancy is not predicted as all zero. We compute CD and NC over 30,000 points.

We show the results of training on the images of synthetically fractured objects from Geometric Breaks (GB) and Breaking Bad (BB) and evaluating on our dataset in the first 3 rows of Table 1. We observe that all approaches struggle to generate accurate repair parts for objects when trained on synthetic data, as shown by the relatively high CD, e.g., of 0.125, 0.068, and 0.092, and 0.165, 0.095, and 0.121 for Pix2Repair and ONet when trained on GB, BB, and BB + GB respectively, compared to the bottom 4 rows of Table 1. Qualitative results in Figure 4 also show visually poor repairs for GB and BB. Pix2Vox++ struggles to repair a large percentage of objects, e.g., only 8.4% when trained on GB, as small objects may be reduced to as few as 100 points during voxelization. We also observe networks trained on BB outperform networks trained on GB in terms of CD and NC, as shown in rows 1 and 2 of Table 1, likely as the physics-based fracture method of Breaking Bad is more realistic than the geometric method of Geometric Breaks.

The results of training on datasets that contain both synthetic and real fractured object images are shown in rows 4-7 of Table 1. For comparison, we also show results training only on objects from our dataset. We observe that adding real images to any combination of synthetic datasets increases performance, e.g., a CD of 0.125 to 0.062 and of 0.068 to 0.051 using Pix2Repair for GB and BB. We also observe that using our dataset without any synthetic data performs comparably to training using synthetic data with real data, e.g., with a CD of 0.051 with Ours+BB and 0.055 for ours using Pix2Repair, and with a CD of 0.051 with Ours+BB and 0.049 with ours using ONet, indicating that for some objects our image dataset contains sufficient variation to generalize to new 3D shapes. Figure 4 shows that most approaches trained on real data produce visually appealing repair parts, e.g., the plate for ONet and the coaster and bowl for Pix2Repair. Similar to training on synthetic datasets only, Pix2Repair outperforms shape completion approaches adapted for completion, except

for ONet on Fan. We also observe that combining Breaking Bad with our dataset produces some of the best repairs in terms of CD and NC, as shown in row 5 of Table 1 and column 5 of Figure 4, e.g., with a CD of 0.051 for ONet and NC of 0.646 for Pix2Repair.

6 Conclusion

We present Pix2Repair, an approach to automatically generate restoration shapes from images. Our approach outperforms existing shape completion approaches adapted for shape repair. In contrast to existing approaches that require high-resolution 3D models, by operating on images of fractured objects Pix2Repair represents a significant step forward in increasing the accessibility of repair algorithms. A limitation is that a single image of the fractured object may not observe parts of the object that are essential to determining its overall structure. Future work may investigate the use of vision-language models to incorporate interactive interfaces that guide the user when capturing the initial image or image set of the fractured object. Another limitation is that restoration shapes predicted by our approach are not accurate to scale, preventing them from being used directly for 3D printing. Accurately determining object scale from color images without external information is challenging. One approach to retain scale is to use depth cameras. Future work may explore the use depth cameras found on smartphones or augmented reality glasses.

References

- [Chen et al., 2026] Chen, X., Chu, F.-J., Gleize, P., Liang, K. J., Sax, A., Tang, H., Wang, W., Guo, M., Hardin, T., Li, X., et al. (2026). Sam 3d: 3dfy anything in images. In *Proc. CVPR*. IEEE.
- [Chen et al., 2022] Chen, Y.-C., Li, H., Turpin, D., Jacobson, A., and Garg, A. (2022). Neural shape mating: Self-supervised object assembly with adversarial shape priors. In *Proc. CVPR*. IEEE.
- [Chen and Zhang, 2019] Chen, Z. and Zhang, H. (2019). Learning implicit fields for generative shape modeling. In *Proc. CVPR*. IEEE.
- [Cheng et al., 2023] Cheng, Y.-C., Lee, H.-Y., Tulyakov, S., Schwing, A. G., and Gui, L.-Y. (2023). Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In *Proc. CVPR*. IEEE.
- [Hermoza and Sipiran, 2018] Hermoza, R. and Sipiran, I. (2018). 3d reconstruction of incomplete archaeological objects using a generative adversarial network. In *Proc. CGI*. ACM.
- [Huang and Kanai, 2025] Huang, Y. and Kanai, T. (2025). Deepfracture: A generative approach for predicting brittle fractures with neural discrete representation learning. In *Computer Graphics Forum*.
- [Huang et al., 2022] Huang, Z., Stojanov, S., Thai, A., Jampani, V., and Rehg, J. M. (2022). Planes vs. chairs: Category-guided 3d shape learning without any 3d cues. In *Proc. ECCV*.
- [Hui et al., 2022] Hui, K.-H., Li, R., Hu, J., and Fu, C.-W. (2022). Neural template: Topology-aware reconstruction and disentangled generation of 3d meshes. In *Proc. CVPR*. IEEE.
- [Jaramillo and Sipiran, 2024] Jaramillo, P. and Sipiran, I. (2024). Cultural heritage 3d reconstruction with diffusion networks. In *ECCV*.
- [Lamb et al., 2022a] Lamb, N., Banerjee, S., and Banerjee, N. K. (2022a). Deepjoin: Learning a joint occupancy, signed distance, and normal field function for shape repair. *ACM Trans. on Graphics*.
- [Lamb et al., 2022b] Lamb, N., Banerjee, S., and Banerjee, N. K. (2022b). Deepmend: Learning occupancy functions to represent shape for repair. In *Proc. ECCV*.

- [Lamb et al., 2022c] Lamb, N., Banerjee, S., and Banerjee, N. K. (2022c). Mendnet: Restoration of fractured shapes using learned occupancy functions. *Proc. Computer Graphics Forum*.
- [Lamb et al., 2023] Lamb, N., Palmer, C., Molloy, B., Banerjee, S., and Banerjee, N. K. (2023). Fantastic breaks: A dataset of paired 3d scans of real-world broken objects and their complete counterparts. In *Proc. CVPR*. IEEE.
- [Li and Zhang, 2021] Li, M. and Zhang, H. (2021). D2im-net: Learning detail disentangled implicit fields from single images. In *Proc. CVPR*. IEEE.
- [Lin et al., 2020] Lin, C.-H., Wang, C., and Lucey, S. (2020). Sdf-srn: Learning signed distance 3d object reconstruction from static images. *Proc. NeurIPS*.
- [Liu et al., 2023] Liu, M., Xu, C., Jin, H., Chen, L., Varma T, M., Xu, Z., and Su, H. (2023). One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Proc. NeurIPS*.
- [Ma et al., 2024] Ma, J., He, Y., Li, F., Han, L., You, C., and Wang, B. (2024). Segment anything in medical images. *Nature communications*.
- [Mescheder et al., 2019] Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., and Geiger, A. (2019). Occupancy networks: Learning 3d reconstruction in function space. In *Proc. CVPR*. IEEE.
- [Murez et al., 2020] Murez, Z., Van As, T., Bartolozzi, J., Sinha, A., Badrinarayanan, V., and Rabinovich, A. (2020). Atlas: End-to-end 3d scene reconstruction from posed images. In *Proc. ECCV*.
- [Sellán et al., 2022] Sellán, S., Chen, Y.-C., Wu, Z., Garg, A., and Jacobson, A. (2022). Breaking Bad: A dataset for geometric fracture and reassembly. In *Proc. NeurIPS*.
- [Tsesmelis et al., 2024] Tsesmelis, T., Palmieri, L., Khoroshiltseva, M., Islam, A., Elkin, G., Shahar, O. I., Scarpellini, G., Fiorini, S., Ohayon, Y., Alali, N., et al. (2024). Re-assembling the past: The repair dataset and benchmark for real world 2d and 3d puzzle solving. In *Proc. NeurIPS*.
- [Wang et al., 2021] Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., and Wang, W. (2021). Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In *Proc. NeurIPS*.
- [Wang et al., 2022] Wang, Y., Zhuang, Y., Liu, Y., and Chen, B. (2022). Mdisn: Learning multiscale deformed implicit fields from single images. *Visual Informatics*.
- [Wu et al., 2021] Wu, T., Pan, L., Zhang, J., Wang, T., Liu, Z., and Lin, D. (2021). Density-aware chamfer distance as a comprehensive metric for point cloud completion. *Proc. NeurIPS*.
- [Wu et al., 2023] Wu, T., Wang, J., Pan, X., Xu, X., Theobalt, C., Liu, Z., and Lin, D. (2023). Voxurf: Voxel-based efficient and accurate neural surface reconstruction. In *Proc. ICLR*.
- [Xie et al., 2020] Xie, H., Yao, H., Zhang, S., Zhou, S., and Sun, W. (2020). Pix2vox++: Multi-scale context-aware 3d object reconstruction from single and multiple images. *International Journal of Computer Vision*.
- [Xu et al., 2020] Xu, Y., Fan, T., Yuan, Y., and Singh, G. (2020). Ladybird: Quasi-monte carlo sampling for deep implicit field based 3d reconstruction with symmetry. In *Proc. ECCV*.
- [Yi et al., 2021] Yi, L., Gong, B., and Funkhouser, T. (2021). Complete & label: A domain adaptation approach to semantic segmentation of lidar point clouds. In *Proc. CVPR*. IEEE.
- [Zhang et al., 2026] Zhang, N., Zhang, Z., Wang, J., Zhao, Y., and Tang, H. (2026). Vasevqa-3d: Benchmarking 3d vlms on ancient greek pottery. In *Proc. ICLR*.

ARAS-FCWNet: Low-Cost and Lightweight Model for Forward Collision Warning (FCW) Systems in Unstructured ARAS Environments

Satyajit Nayak, Lakshman-Prabhu Balasubiramanian, Rakshita Joshi, Ganesh Sistu

anSWer, Valeo India

Abstract

Advanced Riding Autonomous Systems (ARAS) are critical for enhancing safety in modern vehicles, including motorcycles and bicycles. A Forward Collision Warning (FCW) system is a crucial component of ARAS that aims to prevent accidents by alerting the rider to imminent collisions. However, existing FCW systems often rely on multiple high-resolution sensors and sophisticated hardware, which are cost-prohibitive and computationally intensive for widespread adoption, especially in low-cost vehicles. Developing an effective and affordable FCW system for ARAS poses several technical challenges, including sensor limitations, computational constraints, real-time performance, environmental variability, and lightweight algorithms. We propose a low-cost, lightweight perception model optimized for a 1.3 MP RGB camera supplemented by IMU and vehicle odometry. The framework accurately detects potential collisions in real time with minimal false-positive and false-negative rates. We optimize the system for resource-constrained embedded platforms by balancing processing speed, memory footprint, and power consumption. Extensive validation across diverse environments demonstrates the robustness and reliability of the proposed FCW solution for low-cost two-wheeler safety.

Keywords: Advanced Rider Assistance System (ARAS), Forward Collision Warning (FCW), Unstructured-ARAS, RGB Pinhole-Camera, Low-cost Lightweight Model.

1 Introduction

Advanced Rider Assistance Systems (ARAS) are gaining significant attention for improving safety in motorcycles, scooters, bicycles, and other lightweight mobility [Kaye et al., 2024]. Compared with conventional four-wheeler Advanced Driver Assistance Systems (ADAS), ARAS operates under more challenging conditions due to rider instability, narrow vehicle geometry, irregular traffic behavior, and highly unstructured road environment platforms [Naweed and Blackman, 2024]. Among different ARAS functionalities, Forward Collision Warning (FCW) accurately forecasting the trajectories of road users remains highly challenging, because of the inherent uncertainty in human behavior, environmental variability, occlusions, and heterogeneous interactions among multiple dynamic agents, as explained in Fig 1. incorporating radar, LiDAR, stereo cameras, surround-view systems, and high-resolution perception sensors [?]. Although these approaches achieve reliable performance, they significantly increase

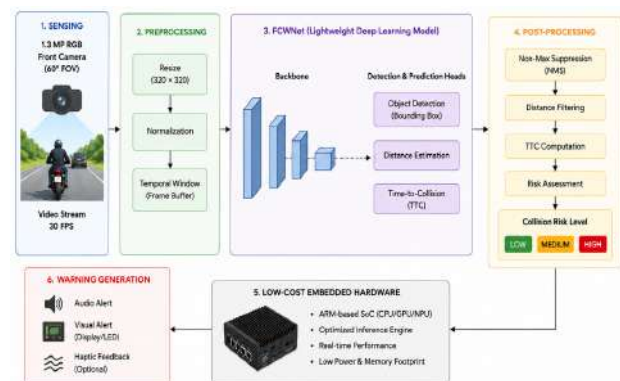


Figure 1: Conceptual Diagram of ARAS-FCWNet

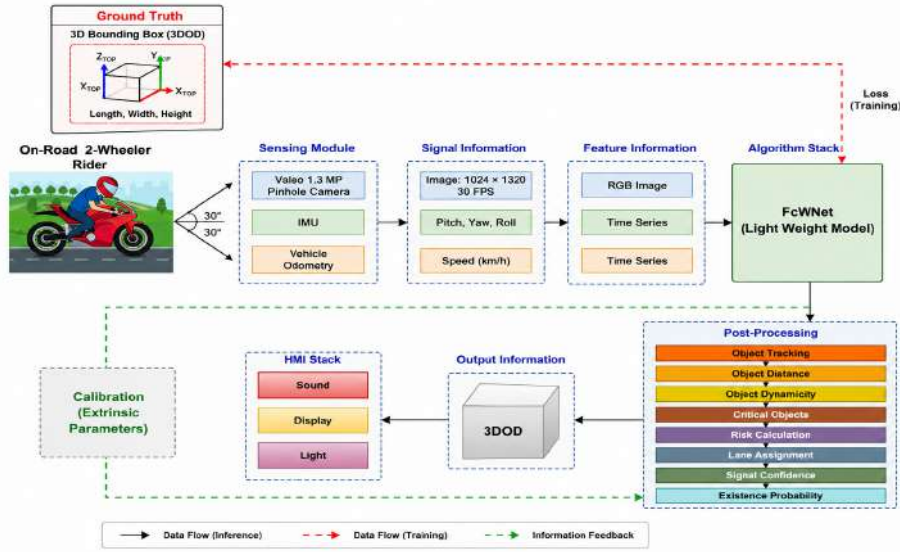


Figure 2: Proposed ARAS-FCWNet Framework.

hardware cost, computational complexity, memory consumption, and power requirements. Such solutions are often unsuitable for low-cost ARAS deployment, particularly in emerging markets where affordable two-wheeler safety systems are essential.

Recent advances in deep learning and computer vision have enabled the development of vision-based FCW systems using object detection and scene understanding frameworks such as Faster R-CNN, SSD, YOLO, and monocular 3D object detection approaches [Manakitsa et al., 2024], [Zhang et al., 2025]. Several lightweight perception models including MobileNet [Howard et al., 2017], EfficientNet [Tan and Le, 2019], and Tiny-YOLO [Redmon and Farhadi, 2018] have also been proposed for embedded deployment. However, most existing methods are designed for structured road environments and depend heavily on lane markings, high-resolution sensors, or powerful GPU-based [Mihalj et al., 2022]. Their performance often degrades in unstructured ARAS scenarios involving mixed traffic, vulnerable road users, occlusions, poor illumination, adverse weather, and irregular road boundaries.

Recent studies further explored graph-based and temporal motion-aware frameworks for intelligent transportation systems, incorporating object tracking, trajectory estimation, and collision risk [Wu et al., 2023]. Nevertheless, many existing FCW approaches focus primarily on object detection while neglecting integrated free-space segmentation, depth estimation, motion understanding, and risk-aware reasoning within a unified lightweight architecture [Zhang et al., 2025]. In addition, deploying such systems on low-power embedded hardware while maintaining real-time performance remains a significant challenge.

To address these limitations, this paper proposes *ARAS-FCWNet*, a low-cost, lightweight forward collision warning framework designed specifically for unstructured ARAS environments (some scenarios are shown in Fig. 3). As illustrated in Fig. 2, the proposed system utilizes a single 1.3 MP front-view RGB pinhole camera integrated with IMU and vehicle odometry signals to perform efficient scene perception and collision risk estimation. The framework adopts a lightweight multi-task deep learning architecture capable of jointly performing object detection, free-space estimation, monocular depth prediction, motion estimation, and collision risk assessment in real time.

Unlike conventional FCW systems that depend on expensive multi-sensor setups [Hassan et al., 2026], ARAS-FCWNet operates entirely on monocular vision and low-cost embedded hardware while maintaining efficient inference capability. The framework further integrates object tracking, lane assignment, object dynamicity estimation, critical object identification, and Time-To-Collision (TTC)-based risk calculation through a lightweight post-processing pipeline. Experimental validation under diverse traffic and environmental conditions demonstrates the effectiveness of the proposed approach for real-time ARAS deployment.

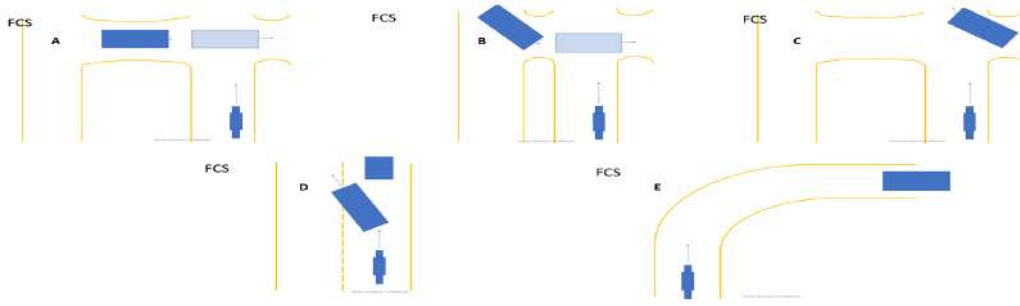


Figure 3: ARAS Scenarios: (A) A stationary car pulls into the motorcycle’s path at a divided highway/stop sign crossing. (B) A stationary car enters the motorcycle’s path while crossing a divided highway. (C) A parked car suddenly moves into the motorcycle’s lane. (D) A following car changes lanes abruptly to avoid a suddenly stopped vehicle, while the motorcycle never observes the stopped vehicle moving. (E) A motorcycle rounds a curve as a car ahead slows to a stop in the same lane.

The primary contributions of this work are summarized as follows:

- A lightweight monocular FCW framework for unstructured ARAS environments using a single low-cost RGB camera.
- A unified multi-task deep learning architecture integrating object detection, free-space segmentation, depth estimation, and motion prediction.
- A computationally efficient embedded perception pipeline optimized for real-time ARAS deployment on low-power hardware platforms.
- A robust collision risk estimation framework incorporating object tracking, TTC analysis, and rider-centric warning generation.

2 Proposed Methodology

The proposed *ARAS-FCWNet* framework is designed as a lightweight monocular Forward Collision Warning (FCW) system for unstructured Advanced Rider Assistance System (ARAS) environments. As illustrated in Fig. 2, the framework utilizes a single low-cost 1.3 MP RGB front-view camera along with IMU and vehicle odometry signals to perform efficient collision risk estimation in real time. The overall architecture consists of five major stages: (i) image acquisition and preprocessing, (ii) lightweight multi-task perception, (iii) object tracking and motion estimation, (iv) collision risk analysis, and (v) rider warning generation.

2.1 Input Acquisition and Preprocessing

Let the monocular RGB image captured at time instant t be represented as:

$$I_t \in \mathbb{R}^{H \times W \times 3} \quad (1)$$

where H and W denote the image height and width, respectively. The captured frame is resized and normalized before being forwarded to the proposed FCWNet backbone. Simultaneously, inertial and odometry measurements are collected:

$$S_t = \{v_t, a_t, \omega_t\} \quad (2)$$

where v_t denotes vehicle velocity, a_t denotes acceleration, and ω_t represents yaw rate obtained from IMU sensors.

2.2 Lightweight Multi-Task Perception Network

ARAS-FCWNet uses a lightweight MobileNetV3-Small backbone (2.5M parameters, 0.42 GFLOPs) operating at a unified input resolution of 320×320 pixels. The shared backbone extracts hierarchical features feeding into task-specific decoders for 2D bounding box detection, free-space segmentation, monocular depth prediction (via pinhole camera projection constraints $d = f \cdot Y / y$), and temporal motion analysis. This unified multi-task design achieves a memory footprint under 18 MB on embedded devices.

2.3 Distance Estimation and Dynamic Object Analysis

Object distance is estimated using monocular geometric constraints based on object scale variation in image space. Relative velocity is computed from temporal distance changes between consecutive frames. Lightweight temporal association is further employed for object tracking and motion-state estimation.

The object dynamicity score is defined as:

$$\Gamma_i = \sqrt{(v_{x,i})^2 + (v_{y,i})^2} \quad (3)$$

where $v_{x,i}$ and $v_{y,i}$ denote horizontal and vertical motion components, respectively.

2.4 Critical Object Selection and Collision Risk Estimation

Collision priority is determined using object distance, motion dynamicity, and detection confidence:

$$C_i = \alpha \frac{1}{d_i} + \beta \Gamma_i + \gamma c_i, \quad (4)$$

where the weighting coefficients are empirically tuned as $\alpha = 0.5$, $\beta = 0.3$, and $\gamma = 0.2$.

Collision risk is estimated using Time-To-Collision (TTC), and the final warning score is computed as:

$$R_i = \exp(-\lambda \cdot TTC_i), \quad (5)$$

where the sensitivity parameter is set to $\lambda = 1.2$.

An FCW alert is triggered whenever the estimated collision risk exceeds a predefined threshold.

2.5 Joint Optimization

The proposed framework is trained end-to-end using a unified multi-task objective that jointly optimizes detection, segmentation, depth estimation, and collision-risk prediction:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{det} + \lambda_2 \mathcal{L}_{seg} + \lambda_3 \mathcal{L}_{depth} + \lambda_4 \mathcal{L}_{risk}, \quad (6)$$

where the loss weights are selected through validation tuning as $\lambda_1 = 1.0$, $\lambda_2 = 0.5$, $\lambda_3 = 0.5$, and $\lambda_4 = 0.2$.

The lightweight design enables efficient real-time deployment on resource-constrained ARAS hardware while maintaining reliable collision warning performance in unstructured traffic environments.

3 Experimental Results

The proposed ARAS-FCWNet framework is implemented using PyTorch and evaluated on a workstation equipped with a 13th Gen Intel Core i7-13850HX processor, 64 GB RAM, and a 4 GB GPU. The model is trained using the Adam optimizer with an initial learning rate of 1×10^{-4} and a batch size of 64. All experiments converge within approximately 45 epochs. To ensure suitability for embedded ARAS deployment, the proposed network is optimized for low memory consumption and real-time inference.



Figure 4: ARAS-FCWNet Multi-Task Perception Network: (i) Object Detection, (ii) Free-space Segmentation, (iii) Monocular-Depth Estimation, and (iv) TTC-based Risk Estimation.

3.1 Dataset and Experimental Setup

ARAS-FCWNet is trained and tested across BDD100K [Yu et al., 2020] (10,000 frames), nuScenes [Caesar et al., 2020] (5,000 keyframes), and a proprietary Valeo ARAS dataset (12,000 frames captured from rider-perspective 1.3 MP cameras across unstructured Indian traffic). Four-wheeler datasets (BDD100K/nuScenes) provide rich front-view traffic dynamics, while the Valeo dataset supplies true two-wheeler lean dynamics and irregular road layouts. Experiments are averaged across 5 independent runs (70:10:20 split).

3.2 Evaluation Metrics

The performance of ARAS-FCWNet is evaluated using standard FCW and perception metrics, including Precision, Recall, F1-score, Mean Average Precision (mAP), False Positive Rate (FPR), False Negative Rate (FNR), and inference latency. In addition, Time-To-Collision (TTC) estimation accuracy and collision warning accuracy are also analyzed.

The Mean Average Precision (mAP) is used to evaluate object detection performance, while the Time-To-Collision (TTC) ground truth is computed using:

$$TTC_{gt} = \frac{d_{rel}}{v_{rel}}, \quad (7)$$

where d_{rel} and v_{rel} denote the relative distance and relative velocity between the ego vehicle and the target object, respectively. The TTC ground truth was generated on the Valeo test split using calibrated 3D bounding boxes obtained from reference LiDAR/radar annotations.

3.3 Quantitative Performance Comparison

For a fair comparison, all baseline models were retrained from scratch using the same unified dataset split and identical training protocol, with an IoU threshold of 0.50 for object detection evaluation. The proposed ARAS-

FCWNet is compared against lightweight real-time perception models, including Tiny-YOLO, MobileNet-SSD, and YOLOv5-Nano. Table 1 reports the combined test performance across the BDD100K, KITTI, and Valeo datasets, demonstrating the effectiveness of the proposed framework under diverse driving scenarios.

Table 1: Performance comparison of FCW models.

Method	mAP↑	F1-score↑	FPR↓	FPS↑
Tiny-YOLO	71.2	74.5	12.6	42
MobileNet-SSD	75.8	78.4	10.2	38
YOLOv5-Nano	81.5	83.1	8.7	34
ARAS-FCWNet	85.7	87.4	6.1	48

The results demonstrate that the proposed ARAS-FCWNet achieves superior detection accuracy while maintaining high real-time performance suitable for low-cost embedded ARAS deployment.

3.4 Collision Warning Performance

To evaluate collision warning capability, TTC-based warning accuracy is analyzed under different riding conditions (debug screen status result of a sample scenario shown in Fig 5). Table 2 presents the TTC estimation and FCW accuracy results.

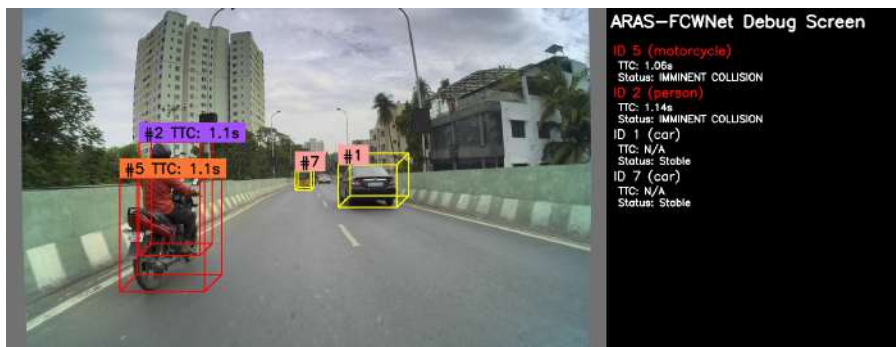


Figure 5: ARAS-FCWNet Debug Screen Results of a Sample Input-Image using TTC Performance-based Estimation of Collision Risk Status (stable in this Scenario)

Table 2: Collision warning evaluation results.

Method	TTC Error↓	Warning Accuracy↑	Latency (ms)↓
Tiny-YOLO	1.42	78.5	41
MobileNet-SSD	1.18	81.2	39
YOLOv5-Nano	0.96	85.7	35
ARAS-FCWNet	0.71	91.3	28

The proposed framework achieves the lowest TTC estimation error and highest warning accuracy, indicating reliable collision risk prediction in unstructured riding environments.

Fig. 4 illustrates qualitative FCW examples generated by ARAS-FCWNet under challenging traffic conditions. The framework successfully detects vehicles, pedestrians, and vulnerable road users while accurately identifying free-space regions and collision threats. The proposed lightweight architecture maintains stable performance under illumination changes, partial occlusions, and dense traffic scenarios.

3.5 Ablation Analysis

An ablation study is conducted to evaluate the contribution of individual modules within ARAS-FCWNet. Table 3 summarizes the effect of removing specific components.

Table 3: Ablation study of ARAS-FCWNet components.

Configuration	mAP $\uparrow$	Warning Accuracy $\uparrow$
Without Depth Estimation	80.6	84.5
Without Motion Estimation	81.2	85.1
Without Free-space Segmentation	79.8	83.7
Full ARAS-FCWNet	85.7	91.3



Figure 6: ARAS-FCWNet Debug Screen Status: Imminent Collision using TTC Performance

The results confirm that integrating monocular depth estimation, motion reasoning, and free-space segmentation significantly improves the robustness of collision prediction and the reliability of FCW.

3.6 Discussion

To evaluate embedded viability, ARAS-FCWNet was deployed on an NVIDIA Jetson Orin Nano (8GB). Operating under an 8W power budget, the TensorRT-optimized engine achieved 42 FPS with a latency of 23.8 ms, consuming 1.2 GB RAM. This confirms real-time embedded performance on low-cost ARM/GPU architectures. Unlike conventional FCW systems relying on expensive multi-sensor fusion, the proposed framework achieves reliable collision warning performance using only a low-cost monocular RGB camera and a lightweight deep learning architecture. The proposed system therefore provides a scalable and affordable solution for next-generation rider assistance systems operating in challenging unstructured traffic environments.

4 Conclusion and Future Scope

This paper presented *ARAS-FCWNet*, a lightweight multi-task perception framework for real-time Forward Collision Warning (FCW) in unstructured ARAS environments. The proposed model integrates multi-task perception with collision-risk assessment using single RGB camera input tightly coupled with vehicle IMU and odometry telemetry. By replacing heavy multi-sensor stacks with fused monocular-inertial cues, ARAS-FCWNet provides an affordable safety framework for two-wheelers in unstructured traffic.

Experiments on BDD100K, nuScenes, and in-house Valeo ARAS datasets demonstrate robust performance across diverse traffic, illumination, and occlusion conditions, while maintaining real-time capability suitable for resource-constrained platforms.

Future work will focus on improving robustness under extreme conditions such as severe occlusions, night-time, and adverse weather using transformer-based temporal modeling and uncertainty-aware risk estimation. Extensions will also explore trajectory-aware prediction, rider intention modeling, and multi-sensor fusion with radar/IMU for enhanced safety in dynamic traffic scenarios.

References

- [Caesar et al., 2020] Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., and Beijbom, O. (2020). nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631.
- [Hassan et al., 2026] Hassan, M., Kirwin, R., Rakurty, C. S., and Mahajan, A. (2026). Advanced sensor technologies in cutting applications: A review. *Sensors*, 26(3):762.
- [Howard et al., 2017] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- [Kaye et al., 2024] Kaye, S.-A., Nandavar, S., Lewis, I., Blackman, R., Schramm, A., McDonald, M., Oviedo-Trespalacios, O., and Haworth, N. (2024). Exploring beliefs and perceptions towards advanced rider assistance systems (aras) in motorcycle safety. *Transportation research part F: traffic psychology and behaviour*, 102:77–87.
- [Manakitsa et al., 2024] Manakitsa, N., Maraslidis, G. S., Moysis, L., and Fragulis, G. F. (2024). A review of machine learning and deep learning for object detection, semantic segmentation, and human action recognition in machine and robotic vision. *Technologies*, 12(2):15.
- [Mihalj et al., 2022] Mihalj, T., Li, H., Babić, D., Lex, C., Jeudy, M., Zovak, G., Babić, D., and Eichberger, A. (2022). Road infrastructure challenges faced by automated driving: A review. *Applied Sciences*, 12(7):3477.
- [Naweed and Blackman, 2024] Naweed, A. and Blackman, R. (2024). The art of riding safely: A critical examination of advanced rider assistance systems in motorcycle safety discourse. *Transportation Research Part F: Traffic Psychology and Behaviour*, 107:1198–1213.
- [Redmon and Farhadi, 2018] Redmon, J. and Farhadi, A. (2018). Yolo3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
- [Tan and Le, 2019] Tan, M. and Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 6105–6114.
- [Wu et al., 2023] Wu, K., Zhou, Y., Shi, H., Li, X., and Ran, B. (2023). Graph-based interaction-aware multi-modal 2d vehicle trajectory prediction using diffusion graph convolutional networks. *IEEE Transactions on Intelligent Vehicles*, 9(2):3630–3643.
- [Yu et al., 2020] Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., and Darrell, T. (2020). Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645.
- [Zhang et al., 2025] Zhang, X., Wang, H., and Dong, H. (2025). A survey of deep learning-driven 3d object detection: Sensor modalities, technical architectures, and applications. *Sensors*, 25(12):3668.

Vision-Based Robotic Hot-Water Weeding for Precision Agriculture

Gregory Heinrich Mantel, James Florin Petri, Edward Jakunskas, Gerard Lacey

Maynooth University

Abstract

Rising herbicide resistance, labour shortages, and sustainability concerns are driving interest in alternative weed control methods. This paper evaluates hot water as a robotic weed-control method and presents a modular hot-water weeder prototype developed for future open-field trials. A FarmBot-based feasibility study delivered water at 98°C to common weed species using different volumes, nozzle heights, and application frequencies. A single 150 ml treatment eliminated Brome (*Bromus sterilis*), Daisies (*Bellis perennis*), Dandelions (*Taraxacum officinale*), and Docks (*Rumex obtusifolius*), while Chickweed (*Stellaria media*) was eliminated after two treatments. Nettles (*Urtica dioica*) showed foliar injury but survived, and tillage radish (*Raphanus sativus*) showed no visible response. These results informed a modular multi-nozzle hot-water weeder using individually controlled nozzles, RGB vision, and YOLO-based weed detection to customise treatment volume by weed type. A 1.5 m FarmBot-mounted 16-nozzle prototype validated the targeting approach, and a mobile robot-mounted prototype extended the system to a 2 m, 32-nozzle boom with onboard water storage, heating, batteries, and ROS2 control. The vision system achieved 0.8736 precision, 0.742 recall, and 0.7611 mAP@0.5, while system testing achieved an 86.67% integrated detection-and-treatment success rate with a 6.67% false-positive rate. The results demonstrate that hot water can be selectively applied using low-cost RGB vision and modular actuation as a maintainable non-chemical approach to robotic precision weeding.

Keywords: Precision weeding, hot water weeding, agricultural robotics, machine vision, YOLO

1 Introduction

Weeds are one of the largest yield constraints in crop production and can act as vectors for disease and parasites. Reliance on herbicides is becoming unsustainable due to increasing resistance and regulatory constraints linked to ecological risk [Duke, 2017, Bamal et al., 2024, Murdoch, 2018]. Manual weeding is ecologically favourable but limited by labour cost and availability [Win et al., 2025], while mechanical weeding can disturb soil, increase erosion and greenhouse gas emissions, disrupt soil microbiota, and damage nearby crops. Precision weeding addresses these limitations by using machine vision and automation to target individual weeds, reducing chemical use and crop or soil disturbance [Slaughter et al., 2008]. As herbicide development becomes more expensive and robotic systems become more affordable, precision approaches are becoming viable tools for sustainable crop production [Upadhyay et al., 2024, Xie et al., 2025].

Integrated Weed Management (IWM) combines prevention, cultural methods, limited targeted chemical use, and robotic weeding. Preventing weed-seed dispersal through machinery hygiene, using rotation, cover



Figure 1: Hot water weeding evaluation.

crops and polyculture, and targeting individual weeds can reduce herbicide dependence and environmental contamination [Murdoch, 2018, Duke, 2017]. Hot water is attractive within this framework because it requires no external chemical agent, does not disturb the soil, has lower fire risk than flame weeding, and can also be adapted for targeted irrigation or fertigation. The main contributions are: (i) hot-water treatment results for common weed species; (ii) a YOLO-based vision-to-actuation pipeline for nozzle-level treatment; (iii) a 16-nozzle FarmBot validation prototype; and (iv) a 32-nozzle mobile robot-mounted prototype developed for future open-field trials.

2 State of the Art

2.1 Alternative Sustainable Weeding Methods

Mechanical weeding is one of the oldest non-chemical approaches and remains common in organic cereal systems [Quan et al., 2022]. Spring tine harrows and rotary hoes are widely used [Lötjönen and Mikkola, 2000], but they do not target weeds individually and may damage crop plants, particularly near stems or within rows [Lötjönen and Mikkola, 2000]. Reviews of weeding strategies highlight the need for practical, maintainable technologies [Bloomer et al., 2024], and farmer surveys show a preference for modular tools that can be repaired on farm with minimal external support [Hameed et al., 2025].

Other non-chemical treatments include vinegar, laser, and thermal weeding. Wood vinegar can inhibit seed germination at low concentrations, but repeated application, variable species response, and uncertain ecological effects remain limitations [Zhang et al., 2022]. Laser systems such as WeLASER provide precise non-contact treatment but face barriers including high cost, energy demand, fire risk, and maintenance complexity [Andreasen et al., 2024]. Thermal weeding damages plants through protein denaturation and membrane rupture [Peerzada and Chauhan, 2018]. Hot water has shown particular promise because it avoids soil disturbance, reduces fire risk, and causes less surface wear than mechanical or flame-based methods [Hansson, 2002, Sarıtaş and Gürbüz, 2024]. Astatkie and Havard [T. Astatkie, 2007] found hot water effective against young weeds at low speeds and consistent across weather conditions without drift or run-off. Martelloni et al. [Martelloni et al., 2021] showed that hot water reduced regrowth and biomass within 30 days, while noting that hot foam can achieve similar effects with lower doses. However, hot water remains simpler to deploy, requires no foaming agent, and is well suited to precision application.

2.2 Weed Detection and Targeting

Precision weeding requires reliable identification of individual weeds. 2D RGB vision systems are widely used because they are low cost, accessible, and compatible with many detection models. They distinguish weeds and crops using colour, size, morphology, and position [Guzel et al., 2024]. Traditional machine vision works best when plants are visually separated [Slaughter et al., 2008], but 2D imaging can struggle under occlusion [Xie et al., 2023]. Potena et al. [Potena et al., 2016] showed that infrared thermography can improve segmentation under difficult lighting, but at increased sensing cost and complexity. Quan et al. [Quan et al., 2022] demonstrated RGB-based crop–weed detection for intra-row maize weeding using a mechanically actuated tool, illustrating both the potential of vision-guided weeding and the complexity of mechanical end effectors.

Vision-guided spraying offers a simpler actuation route. Terra et al. [Terra et al., 2021] linked real-time weed detection to electronically controlled nozzles for site-specific application, while Wu et al. [Wu et al., 2019] showed that multiple RGB cameras can broaden detection coverage. Similar vision-based selective spraying systems have demonstrated the potential to reduce herbicide use by treating only detected weed locations [Raja et al., 2023]. Commercial systems such as Verdant Robotics' SharpShooter [VerdantRobotics, 2025] demonstrate the potential of targeted application and report large reductions in chemical and labour use, but their sophistication may raise concerns around maintainability and durability. This motivates simpler modular systems for field robotics.

3 My Work

3.1 Validating Hot Water for Weeding

3.1.1 Experiment Setup

To validate boiling water as a cereal-crop weed control method, an experiment studied the effect of water heated to 98°C on common weeds while varying volume, application frequency, and nozzle height. Consultation with local farmers identified Brome, Chickweed, Nettles, grass weeds, and tillage radish as common in European temperate-zone cereal crops. A 3×2 m mown grass area, containing approximately 75% fescue and 25% perennial rye, was also included to investigate Docks, Daisies, and Dandelions for grassland or golf-course-style applications. Rows of oats were planted between weeds to observe crop effects.

The platform was a FarmBot Genesis XL installed on a raised greenhouse bed. It was modified with an in-line water boiler, heat-resistant piping installed in parallel with the cold-water lines, and a solenoid valve for hot-water release. Weeds were planted by the robot in a known coordinate system and received frequent light watering. The FarmBot was controlled using an open-source ROS2 interface. In total, 26 plants of each species were planted, allowing each treatment parameter to be applied to two plants. The same treatment schedule was applied within each tested species group, as shown in Table 1. Some parameters followed [Hansson and Mattsson, 2002]. Plants were imaged using a Luxonis OAK-D Lite before treatment and seven days after treatment, providing both health records and a dataset for future weed-detection models.

Plant No.	Frequency	Nozzle Height	Volume
P1	Control	-	-
P2	Every Three Weeks	18 cm	300 ml
P3	Every Three Weeks	18 cm	150 ml
P4	Every Three Weeks	9 cm	300 ml
P5	Every Three Weeks	9 cm	150 ml
P6	Every Two Weeks	18 cm	300 ml
P7	Every Two Weeks	18 cm	150 ml
P8	Every Two Weeks	9 cm	300 ml
P9	Every Two Weeks	9 cm	150 ml
P10	Every Week	18 cm	300 ml
P11	Every Week	18 cm	150 ml
P12	Every Week	9 cm	150 ml
P13	Every Week	9 cm	300 ml

Table 1: Spray treatment parameters applied to each plant within each tested species group.

3.1.2 Experiment Results & Discussion

The six-week experiment showed clear species-dependent responses. Brome was completely eradicated under all treatment conditions. Chickweed required two applications and would likely benefit from a broader spray pattern to improve full-plant coverage. Tillage radish (*Raphanus sativus*) showed no visible response, indicating high tolerance. Nettles (*Urtica dioica*) sustained foliar injury but survived due to robust root systems, suggesting that earlier and higher-volume treatments may improve control. Dandelions, Daisies, and Docks were effectively killed, including damage to large taproots, though higher water volumes caused collateral grass damage. Neighbouring oat crops were unaffected and produced a strong yield. Overall, with the exception of Nettles and tillage radish, the tested species were controlled using relatively low volumes and low application frequency. Nozzle height had little effect on efficacy, and the results suggest that hot water is most effective on younger weeds before extensive root systems develop.

3.2 Precision Weeder Design & Implementation

3.2.1 Computer Vision

The computer vision system was developed to detect weeds, classify them by species, and provide spatial information for nozzle-level hot-water treatment. Unlike a standalone image-classification task, the output of the vision model had to be suitable for robotic actuation. Each detection therefore needed to provide a weed

class, confidence score, and image-space location that could be converted into a centroid and passed to the control pipeline.

The dataset formed during the hot-water validation experiment was expanded using a third-party dataset [AUENG Vision Group, 2025]. This increased variation in lighting conditions, soil types, plant growth stages, and image backgrounds. Images collected during the FarmBot experiment were particularly useful because they represented the same camera perspective, crop context, and treatment environment used during system validation. The combined dataset was labelled using bounding boxes, as this format is directly compatible with YOLO-based object detection and provides a simple method for extracting weed centroids for downstream robotic control.

A YOLOv8n model was trained using mAP, precision, and recall as evaluation metrics. Image augmentations including Flip, Crop, Mosaic, and Shear were applied to improve robustness to variation in weed orientation, scale, partial visibility, and field background. The best-performing model checkpoint was selected for inference. The trained model achieved 0.8736 precision, 0.742 recall, and 0.7611 mAP@0.5. This performance showed a useful balance between minimizing false positives, which could result in crop damage, and maintaining sufficient true positive detections for practical weed treatment. Training curves showed steady reductions in classification and bounding-box losses, while validation losses fluctuated slightly due to dataset variability but remained stable overall.

Compared with existing RGB crop–weed detection work, the achieved YOLOv8n precision of 0.8736 and mAP@0.5 of 0.7611 show that the detector provides a practical baseline for real-time robotic treatment. Direct numerical comparison is limited because weed-detection results are strongly affected by species, crop type, camera viewpoint, lighting conditions, dataset composition, and evaluation protocol. However, unlike mechanically actuated intra-row systems [Quan et al., 2022], the proposed approach links RGB detections to selective nozzle actuation, and unlike conventional site-specific spraying systems [Terra et al., 2021, Raja et al., 2023], the treatment medium is hot water rather than herbicide.

For robotic use, the detector output was converted into a treatment decision rather than being used only as an image-level result. Each bounding box was converted to a centroid, associated with its predicted weed class and confidence score, and passed to the ROS2 control pipeline. The centroid position was then related to the camera frame and used to estimate whether the weed lay within the effective spray corridor. Detections outside the workable region were ignored, while valid detections were tracked as the robot moved forward. This allowed the system to convert image-space detections into nozzle-specific treatment commands.



Figure 2: Custom trained weed detection model validated on Chickweed and oat crop.

The use of bounding-box detection was chosen as a practical compromise between localization accuracy and computational efficiency. Pixel-level segmentation could improve localization where plants overlap, but it would also increase annotation effort and processing requirements. For the current nozzle spacing and spray pattern, bounding-box centroids provided sufficient spatial information for initial prototype validation. Future work will expand dataset diversity across species, growth stages, lighting conditions, and outdoor field environments. Segmentation may also be introduced later to improve targeting where weeds and crops overlap or where the weed canopy extends beyond the bounding-box centre.

3.2.2 Robot Design

A 1.5 m FarmBot-mounted sprayer array was first developed to validate the vision-guided targeting method. The array used 16 individually controlled spray nozzles mounted to an aluminium frame. Each nozzle was regulated by a solenoid valve actuated through two 8-channel relay modules and a 16-channel PWM driver, in-

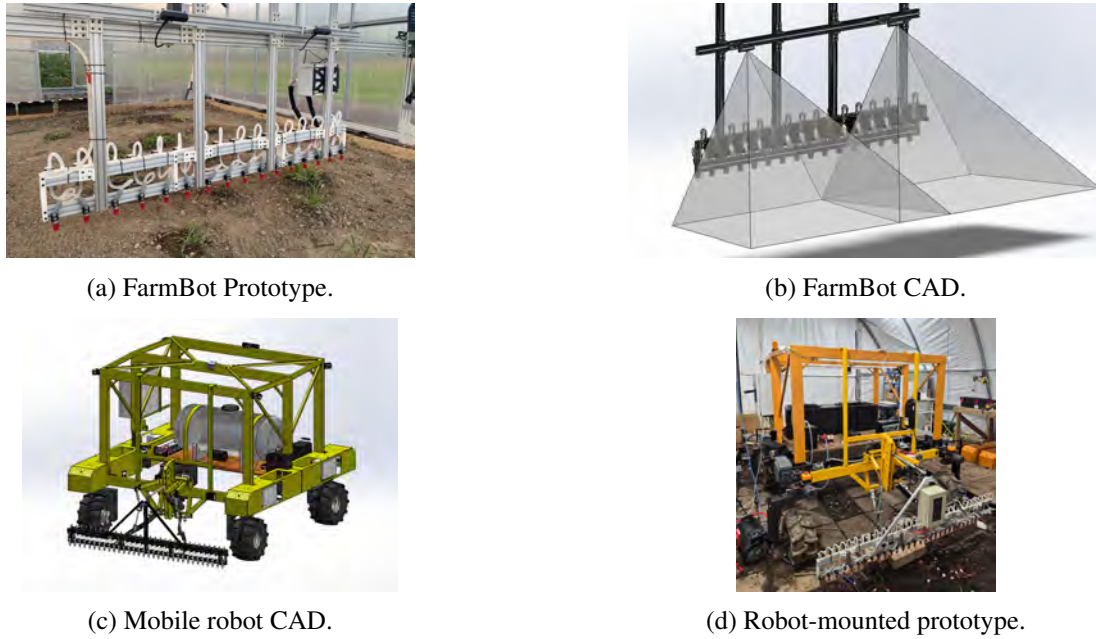


Figure 3: Development from the FarmBot validation prototype to the robot-mounted prototype.

terfaced with a Raspberry Pi 4B through a custom daughterboard. Two Luxonis OAK-D Lite cameras provided image input, while a networked computer ran the YOLOv8n model and returned detected weed centroids. The validation assembly is shown in Fig. 3.

A mobile robot-mounted prototype was then implemented as a hot-water weeding tool. The boom was scaled to 2 m to match the mobile robot’s track width and fitted with 32 nozzles, improving targeting resolution. It was mounted to the three-point hitch using custom Category 1 steel mounts, with the lower arms carrying the vertical load and the top link setting boom pitch.

The onboard prototype system included a 200 L water tank, two 24 V LiFePO₄ batteries, a 2000 W inverter, a 1500 W instant boiler, a 24 V diaphragm pump, an immersion heater, and high-temperature silicone tubing feeding the 32-nozzle manifold. Water was drawn from the tank, preheated, passed through the boiler, and delivered to the boom above 95°C under typical operating flow. For the mobile robot-mounted prototype, an OAK-D PoE camera and the robot’s onboard NVIDIA Orin replaced the external FarmBot computer, while ROS2 nodes handled camera input, inference, weed tracking, odometry, and nozzle control.

3.2.3 Control Architecture

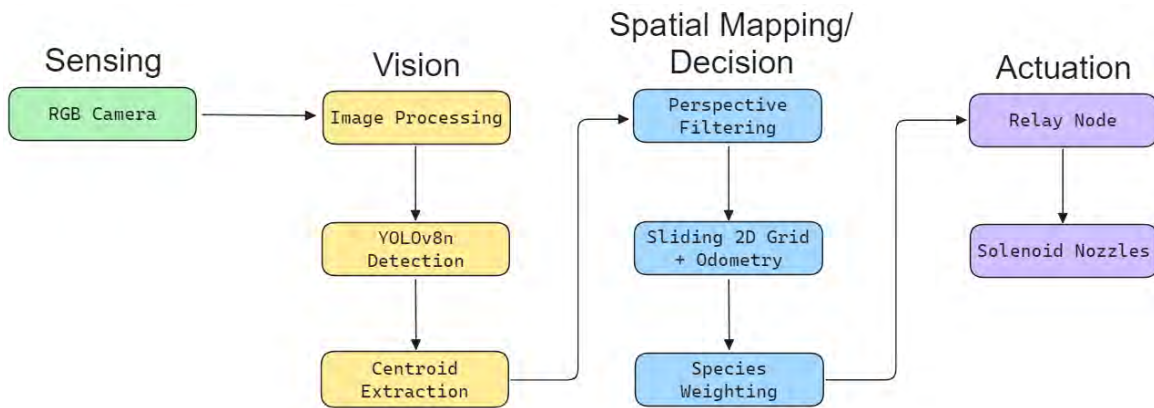


Figure 4: System architecture linking RGB weed detection to nozzle-level hot-water actuation.

The ROS2 control architecture, shown in Fig. 4, is modular and platform independent. Communication with the main robot is limited to master deactivation, initialization configuration, and odometry. Each camera

has its own ROS run node and processing node, with both processing nodes communicating independently with the relay control node. The processing node contains image processing, model inference, and interpretation. The odometry node provides distance travelled, allowing weed positions to be represented relative to the nozzle array over time.

Due to camera perspective, some visible weeds are outside the sprayer trajectory shown in Fig. 5. These are removed using trigonometric height-wise planar weed localization based on the camera frustum, enabling real-time localization without repeated camera recalibration. The height-wise variable computation and nozzle mapping are expressed as:

$$l_p = 2h \tan \frac{FOV_h}{2}, \quad L_p = 2\sqrt{h^2 + (h \tan(FOV_v))^2} \tan \frac{FOV_h}{2}, \quad l' = \frac{l}{L} Y_{cam} \quad (1)$$

$$\forall P_i(x_i, y_i): \quad (a+b) = \frac{l'}{2} + \frac{y_i}{y_{cam}} \left(\frac{X_{cam}}{2} - \frac{l'}{2} \right), \quad b = \left| x_i - \frac{X_{cam}}{2} \right|$$

If $b > a+b$, ignore detection. Otherwise: (2)

$$X_r = \begin{cases} \frac{X_{cam}}{2} - \frac{b}{a+b} \frac{X_{cam}}{2}, & x_i < \frac{X_{cam}}{2} \\ \frac{X_{cam}}{2} + \frac{b}{a+b} \frac{X_{cam}}{2}, & \text{otherwise} \end{cases}$$

Here, h is camera height, FOV_h and FOV_v are camera fields of view, and X_{cam} and Y_{cam} are image dimensions. $P_i(x_i, y_i)$ is the detected weed centroid, X_r is the nozzle-mapping coordinate, l_p , L_p , l' , l , and L are perspective-scaling terms, and a and b define the spray-trajectory acceptance region.

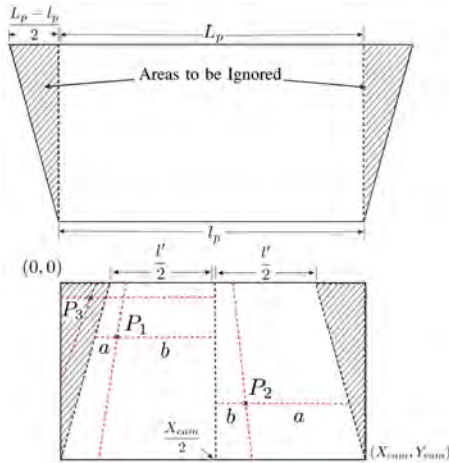


Figure 5: Camera ground footprint and corresponding image frame. P_3 is ignored as it lies outside of the nozzle trajectory

A weed prediction score was computed by combining detection confidence with observation duration. As the robot moved, weeds were tracked toward the nozzle array using a sliding 2D grid with 8 columns per camera and a variable number of perspective-scaled rows. When a weed was detected, its score was assigned to the relevant cell. As odometry advanced, lower rows were popped and new upper rows added. If a detection exceeded a calibrated threshold, the nozzle corresponding to that row was activated once popped. The score was further weighted by weed species so valves could remain open for different durations and deliver species-specific volumes aligned with the hot-water experiment. Because no pressure control was present, simultaneous nozzle activation reduced pressure per nozzle; this was considered during treatment timing.

3.3 Precision Weeder Evaluation

Four experiments evaluated precision, accuracy, efficacy, and reliability. Targeting precision was calculated by comparing the centre point of a weed with the centroid of the surrounding wet area produced by the weeder. Measurements from Brome, Chickweed, and Nettle are shown in Table 2, where M1–M3 are repeated offset measurements.

Weed	M1	M2	M3	Mean	Std. Dev.
Brome	27	35	8	23.3	11.2
Chickweed	15	50	25	30.0	15.9
Nettle	25	60	40	41.7	14.6

Table 2: Targeting offset measurements in mm.

Across species, the mean offset was 31.7 mm with an average standard deviation of 13.9 mm. Compared to the FarmBot nozzle spacing of 93.75 mm, these offsets show that the system targeted the correct region, but also that a single nozzle may not cover all centroid positions. Finer nozzle resolution is therefore beneficial and motivated the 32-nozzle robot-mounted boom.

In a controlled test, the system detected and treated 26 out of 30 weeds, giving an 86.67% integrated detection-and-treatment success rate. This provides a system-level result, while the YOLOv8n precision value of 0.8736 reports model-level detection performance. Reliability was also demonstrated, as only one of 15 crop plants, including tomato, oats, barley, and wheat, was mistakenly targeted, corresponding to 93.33% crop protection success and a 6.67% false-positive rate. These results indicate strong baseline accuracy and reliability, with potential improvement through expanded training data and inclusion of non-weed detections.

Dose delivery was tested by directing a nozzle into a measuring jug as the weeder advanced toward a weed. The system was calibrated to deliver at least 150 ml to Brome and Chickweed and 300 ml to Nettle. The 150 ml dose ranged from 130–190 ml with a standard deviation of 28.3 ml, while the 300 ml dose ranged from 250–390 ml with a standard deviation of 60.6 ml. Variation was mainly attributed to nozzle tolerances, adjustment, and pressure reduction during simultaneous actuation. Overall, the robot-mounted prototype extends the FarmBot validation system to a mobile platform with onboard water, heat, power, sensing, and ROS2 control.

4 Conclusion, Limitations and Future Work

This paper investigated hot water as a weed-control method for future open-field robotic application. A gantry-based robot applied water at 98°C to weeds common in temperate cereal farms and grassland environments. Brome, Daisies, Dandelions, and Docks were controlled after one 150 ml treatment, Chickweed required two treatments, Nettles survived due to robust root systems, and tillage radish showed no visible response. A 1.5 m FarmBot-mounted precision weeding tool was then evaluated using 16 electronically controlled nozzles and a vision system trained for Brome, Chickweed, and Nettle detection. The system achieved an 86.67% integrated detection-and-treatment success rate, 93.33% crop protection success, and species-specific dose delivery. The design was extended to a mobile robot-mounted prototype with a 2 m, 32-nozzle boom, onboard water storage, heating, power, sensing, NVIDIA Orin inference, and ROS2 control.

The main limitations are targeting resolution, pressure variation during simultaneous nozzle activation, operating speed, energy demand, and model robustness under diverse field conditions. Future work will include open-field trials on the mobile robot, evaluation of the 32-nozzle system, improved heating and pressure control, expanded training data, and workshops with farmers to assess adoption, maintainability, and usability.

References

- [Andreasen et al., 2024] Andreasen, C. et al. (2024). Sustainable weed management in agriculture with laser-based autonomous tools: Environmental performance assessment. *The International Journal of Life Cycle Assessment*, 29:1021–1052.
- [AUENG Vision Group, 2025] AUENG Vision Group (2025). OPPD: Open Plant Phenotyping Dataset. <https://gitlab.au.dk/AUENG-Vision/OPPD/-/tree/master>. Accessed: 2025-06-04.
- [Bamal et al., 2024] Bamal, D., Duhan, A., Pal, A., and Beniwal, R. (2024). Herbicide risks to non-target species and the environment: A review. *Environmental Science and Pollution Research*, 31:2977–3032.
- [Bloomer et al., 2024] Bloomer, D. J., Harrington, K. C., Ghanizadeh, H., and James, T. K. (2024). Robots and shocks: emerging non-herbicide weed control options for vegetable and arable cropping. *New Zealand Journal of Agricultural Research*, 67(1):81–103.
- [Duke, 2017] Duke, S. O. (2017). Weed management in 2050: Perspectives on the future of weed science. *Weed Science*, 65(4):468–470.
- [Guzel et al., 2024] Guzel, M., Karakaya, M., and Yildirim, M. (2024). Deep learning for image-based detection of weeds from emergence to maturity in wheat fields. *Smart Agricultural Technology*, 6:100235.

- [Hameed et al., 2025] Hameed, R. U., Meade, C., and Lacey, G. (2025). Technology advancements and the needs of farmers: Mapping gaps and opportunities in row crop farming. *Agriculture*, 15(15):1664.
- [Hansson, 2002] Hansson, D. (2002). Hot water for weed control on urban hard surface areas. In *5th EWRS Workshop on Physical and Cultural Weed Control*.
- [Hansson and Mattsson, 2002] Hansson, D. and Mattsson, J. (2002). Effect of drop size, water flow, wetting agent and water temperature on hot-water weed control. *Journal of Agricultural Engineering*.
- [Lötjönen and Mikkola, 2000] Lötjönen, T. and Mikkola, H. J. (2000). Three mechanical weed control techniques in spring cereals. *Agricultural and Food Science*, 9(4):269–278.
- [Martelloni et al., 2021] Martelloni, L., Frascioni, C., Sportelli, M., Fontanelli, M., Raffaelli, M., and Peruzzi, A. (2021). Hot foam and hot water for weed control: A comparison. *Journal of Agricultural Engineering*, LII(2):229–236.
- [Murdoch, 2018] Murdoch, A. (2018). Sustainable weed control in small grain cereals (wheat/barley). *Weed Control: Sustainability, Hazards and Risks in Cropping Systems Worldwide*.
- [Peerzada and Chauhan, 2018] Peerzada, A. M. and Chauhan, B. S. (2018). Thermal weed control: History, mechanisms, and impacts. In Abouziena, M. Z., editor, *Non-Chemical Weed Control*, pages 9–31. Academic Press.
- [Potena et al., 2016] Potena, C., Nardi, D., and Pretto, A. (2016). Fast and accurate crop and weed identification with summarized train sets for precision agriculture. In *Proceedings of the 14th International Conference on Intelligent Autonomous Systems (IAS-14)*, pages 105–121. Springer.
- [Quan et al., 2022] Quan, L., Jiang, W., Li, H., Li, H., Wang, Q., and Chen, L. (2022). Intelligent intra-row robotic weeding system combining deep learning technology with a targeted weeding mode. *Biosystems Engineering*, 216:13–31.
- [Raja et al., 2023] Raja, R., Slaughter, D. C., Fennimore, S. A., and Siemens, M. C. (2023). Real-time control of high-resolution micro-jet sprayer integrated with machine vision for precision weed control. *Biosystems Engineering*, 228:31–48.
- [Saritaş and Gürbüz, 2024] Saritaş, A. G. and Gürbüz, R. (2024). Effects of hot water obtained by solar energy on the weeds *Convolvulus arvensis* L., *Setaria viridis* (L.) p. beauv. and *Amaranthus retroflexus* L. *Yüzüncü Yıl University Journal of Agricultural Sciences*, 34(2):346–353.
- [Slaughter et al., 2008] Slaughter, D. C., Giles, D. K., and Downey, D. (2008). Technology for automation of weed control in specialty crops. *Weed Technology*, 22(3):364–370.
- [T. Astatkie, 2007] T. Astatkie, P. Havard, J. A. (2007). Efficiency and performance of hot water, infrared, and flame weeder thermal units to control weeds. *Biological Agriculture & Horticulture*, 25:275–289.
- [Terra et al., 2021] Terra, F. P., do Nascimento, G. H., Duarte, G. A., and Jr, P. L. J. D. (2021). Autonomous agricultural sprayer using machine vision and nozzle control. *Journal of Intelligent & Robotic Systems*, 102(38):1–13.
- [Upadhyay et al., 2024] Upadhyay, A., Zhang, Y., Koparan, C., Rai, N., Howatt, K., Bajwa, S., and Sun, X. (2024). Advances in ground robotic technologies for site-specific weed management in precision agriculture: A review. *Computers and Electronics in Agriculture*, 225:109363.
- [VerdantRobotics, 2025] VerdantRobotics (2025). Product - sharpshooter. <https://www.verdantrobotics.com/product>. Accessed: 2025-09-04.
- [Win et al., 2025] Win, M. T., Rutledge, Z., and Maredia, M. K. (2025). Labor shortages and farmer adaptation strategies. *Applied Economic Perspectives and Policy*. Forthcoming.
- [Wu et al., 2019] Wu, Q., Aravecchia, S., and Pradalier, C. (2019). Design and implementation of computer vision based in-row weeding system. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9645–9651. IEEE.
- [Xie et al., 2025] Xie, S., Goldberg, K., and Song, D. (2025). Energy efficient planning for repetitive heterogeneous tasks in precision agriculture. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7139–7145.
- [Xie et al., 2023] Xie, S., Hu, C., Wang, D., Johnson, J., Bagavathiannan, M., and Song, D. (2023). Coupled active perception and manipulation planning for a mobile manipulator in precision agriculture applications. *arXiv preprint*.
- [Zhang et al., 2022] Zhang, Z., Yu, J., Chen, L., et al. (2022). Evaluating the herbicidal properties of wood vinegar derived from pruned apple branches. *Agronomy*, 12(3120):1–13.

Training-Free Geometric Stabilization for Monocular 3D Detection in Advanced Rider-Assistance Systems

Lakshman Prabhu Balsubiramaniam*, Satyajit Nayak, Cyril Poulet, and Ganesh Sistu

anSWer, Valeo, India

Abstract

Monocular 3D object detectors fail on dynamic platforms like motorcycles, where roll and pitch variations violate the stable-camera assumption learned from automotive datasets. This geometric domain shift, which 2D augmentations cannot model, causes catastrophic 3D localization errors. We introduce Lean-Aware Geometric Stabilization (LGS), a training-free pre-processing module that corrects this instability at inference time. LGS applies a physically-grounded inverse reprojecton to warp the input image to a canonical, stable orientation before it reaches a frozen, off-the-shelf detector. On a challenging motorcycle dataset, LGS recovers BEV AP@0.2 from 0.06 to 0.20 under a 10° roll drift, nearly matching the original 0.22 performance on stable views. The module adds only 5-10 ms of CPU latency, offering a practical method to deploy existing 3D detectors on geometrically unstable platforms without retraining.

Keywords: Monocular 3D Object Detection, Geometric Robustness, Viewpoint Augmentation, Training-Free Adaptation, Advanced Rider-Assistance Systems (ARAS).

1 Introduction

Monocular 3D object detectors [Liu et al. (2020), Zhang et al. (2021), Ma et al. (2021)] learn strong geometric priors from large-scale automotive datasets. This success is predicated on a stable camera platform that maintains a fixed extrinsic pose and a level horizon.

This "automotive assumption" is fundamentally violated on dynamic platforms like motorcycles, which exhibit continuous roll and pitch. During dynamic maneuvering (e.g., leaning in turns), vehicles exhibit continuous roll and pitch variations—violating the level-horizon extrinsic baseline. We term this Dynamic Extrinsic Drift (DED), which induces geometric domain shift in camera projection while visual appearances remain identical. Consequently, a detector trained on stable data (e.g., KITTI [Geiger et al. (2012)]) fails catastrophically when its learned priors are invalidated, as shown in Figure 1. Standard 2D training augmentations (e.g., planar rotations) fail because they treat pitch/roll as affine image shifts, ignoring non-linear perspective distortions

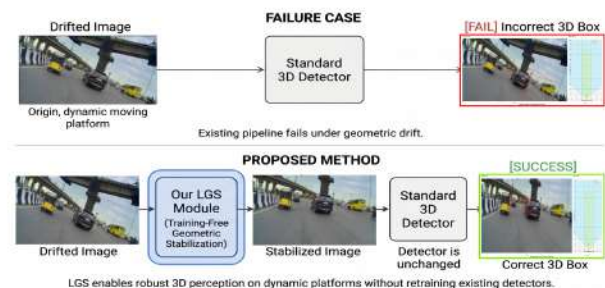


Figure 1: **Core Idea:** (Top) A standard detector fails on a geometrically drifted image. (Bottom) Our training-free LGS module stabilizes the input, enabling the same detector to succeed without retraining. and violating depth projection physics. This work

*Corresponding author: lakshmanprabhu.b@gmail.com

therefore addresses whether a lightweight, training-free geometric stabilization can counteract this domain shift, enabling a pre-trained detector to generalize to dynamic motion without costly retraining.

We introduce *Lean-Aware Geometric Stabilization (LGS)*, a model-agnostic pre-processing module that corrects for extrinsic drift at inference time. Our contributions are:

1. We formalize DED, a physically-grounded viewpoint augmentation, to create a challenging benchmark for evaluating geometric robustness.
2. We propose LGS, a model-agnostic, training-free pre-processing module that restores geometric consistency to input images at inference time.
3. We demonstrate empirically that LGS enables a detector trained only on stable data to generalize to out-of-distribution camera poses, substantially recovering 3D localization accuracy.

2 Prior Art Study

Our work is positioned at the intersection of geometric robustness, test-time adaptation, and perception for dynamic platforms. **Geometric Robustness.** The fragility of monocular 3D detectors to camera pose variations is well-documented [Moon et al. (2023)]. A common defense is training-time augmentation; for instance, GRAMO [Guan et al. (2024)] applies ground-plane constraint resampling to synthesize geometrically consistent samples during training. Our DED simulation can serve this purpose, but our primary contribution, LGS, is a complementary, inference-time stabilization technique that robustifies existing detectors without retraining. This is distinct from work on camera *intrinsic* robustness [Zhang et al. (2026)], as we focus on *extrinsic* drift. **Test-Time Adaptation (TTA).** Another strategy is to adapt the model at test time. MonoTTA [Lin et al. (2024)] uses self-supervision to adapt to domain shifts, while other methods rectify latent features using online pose estimates [Zhou et al. (2021)]. These approaches often require access to model internals or add significant overhead. In contrast, LGS is a model-agnostic pre-processing step. By correcting geometry in the image space *before* the detector, it provides a simpler, modular, and training-free solution. It is a form of test-time adaptation that operates on the input data, not the model, and is distinct from methods correcting rolling shutter artifacts [Mo et al. (2022)], which address intra-frame distortions. **Attitude Estimation and Hardware Compensation.** LGS assumes vehicle roll and pitch are available from an IMU. The problem of estimating these angles [Ok et al. (2021)] is orthogonal to our work; we focus on how to use this information for visual stabilization. Similarly, our approach is distinct from hardware-level compensation like Optical Image Stabilization (OIS) [Liu et al. (2022)], which handles high-frequency camera shake. LGS addresses a different problem: large-scale, low-frequency changes in the vehicle’s extrinsic pose (e.g., a motorcycle lean), a challenge not addressed by OIS.

3 Methodology: Geometric Augmentation and Stabilization

Our framework is built on the principle of virtual camera reprojection. We first define the core geometric transformation before detailing its dual use for simulating drift (DED) and correcting it (LGS).

3.1 Geometric Formulation

The core of our method is a physically-grounded warp that maps pixels from a source camera view to a virtual, rotated one. Assuming two cameras share an optical center and intrinsic matrix $\mathbf{K}$, but are rotated by $\mathbf{R}_{\text{drift}}$ relative to each other, we can find the location $\mathbf{x}_{\text{src}}$ in the source image corresponding to a pixel $\mathbf{x}_{\text{dst}}$ in the destination (virtual) image via inverse reprojection:

$$\mathbf{x}_{\text{src}} \sim \mathbf{K} \mathbf{R}_{\text{drift}}^{-1} \mathbf{K}^{-1} \mathbf{x}_{\text{dst}} \quad (1)$$

This equation traces a ray from the virtual camera’s image plane, rotates it into the source camera’s frame, and re-projects it. The drift rotation $\mathbf{R}_{\text{drift}}$ models the orientation change, which we parameterize by roll (θ_r) and pitch (θ_p) angles. The rotation is constructed in the camera’s local coordinate system (X-right, Y-down, Z-forward):

$$\mathbf{R}_{\text{drift}} = \mathbf{R}_z(\theta_r)\mathbf{R}_x(\theta_p) \quad (2)$$

This formulation is powerful because it is agnostic to any global ‘world’ coordinate system used for 3D annotations. It operates purely on camera geometry, making it adaptable to diverse dataset conventions.

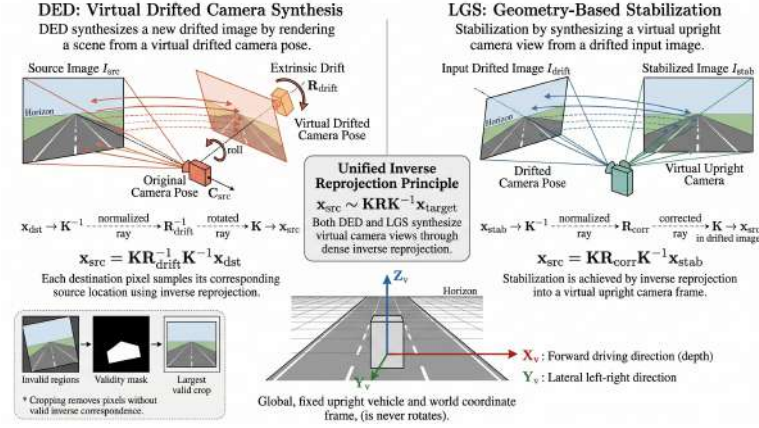


Figure 2: **Geometric Principle of DED and LGS:** Our framework is based on inverse reprojection. (Left) To simulate drift (DED), we trace pixels from a virtual drifted camera back to the original source image using the inverse drift rotation ($\mathbf{R}_{\text{drift}}^{-1}$). (Right) To correct drift (LGS), we trace pixels from a virtual upright camera back to the drifted input image using the corrective rotation ($\mathbf{R}_{\text{corr}}$). This visually explains the geometry-consistent warp defined in Eq. 1.

3.2 Dynamic Extrinsic Drift (DED) Simulation

To evaluate model robustness, we use Eq. 1 to simulate the effect of drift. This process, which we call Dynamic Extrinsic Drift (DED), generates a new, geometrically consistent image from the perspective of a virtual, drifted camera. The resulting map is used with bilinear interpolation to produce the warped image. To remove boundary artifacts, the warped image is cropped to the largest axis-aligned inner rectangle using corner ray-casting; while this slightly reduces peripheral FOV, key central drivable regions remain preserved.

Crucially, DED is a full 3D augmentation that transforms 3D ground truth labels in addition to the image, a key differentiator from 2D-only methods. The exact transformation depends on the annotation coordinate system. For datasets with a stable world frame (e.g., our ARAS dataset), the drift $\mathbf{R}_{\text{drift}}$ updates the camera’s extrinsic rotation matrix. For datasets where annotations are in the camera’s coordinate system (e.g., KITTI), the 3D ground truth points are simply rotated by $\mathbf{R}_{\text{drift}}$. This rigorous label transformation (Figures 3, 4) is essential for correct model evaluation.



Figure 3: **DED on our ARAS Dataset.** (Top) Original image and 3D box. (Middle) Warped images for different drift parameters. (Bottom) Final cropped images. Our method applies the same 3D rotation to the ground truth, producing new, physically-correct labels for the augmented views. Drifts (L-R): (R:+20°, P:+10°), (R:-20°, P:+5°), (R:-10°, P:-5°).



Figure 4: **DED Generality on KITTI.** We apply DED to KITTI to demonstrate the generality of our augmentation. (Top) Original image. (Middle) Warped images showing perspective distortion. (Bottom) Final cropped images with transformed ground truth. This creates a challenging benchmark for geometric robustness on standard automotive data.



Figure 5: **DED on nuScenes.** To further prove generality, we apply DED to the nuScenes dataset [Caesar et al. (2020)], which also uses a camera-centric coordinate system. This demonstrates DED’s utility as a cross-dataset tool for robustness analysis. Drifts (L-R): (R:+15°, P:+8°), (R:0°, P:+8°), (R:-15°, P:-8°).

3.3 Lean-Aware Geometric Stabilization (LGS)

LGS is the corrective counterpart to DED, applied at inference time. Given a drifted input image and an estimate of the drift angles (e.g., from an IMU), LGS computes a corrective rotation $\mathbf{R}_{\text{corr}} = \mathbf{R}_{\text{drift}}^{-1}$. It then applies the same inverse reprojection warp from Eq. 1 to transform the image back to a stable, upright orientation that matches the detector’s training distribution. This stabilized image is then passed to the off-the-shelf detector, as shown in Figure 1. The entire process is a single, efficient pre-processing step.

4 Experiments

Our experiments are designed to quantify the performance degradation caused by DED and measure the recovery provided by LGS.

4.1 Setup

Datasets and Protocol. We use a challenging in-house **ARAS dataset** from a motorcycle and the standard **KITTI benchmark** [Geiger et al. (2012)]. To mirror real-world lean dynamics captured in dataset telemetry, we fine-tune a SMOKE detector [Liu et al. (2020)] on stable ARAS subsets (roll $< 5^\circ$) and test on simulated DED sequences matching real-world lean trajectories. This specialist model is then evaluated on its ability to generalize to images with significant, unseen drift (up to 25°), as shown in Figure 6. We evaluate three scenarios: (1) **Baseline** (on original images), (2) **Drifted** (on DED-warped images), and (3) **Stabilized** (Drifted images corrected by LGS). We report BEV AP@0.2 for 3D localization and Yaw Variance for orientation stability. For ablation, we compare against a naive 2D rotation correction. We also show DED’s generality on nuScenes [Caesar et al. (2020)] (Figs. 4, 5).

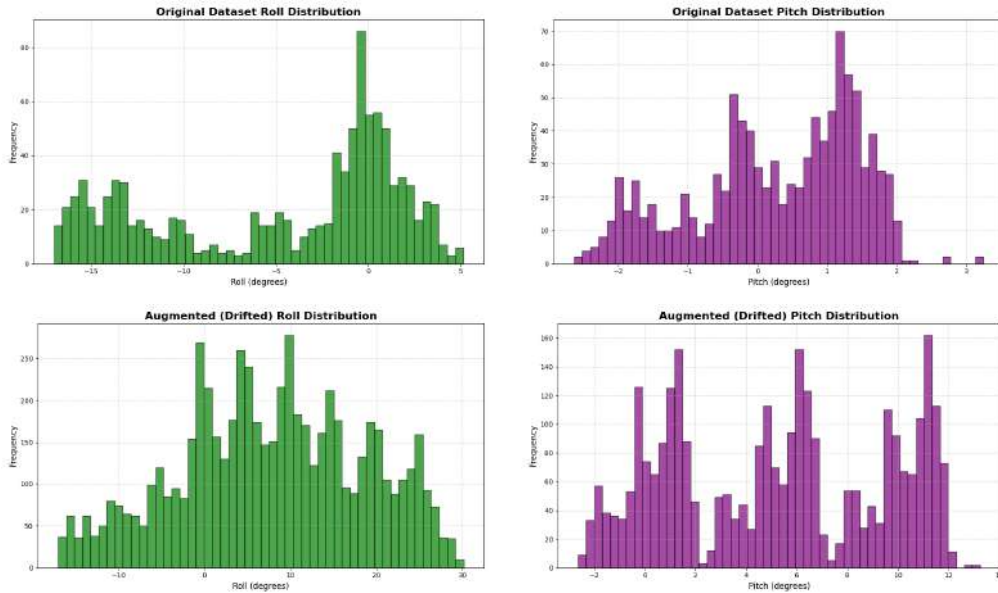


Figure 6: **Geometric Domain Shift for Evaluation.** The training set distributions for roll and pitch (top row) contain only stable views near 0° . In contrast, our DED-augmented test set (bottom row) covers a wide range of angles, creating a challenging OOD scenario to test generalization.

4.2 Results and Analysis

Quantitative Results. As shown in Table 1, the performance of the baseline model, trained on upright images, degrades severely under unseen drift. At 10° roll, BEV AP@0.2 plummets from 0.224 to

0.056, confirming the model’s failure to generalize geometrically. LGS provides a substantial recovery, restoring performance to 0.199 (+0.143 AP), nearly reaching the original performance on stable views and demonstrating its effectiveness at bridging the geometric domain gap.

Table 1: Performance of the baseline model, trained on stable images, degrades catastrophically under roll-induced DED. LGS substantially recovers performance, enabling generalization to unseen drift angles at inference time. Best results on drifted data are in bold.

Roll Drift	Method	BEV AP@0.2 $\uparrow$	Yaw Variance (rad ²) $\downarrow$
0°	Baseline	0.224	0.01
10°	Drifted	0.056	0.35
	LGS (Ours)	0.199 ($\Delta+0.143$)	0.04 ($\Delta-0.31$)

4.3 Ablation Study

To prove our 3D reprojection necessity, we compare LGS against naive 2D correction (Table 2), which simply applies inverse 2D affine image rotation without accounting for intrinsic focal length K or 3D perspective rays. The results are unequivocal: simple 2D rotation is geometrically insufficient and degrades performance further, confirming that correctly modeling the 3D-to-2D projection is essential.

Table 2: Ablation study at 10° roll drift. The naive 2D correction is insufficient, proving the necessity of our physically-grounded geometric reprojection.

Method (at 10° roll)	BEV AP@0.2 $\uparrow$	Yaw Variance (rad ²) $\downarrow$
Drifted (No Correction)	0.056	0.35
+ Naive 2D Un-rotation	0.040	0.75
+ LGS (Ours)	0.199	0.04

4.4 Qualitative Analysis

Figure 7 provides a visual comparison. On drifted images, the detector is confused by distorted geometric cues, leading to catastrophic failures. After our LGS module stabilizes the input, the same detector produces a precise 3D bounding box, directly illustrating how LGS bridges the geometric domain gap.

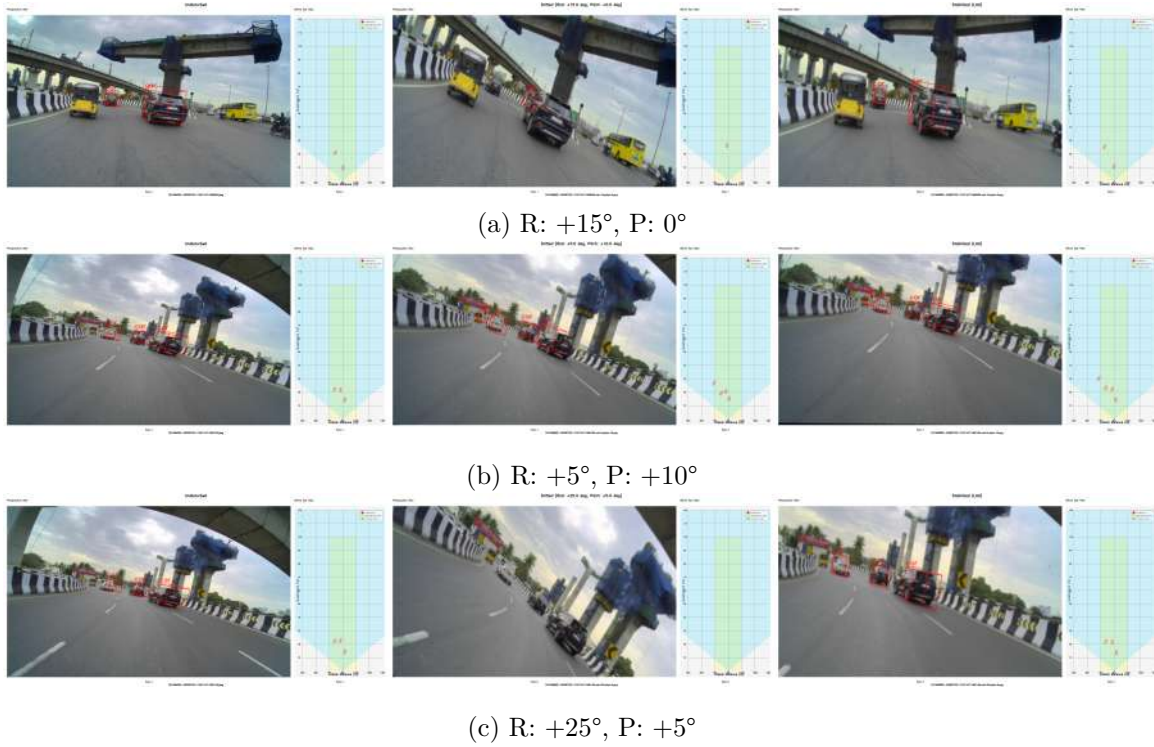


Figure 7: Qualitative Analysis: Each row shows a test case with its drift angles (R: Roll, P: Pitch). Left: Baseline prediction on original image. Middle: Catastrophic failure on DED-warped image. Right: Recovered, accurate prediction after applying our LGS module. LGS enables robust detection under significant, unseen drift.

4.5 Runtime Analysis

LGS is computationally efficient. Implemented in OpenCV on a CPU, the warp adds only **5-10 ms** of latency. With a detector inference time of 39.9 ms on a Tesla T4 GPU, the total pipeline latency is **45-50 ms** per frame (>20 FPS). This minimal overhead makes LGS practical for real-time deployment. Sensitivity tests show LGS maintains robust detection under typical IMU noise ($\pm 1^\circ$ angle jitter) and temporal lag (< 20 ms sensor delay).

5 Conclusion

We have demonstrated that dynamic camera extrinsics on platforms like motorcycles cause a critical geometric domain shift that cripples standard monocular 3D detectors. To solve this, we introduced Lean-Aware Geometric Stabilization (LGS), a lightweight, training-free pre-processing module that restores geometric consistency at inference time. By applying a physically-grounded inverse reprojection, LGS enables a frozen, off-the-shelf detector to generalize to significant, unseen camera poses, substantially recovering 3D localization performance. This work not only provides an immediate, practical solution for deploying existing detectors on dynamic platforms but also opens future avenues. These include using our DED simulation as a training-time augmentation to build inherent robustness into models and extending LGS to handle extreme angles and sensor noise. Ultimately, our framework underscores the primacy of geometric integrity and offers an efficient path toward robust 3D perception in challenging, real-world environments.

References

- [Liu et al. (2020)] Liu, Z., Wu, Z., & Tóth, R. (2020). "SMOKE: Single-Stage Monocular 3D Object Detector via Keypoint Estimation." *CVPR Workshops*, pp. 4289-4298.
- [Zhang et al. (2021)] Zhang, Y., Lu, J., & Zhou, J. (2021). "Objects Are Different: Flexible Monocular 3D Object Detection." *CVPR*, pp. 3289-3298.
- [Ma et al. (2021)] Ma, X., et al. (2021). "Delving into Localization Errors for Monocular 3D Object Detection." *CVPR*, pp. 339-348.
- [Geiger et al. (2012)] Geiger, A., Lenz, P., & Urtasun, R. (2012). "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite." *CVPR*, pp. 3354-3361.
- [Guan et al. (2024)] Guan, H., Song, C. & Zhang, Z. (2024). "GRAMO: geometric resampling augmentation for monocular 3D object detection." *Frontiers of Computer Science*, 18, 185706. <https://doi.org/10.1007/s11704-023-3242-2>
- [Lin et al. (2024)] Lin, H., Zhang, Y., Niu, S., Cui, S., & Li, Z. (2024). "Fully Test-Time Adaptation for Monocular 3D Object Detection." In *European Conference on Computer Vision*.
- [Zhou et al. (2021)] Zhou, Y., He, Y., Zhu, H., Wang, C., Li, H., & Jiang, Q. (2021). "Monocular 3D Object Detection: An Extrinsic Parameter Free Approach." *arXiv preprint arXiv:2106.15796*.
- [Moon et al. (2023)] Moon, S., Bae, J., & Im, S. (2023). "Rotation Matters: Generalized Monocular 3D Object Detection for Various Camera Systems." *arXiv preprint arXiv:2310.05366*.
- [Zhang et al. (2026)] Zhang, Z., Kumar, A., Liu, X. (2026). "Towards Intrinsic-Aware Monocular 3D Object Detection." *arXiv preprint arXiv:2603.27059*.
- [Chen et al. (2021)] Chen, H., Huang, Y., Tian, W., Gao, Z., & Xiong, L. (2021). "MonoRUn: Monocular 3D Object Detection by Reconstruction and Uncertainty Propagation." In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10374-10383.
- [Caesar et al. (2020)] Caesar, H., et al. (2020). "nuScenes: A Multimodal Dataset for Autonomous Driving." In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11618-11628.
- [Mo et al. (2022)] Mo, J., Islam, M.J. & Sattar, J. (2022). "IMU-Assisted Learning of Single-View Rolling Shutter Correction." In *Proceedings of the 5th Conference on Robot Learning*, in *Proceedings of Machine Learning Research*, vol. 164, pp. 861-870. Available from <https://proceedings.mlr.press/v164/mo22a.html>.
- [Ok et al. (2021)] Ok, M., Ok, S., & Park, J. H. (2021). "Estimation of Vehicle Attitude, Acceleration, and Angular Velocity Using Convolutional Neural Network and Dual Extended Kalman Filter." *Sensors*, 21(4), 1282. <https://doi.org/10.3390/s21041282>
- [Liu et al. (2022)] S. Liu, H. Li, Z. Wang, J. Wang, S. Zhu and B. Zeng, "DeepOIS: Gyroscope-Guided Deep Optical Image Stabilizer Compensation," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2856-2867, May 2022, doi: 10.1109/TCSVT.2021.3103281.

Diffusion-Based Saliency Map Generation from RGB Images

Mark Langtry, Oisín O’Sullivan, Salaheddin Alakkari

*Immersive Software Engineering, Department of Computer Science and Information Systems,
University of Limerick*

Abstract

We propose SalFusion, a diffusion-based model for generating dense visual saliency maps from RGB images. The model combines a U-Net backbone with spatial self-attention and cross-attention, trained with a denoising diffusion objective and auxiliary reconstruction losses (L1, MSE, SSIM). Classifier-free guidance enables a controllable sharpness–structure trade-off during sampling, and a faster deterministic sampler is also supported. On SALICON, our results are competitive on standard saliency metrics, with particularly strong gains in AUC-Judd and NSS, while the lower CC highlights a trade-off between localization accuracy and global density matching.

Keywords: Visual saliency, diffusion models, DDPM, DDIM, classifier-free guidance

1 Introduction

Visual saliency prediction aims to estimate where humans are likely to fixate when viewing a scene. Early models relied on hand-crafted centre–surround features [1], while modern approaches use deep neural networks to learn hierarchical representations directly from fixation data, significantly improving performance on benchmark datasets. Diffusion models, particularly Denoising Diffusion Probabilistic Models (DDPMs) [2], have recently emerged as stable and expressive generative models. Trained via a simple noise-prediction objective rather than adversarial learning, diffusion models iteratively refine noisy samples into structured outputs, making them well-suited for dense prediction tasks that require both global coherence and local detail. Their success in other structured vision problems motivates their use for saliency estimation.

In this work, we formulate visual saliency prediction as a conditional diffusion process that generates a normalized single-channel saliency map from an RGB image. Our model learns the reverse denoising process of a DDPM conditioned on image features through both early fusion and cross-attention mechanisms. This iterative generative formulation enables the model to capture global scene context and produce structurally coherent saliency maps. We evaluate the approach on the SALICON and MIT1003 datasets and analyze the impact of attention mechanisms, classifier-free guidance, and sampling strategies on saliency quality.

1.1 Main Contributions

Our contributions are threefold. First, we introduce a conditional denoising diffusion framework for visual saliency estimation, modelling saliency maps as samples from a learned generative process rather than deterministic regression outputs. Second, we design a U-Net–based DDPM augmented with spatial self-attention and cross-attention, and apply classifier-free guidance during inference to produce sharper and more localized saliency predictions. Third, we benchmark the proposed model on SALICON and MIT1003 using standard saliency metrics, and we conduct ablation studies on attention modules, guidance, and sampling strategies to highlight the main performance trade-offs.

Diffusion models offer an iterative denoising formulation that aggregates multi-scale context while preserving fine detail. When conditioned on image features and combined with classifier-free guidance, they provide

an explicit mechanism to trade off sharpness and structural fidelity, which is particularly useful for saliency where precise localization of fixation peaks matters. This is a practical advantage over purely feed-forward predictors, because the sampler can be tuned to favor either sharper localization or smoother density structure. This iterative, generative viewpoint also enables sampling-based uncertainty estimates and multiple plausible saliency predictions for ambiguous scenes.

2 Related Work

2.1 CNN-Based Saliency Prediction

Early deep learning approaches to visual saliency were dominated by convolutional neural networks (CNNs). These models used hierarchical feature representation to approximate human fixation. Classical architectures such as MLNet and SalNet proved that CNN-based saliency prediction was viable. Later models improved upon these results by introducing multi-scale feature aggregation and improving spatial reasoning. Models such as SimpleNet [3], MDNSal [3], and MSI-Net [4] made further improvements using dilated convolutions, multi-branch encoders, and probabilistic modeling. Despite strong performance, CNN-based models are limited by their inherently local receptive fields, restricting their ability to model long-range dependencies.

2.2 GAN-Based Models

Following the success of generative adversarial networks (GANs) [5] in learning data distributions through adversarial training, saliency prediction has also benefited from applying this technique to saliency map prediction. SalGAN [6] uses this framework where a generator produces saliency given an image and a discriminator distinguishes generated vs GT saliency maps. The architecture consists of a VGG-16-based encoder-decoder generator paired with a discriminator trained to distinguish ground-truth saliency maps from predicted ones. The generator is optimized with a combination of a binary cross-entropy (BCE) content loss and an adversarial loss imposed by the discriminator. SalGAN was evaluated with saliency-specific metrics (CC, NSS, AUC-Judd) on SALICON and MIT benchmarks. SalGAN achieved competitive scores compared to state-of-the-art CNN-based saliency models at the time, outperforming traditional regression networks trained solely with pixel-wise losses.

2.3 Recent Work

While CNN and GAN-based approaches generate saliency maps in a single forward pass, the use of attention mechanisms enables other models to iteratively refine saliency maps over multiple timesteps. This allows the model to accumulate spatial context, correct earlier predictions and gradually refine their focus to the most important parts of the image. The Saliency Attentive Model (SAM) [7] is a family of models that adopted this technique. There are two variations of this model, SAM-VGG and SAM-ResNet, which use VGG-16 and ResNet-50 backbones respectively. SAM starts with a learned understanding of where humans usually look in images, and then uses an attention-based recurrent module to repeatedly focus on important regions, refining the saliency map step by step., leading to substantial gains over earlier CNN and GAN-based approaches. On the SALICON validation set, SAM achieves strong performance, with SAM-ResNet in particular delivering high NSS and CC scores due to its deeper feature representation and recurrent refinement process. Vision-language based approaches to saliency prediction have gained popularity, leveraging recent advances in integrating large language and vision models. Recent multimodal and transformer-based methods such as ALBEF [8], BLIP [9], ImageBind [10], and VLSal [11] show that semantic and cross-modal representations can improve saliency prediction, especially for complex real-world scenes. Our work is complementary to this line of research: we study whether an iterative generative formulation can provide competitive localization performance without relying on large pretrained multimodal backbones.

3 Method

3.1 Diffusion Models

DDPMs [2] learn to reverse a forward Gaussian noising process via noise prediction. DDIM [12] enables deterministic or fast sampling, while classifier-free guidance [13] improves conditional fidelity. Diffusion has shown strong performance in dense prediction tasks such as segmentation and depth estimation.

3.2 Saliency Prediction

Earlier saliency modelling methods relied on handcrafted features [1]. With deep learning, CNN-based regressors dominated until adversarial methods such as SalGAN [6] introduced GAN losses to encourage realistic saliency distributions.

3.3 U-Net Architectures

U-Net [14] architectures remain foundational for dense prediction. Residual connections [15], GroupNorm [16], and spatial attention modules further enhance representational power. We employ a U-Net with cross-attention to integrate image context.

3.4 Model Architecture

We adopt a conditional U-Net backbone tailored for diffusion-based dense prediction. The network takes as input a 4-channel tensor formed by concatenating the RGB image with the noisy saliency map at diffusion timestep t . A sinusoidal timestep embedding is computed and projected through a two-layer MLP to a 128-dimensional vector, which is injected into every residual block via additive conditioning. This allows the network to adapt its denoising behaviour across timesteps rather than learning a single, timestep-agnostic mapping.

The U-Net follows a symmetric encoder–decoder structure with three downsampling stages and corresponding upsampling stages. The encoder progressively increases channel capacity from 64 to 128 and 256 channels using residual blocks, followed by a bottleneck operating at 512 channels. Each stage consists of a residual convolutional block with Group Normalization and SiLU activation, followed by spatial downsampling. The decoder mirrors this structure using transposed convolutions and skip connections to fuse encoder features, enabling recovery of fine spatial detail while preserving high-level semantics. The full backbone contains approximately 11.8M trainable parameters.

To enhance global reasoning, we incorporate spatial self-attention at the bottleneck, where the feature maps have the lowest spatial resolution and highest semantic abstraction. This module uses multi-head attention to model long-range dependencies between distant regions, which is important for saliency prediction in cluttered scenes with multiple competing objects. In parallel, a lightweight RGB context encoder processes the clean image through three convolutional layers (128, 256, and 512 channels) to produce features aligned with the bottleneck resolution. These features are injected via cross-attention, enabling the denoising pathway to align saliency generation with high-level semantic cues rather than relying solely on early-layer concatenation.

The decoder fuses the attention-enhanced bottleneck representation with encoder skip features to reconstruct a full-resolution output. A final 1×1 convolution projects the last 64-channel feature map to a single channel corresponding to the predicted noise component. This prediction is used by the diffusion sampler to iteratively recover the clean saliency map. Overall, the architecture combines multi-scale feature fusion, timestep-aware conditioning, and attention-based image guidance to balance spatial precision with global contextual understanding during the reverse diffusion process. A structured summary of the network configuration is provided in Table 1.

Table 1: Summary of the conditional U-Net architecture used in the diffusion model. Resolutions are given for input size 256×192 .

Stage	Resolution	Channels	Block Type
Input	256×192	4	Residual Block
Encoder 1	128×96	64	Residual Block + MaxPool
Encoder 2	64×48	128	Residual Block + MaxPool
Encoder 3	32×24	256	Residual Block + MaxPool
Bottleneck	16×12	512	Residual Block + Self-Attn + Cross-Attn
Decoder 3	32×24	256	Transposed Conv + Residual Block (skip)
Decoder 2	64×48	128	Transposed Conv + Residual Block (skip)
Decoder 1	128×96	64	Transposed Conv + Residual Block (skip)
Output Head	256×192	1	1×1 Conv (Noise Prediction)

3.5 Loss Function

Training uses a multi-component objective that combines the standard DDPM noise-prediction loss with auxiliary reconstruction and perceptual structure terms. The auxiliary terms provide a more direct supervised signal on the predicted clean saliency map and help reduce over-smoothing, which can otherwise blur fixation peaks. For clarity, all losses are computed on saliency maps normalized to $[0, 1]$ and weighted to operate on comparable scales. The overall objective is:

$$\mathcal{L}_{\text{total}} = \lambda_{\varepsilon} \mathcal{L}_{\varepsilon} + \lambda_1 \mathcal{L}_1 + \lambda_2 \mathcal{L}_2 + \lambda_s \mathcal{L}_{\text{SSIM}}, \quad (1)$$

where we set $\lambda_{\varepsilon} = 1.0$, $\lambda_1 = 0.5$, $\lambda_2 = 0.5$, and $\lambda_s = 0.5$.

The noise-prediction term follows the DDPM objective. We sample a timestep $t \sim \mathcal{U}\{1, \dots, T\}$ and construct a noisy target saliency map $\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon$, where $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\bar{\alpha}_t$ is the cumulative product of the noise schedule. The model predicts the injected noise conditioned on the RGB input $\mathbf{c}$, and we minimize:

$$\mathcal{L}_{\varepsilon} = \mathbb{E}_{\mathbf{x}_0, t, \varepsilon} \|\varepsilon - \varepsilon_{\theta}(\mathbf{x}_t, t, \mathbf{c})\|_2^2. \quad (2)$$

This term is the core objective enabling accurate reverse diffusion.

To encourage spatial fidelity and well-calibrated intensities, we additionally supervise the estimated clean saliency map $\hat{\mathbf{x}}_0$ with pixel-wise losses:

$$\mathcal{L}_1 = \|\hat{\mathbf{x}}_0 - \mathbf{x}_0\|_1, \quad \mathcal{L}_2 = \|\hat{\mathbf{x}}_0 - \mathbf{x}_0\|_2^2. \quad (3)$$

The L1 term promotes sharper peaks and reduces blur, while the L2 term provides smooth gradients and stabilizes optimization by penalizing large deviations more strongly.

Because saliency maps are smooth density fields where perceptual structure matters, we also incorporate a lightweight SSIM-based loss:

$$\mathcal{L}_{\text{SSIM}} = 1 - \text{SSIM}(\hat{\mathbf{x}}_0, \mathbf{x}_0), \quad (4)$$

which encourages agreement in local luminance, contrast, and spatial structure beyond pixel-wise matching. In combination, these terms balance accurate denoising with structural fidelity, improving both localization of fixation peaks and global map coherence.

3.6 Training and Inference

Training uses AdamW optimizer with learning rate $1e-4$, batch size 8, timesteps 500, and cosine annealing noise scheduling over 60 epochs. Training took ~ 3 hours on a NVIDIA RTX A4500 GPU. Mixed precision (AMP) accelerates training.

Compute: Training on SALICON with batch size 8 required approximately 3 hours on a single NVIDIA RTX A4500. For inference we report timings for both full DDPM sampling and a faster DDIM sampler (50

steps) as a practical trade-off: DDIM reduces sampling cost substantially while retaining most accuracy. Practical deployment options to reduce cost include timestep reduction, denoising distillation, or a lightweight deterministic refinement head.

At inference, we use DDPM sampling for best quality, as the full stochastic reverse process better matches the training objective and yields more refined spatial structure. We optionally employ DDIM for $\sim 10\times$ faster sampling due to the reduced number of denoising steps, at the cost of some loss in fine detail and localization accuracy. Classifier-free guidance with scale $s = 1.5$ is applied to strengthen conditioning on the RGB image. By amplifying the difference between conditional and unconditional predictions, guidance encourages sharper activations around likely fixation regions and improves fixation localization, though overly large guidance scales were observed to produce overconfident or overly peaked saliency maps.

4 Results

4.1 Dataset and Preprocessing

We train on the SALICON dataset [17], which provides MS-COCO images with mouse-tracking fixation data. The dataset contains 10k training images, 5k validation images, and 5k test images, with no fixation annotations available for the test split.

Evaluation protocol: our outputs are resized to 256×192 and we apply Gaussian smoothing ($\sigma = 8$) prior to metric computation. For the comparison table, prior-method values are reported from their published results, while SalFusion entries are our own results computed using this evaluation protocol. We report single-run results in this submission, so no confidence intervals are included.

Fixation JSONs are converted into continuous saliency maps using Gaussian smoothing ($\sigma = 8$), normalized into $[0, 1]$, and resized to 256×192 .

4.2 Evaluation Metrics

We evaluate model performance using three standard saliency prediction metrics computed on the validation set. CC measures the linear correlation between the predicted saliency map and the ground-truth human saliency map, NSS evaluates the average normalized saliency value at fixation points, and AUC-Judd measures how well the model ranks human fixations above non-fixated pixels across the image.

4.3 Quantitative Results

Table 2 presents the quantitative comparison between our model, SalFusion, and other models on the SALICON (validation) and MIT1003 datasets. The values for prior methods are reported in [11], while the SalFusion entries are our own results under the evaluation protocol described above.

Based on these results, our diffusion model is stronger on the AUC and NSS metrics, but weaker on CC. This reflects a trade-off between discriminative ranking performance and pixelwise structural fidelity. The large gains in AUC-Judd and NSS indicate that the model is more effective at ranking salient versus non-salient regions and at placing high values near fixation locations, while the lower CC suggests that the predicted maps are less linearly aligned with the full smoothed density structure of the ground truth. Overall, our diffusion approach produces saliency maps that are more effective for localization and detection-style evaluation but less accurate in reproducing the full spatial density structure captured by CC.

Limitations and calibration: The lower CC scores reflect reduced per-pixel linear agreement with smoothed ground-truth densities, caused by the model’s tendency to produce sharper, peakier maps. This is a practical limitation of the current formulation, particularly for applications that require faithful global density matching rather than strong fixation localization. We find that light post-hoc Gaussian smoothing or a small learned intensity-calibration stage can substantially increase CC with only minor reductions in NSS/AUC. Diffusion models are also more computationally expensive at inference than single-pass feed-forward baselines, so the

Table 2: Quantitative comparison of models trained on SALICON train and evaluated on SALICON validation and MIT1003. Prior-method values are reported in [11]; SalFusion entries are our own results.

SALICON (Validation)				MIT1003			
Model	NSS↑	CC↑	AUC-J↑	Model	NSS↑	CC↑	AUC-J↑
SALICON	1.5003	0.7486	0.8304	SALICON	1.6648	0.5012	0.8380
MLNet	1.5021	0.6746	0.8094	MLNet	1.7424	0.4836	0.8241
SalGAN	1.7989	0.8601	0.8601	SalGAN	2.1328	0.6245	0.8779
SAM-VGG	1.7637	0.8247	0.8524	SAM-VGG	2.0078	0.5788	0.8539
SAM-ResNet	1.8121	0.8398	0.8543	SAM-ResNet	2.1320	0.6153	0.8694
TranSalNet	1.7996	0.8325	0.8496	TranSalNet	1.7829	0.5520	0.8693
TempSAL	1.7656	0.8245	0.8395	TempSAL	1.7650	0.5203	0.8497
GSGNet	1.8339	0.8466	0.8548	GSGNet	1.9846	0.5795	0.8656
TGSal	1.8606	0.8816	0.8658	TGSal	2.2856	0.6632	0.8873
ALBEF	1.7368	0.8469	0.8564	ALBEF	1.9971	0.5958	0.8743
BLIP	1.8070	0.8649	0.8603	BLIP	2.0964	0.6172	0.8768
ImageBind	1.7248	0.8423	0.8558	ImageBind	1.9851	0.5925	0.8728
VLSal	1.9492	0.9085	0.8718	VLSal	2.4658	0.7088	0.8986
SalFusion	2.6650	0.7245	0.9291	SalFusion	2.5327	0.4855	0.8925

current work prioritizes quality and controllability over sampling efficiency. We report these trade-offs explicitly and discuss them further in the supplementary material.

4.4 Qualitative Results

Fig. 1 shows that model captures main fixation regions, but predicts sharper peaks and occasionally misses small secondary regions. It appears that our model favours more confident predictions over matching the per-pixel intensity of the ground-truth saliency maps.



Figure 1: Representative qualitative examples. Each row shows an RGB image, the ground-truth saliency map, and the predicted saliency map from our model.

4.5 Ablation Study

We conduct an ablation study to assess the impact of toggling features on / off on model performance. The results of our study captured in Table 3 suggest that guidance is the most influential factor, while DDIM harms accuracy in our setup.

Table 3: Ablation Results (CFG = classifier-free guidance).

CFG	DDIM	SA	CA	AUC-J	NSS	CC
1.5	Off	T	T	0.93	2.62	0.7185
0.0	Off	T	T	0.91	2.257	0.5895
1.5	On	T	T	0.86	1.58	0.42
1.5	Off	F	F	0.89	2.42	0.639

5 Conclusion

Our research and experiments show that diffusion models show competitive performance relative to GANs for saliency prediction. Despite a slight underperformance in CC, our diffusion model outperformed other models in other key areas. The stronger CC score of GAN-based approaches is due to its adversarial training, which encourages generators to match the global distribution of saliency maps. Diffusion models prioritise producing high-confidence predictions at likely fixation points, which boosts metrics such as AUC and NSS. This behaviour is consistent with our qualitative analysis, where saliency concentrated sharply around high-confidence regions.

Beyond GANs, our results also demonstrate that diffusion models compare favourably to traditional CNN-based saliency networks such as SimpleNet and MDNSal, achieving higher AUC and NSS while offering more stable training and better spatial coherence. Compared to recurrent attention models like SAM-VGG and SAM-ResNet, diffusion models avoid the complexity and computational overhead of multi-step refinement while still delivering competitive localisation accuracy. SAM achieves strong NSS through iterative ConvLSTM refinement, but diffusion models provide a simpler, more general framework that excels at precise fixation localisation.

Overall, whether a diffusion model or a GAN is more suitable depends on the application. Diffusion models may be more reliable in scenarios where accurate, sharply localised fixation prediction is essential, whereas GAN-based approaches may be preferable when matching global saliency statistics is the primary objective. For saliency prediction, the main advantage of diffusion is the ability to model structured spatial outputs while providing an explicit control over sharpness and structural fidelity through guidance.

6 Acknowledgments

We thank the faculty and staff of Immersive Software Engineering in the University of Limerick for their support, in particular Salaheddin Alakkari for his guidance on this paper. We thank the anonymous reviewers for their constructive feedback and the authors of the cited methods for making their work publicly available.

References

- [1] Laurent Itti, Christof Koch, and Ernst Niebur, “A model of saliency-based visual attention for rapid scene analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.
- [2] Jonathan Ho, Ajay Jain, and Pieter Abbeel, “Denoising Diffusion Probabilistic Models,” Dec. 2020.
- [3] Navyasri Reddy, Samyak Jain, Pradeep Yarlagadda, and Vineet Gandhi, “Tidying Deep Saliency Prediction Architectures,” Mar. 2020.
- [4] Alexander Kroner, Mario Senden, Kurt Driessens, and Rainer Goebel, “Contextual Encoder-Decoder Network for Visual Saliency Prediction,” *Neural Networks*, vol. 129, pp. 261–270, Sept. 2020.
- [5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, “Generative Adversarial Networks,” June 2014.
- [6] Junting Pan, Cristian Canton Ferrer, Kevin McGuinness, Noel E. O’Connor, Jordi Torres, Elisa Sayrol, and Xavier Giro-i-Nieto, “SalGAN: Visual Saliency Prediction with Generative Adversarial Networks,” July 2018.
- [7] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara, “Predicting Human Eye Fixations via an LSTM-based Saliency Attentive Model,” *IEEE Transactions on Image Processing*, vol. 27, no. 10, pp. 5142–5154, Oct. 2018.
- [8] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi, “Align before fuse: Vision and language representation learning with momentum distillation,” in *Advances in Neural Information Processing Systems*, 2021.
- [9] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi, “Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation,” in *International Conference on Machine Learning*, 2022.
- [10] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Caiming Xiong, Ishan Misra, and Armand Joulin, “Imagebind: One embedding space to bind them all,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.
- [11] Anonymous, “Exploring the potential of vision-language models for pure-image and text-guided image saliency prediction,” IEEE Xplore, 2024, Preprint; available at <https://ieeexplore.ieee.org/document/11084645>.
- [12] Jiaming Song, Chenlin Meng, and Stefano Ermon, “Denoising Diffusion Implicit Models,” Oct. 2022.
- [13] Jonathan Ho and Tim Salimans, “Classifier-Free Diffusion Guidance,” July 2022.
- [14] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” May 2015.
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep Residual Learning for Image Recognition,” Dec. 2015.
- [16] Yuxin Wu and Kaiming He, “Group Normalization,” June 2018.
- [17] SALICON, “Salicon: Saliency in context,” <https://www.salicon.net/>, 2015, Accessed: 2026-04-28.

Thumbnail Guided Tissue Compaction (TGTC) - Whole Slide Image Preprocessing for Lossless Image Compression

Christopher Smith, Leo Galway, Richard Gault

Centre for Intelligent Sustainable Computing, School of Electronics, Electrical Engineering, and Computer Science, Queen's University, Belfast

Abstract

Whole-slide images (WSI's) of histopathology slides are difficult to store and transmit because scans routinely occupy multiple gigabytes before compression. While some details of the image must never be lost due to their clinical significance, there are frequently background details that are less relevant. This paper proposes Thumbnail-Guided Tissue Compaction (TGTC) that detects tissue on a low-resolution thumbnail of the sample before a crop-and-rotate calculation is applied. This acts as a preprocessing stage to existing image compression algorithms aiming to optimally position the sample to minimise background information. TGTC isolates and extracts the important tissue details and reduces the amount of redundant background information being stored in the compressed file. This study considers 12 WSI's captured from a diverse range of pathologies. On average, TGTC reduces the storage constraints by 21.35% and up to 69.91% in some cases. A thumbnail-level retention audit evaluated the efficacy of identifying tissue with such a low-resolution image and found a mean retained tissue fraction 99.94%. Across 48 paired slide-method comparisons, TGTC reduces final compressed size in 40/48 cases, with mean improvement 10.39% and mean codec-ratio delta of 0.71. These results support TGTC as a practical WSI preprocessing stage when the goal is reduced final storage with an explicit tissue-preservation audit trail.

Keywords: whole-slide imaging, lossless compression, digital pathology, preprocessing, tissue detection.

1 Introduction

Whole-slide imaging is central to digital pathology, but its storage demands remain a practical barrier for archival use, dataset sharing, and large-scale computational pathology studies. Studies found that the average storage cost for digital pathology labs across seven sites in the European Union is €155,000 per site annually [1]. A level-0 RGB representation of a single slide can occupy several gigabytes before compression, and benchmarking many such slides across multiple codecs quickly becomes expensive in both runtime and disk use. This work studies a simple idea: rather than compressing the full slide canvas, first identify the tissue footprint from a thumbnail, estimate a tight rotated bounding region, and then compress only the resulting tissue-focused crop. The proposed Thumbnail-Guided Tissue Compaction (TGTC) pipeline is designed to answer two questions at once. First, does TGTC reduce the number of bytes that must ultimately be stored? Second, can that reduction be defended as effectively lossless with respect to tissue retention?

The answer is not obvious. A smaller image does not automatically imply a better codec ratio (or compression ratio) because the transformed image may be less compressible per byte and the distinction between these two goals is important. The downstream codec stage can be tested for strict bitwise losslessness tile by tile. The TGTC preprocessing stage cannot make the same claim in original slide coordinates because it changes orientation and canvas extent. Instead, it should be assessed through an explicit tissue-retention audit. This paper therefore separates raw size reduction, codec ratio, final compressed size, timing cost, and tissue-retention evidence rather than treating compression ratio alone as the sole objective.

The main contributions of the study are:

1. A novel pre-processing stage to extend existing lossless compression methods that reduces redundant high-fidelity compression seen in existing methods.
2. A reviewer-facing definition of tissue-preservation based on retention-audit evidence rather than pixel-wise preservation after crop-and-rotate preprocessing, as rotation inherently changes pixel coordinates.
3. A storage-centric analysis showing that TGTC often reduces final compressed size even if the transformed image exhibits a lower codec ratio than the baseline canvas.

2 Literature Review

WSI's can reach gigapixel scales meaning efficient compression is hugely important for the practical utilisation of these data. The literature shows a clear progression from general-purpose algorithms to specialised, high-performance tools for WSI compression. Welch [2] detailed the LZW (Lempel-Ziv-Welch) algorithm that became a standard for lossless data compression (notably in GIF and TIFF formats). While reliable, LZW often struggles with the high-entropy demands of modern high-resolution imaging. Collet and Kuchera [3] standardised the Zstandard (Zstd) algorithm in 2021. Zstd offers a balanced performance profile, delivering compression ratios comparable to traditional dictionary-based methods like DEFLATE and zlib, alongside significantly higher decompression speeds suitable for real-time data streaming in digital systems. Alakuijala et al. [4] introduced JPEG-XL, designed specifically for the requirements of high-fidelity images. It offers superior compression efficiency and features - such as lossless transcoding of legacy JPEGs and responsive web delivery - that are particularly relevant for the long-term storage and remote viewing of medical slides.

Mao et al. [5] introduced the WISE framework, a classical, non-learned approach to lossless compression of gigapixel whole-slide images. WISE combines hierarchical predictive transforms, bitplane decomposition, and dictionary-based entropy coding, motivated by the observation that histopathology slides exhibit significant spatial redundancy and background sparsity. Operating on tiled representations, the framework is designed to scale to gigapixel data without requiring full-resolution processing. More broadly, WISE highlights the importance of content-aware redundancy in gigapixel imagery, motivating the exploration of adaptive, learned compression approaches considered in this study.

While conceptually grounded in established lossless compression techniques, WISE is presented at a high level and does not specify several implementation-critical details, such as residual representation, bitplane encoding, and entropy coding constraints. In the absence of an official reference implementation, exact reproducibility is therefore not guaranteed. In this study, a faithful implementation approximation of WISE is included based on the available description and its use as one of several comparative baselines. Nevertheless, the framework provides a useful reference point for assessing redundancy reduction strategies in WSI's.

3 Experimentation

3.1 Data

A selection of 12 whole slide images were extracted from The Cancer Genome Atlas (TCGA) [6] in their original .svs proprietary file format to represent 12 different potential cancerous sites of the human body; specifically adrenal gland, bladder, brain, breast, cervix, colon, heart, kidney, lung, ovary, pancreas, and skin. These sites were chosen to exemplify the wide variability of background content found in different histopathologies.

3.2 Thumbnail-Guided Tissue Compaction (TGTC)

The proposed pre-processing block, TGTC, comprises of 3 core steps:

1. foreground/background separation with Otsu Thresholding [7],
2. the isolation of biological tissue within the image, and

3. rotation & cropping to produce a minimal tissue region along the angle of inclination, where the images sides run along the x- and y-axis.

Each source WSI `.svs` file is opened with OpenSlide and read at level 0. The baseline export streams RGB tiles into memory-mapped raw unsigned 8-bit integers (`uint8`). The resulting rectangle is mapped back into full-resolution coordinates. A perspective transform is then used to stream the rotated crop, tile by tile, into a new `.dat` file where it is then saved. Each slide has a baseline memory-mapped RGB `.dat` representation stored in `uint8` and a TGTC-transformed memory-mapped RGB `.dat` representation created from the original `.svs` file. This tiled extraction avoids creating a massive intermediate rotated image in memory and makes the method feasible on large WSI canvases.

3.3 Compression Methods

Four lossless compression methods are benchmarked based on the review of the existing literature, namely:

1. **LZW**: dictionary-based lossless compression in the spirit of the classic LZW formulation [2].
2. **Zstandard**: byte-oriented entropy coding using Zstandard at compression level 3, the default setting [3].
3. **JPEG-XL (lossless)**: lossless still-image compression via JPEG-XL [4].
4. **WISE (Custom)**: a custom three-stage pipeline consisting of horizontal differencing, bit-plane reordering, and a final Zstandard compression stage, based off a framework from Mao et al. [5].

Due to the sheer size of the WSI cached for compression and the strain it will put on resources, the whole file is instead tiled into smaller images of size 1024×1024 . Each tile is compressed, decompressed, and then compared with the original tile by exact array equality before the result is accepted.

3.4 Metrics

To evaluate the performance of the algorithms (with and without TGTC), a range of metrics are considered to assess absolute size reduction, relative compression rate, and whether the method is lossless for tissue information. This study uses five metric families:

1. **Raw TGTC size reduction**

$$R_{\text{raw}} = \left(1 - \frac{S_{\text{TGTC}}}{S}\right) \times 100 \quad (1)$$

where S is the uncompressed `.dat` size and S_{TGTC} is the file size after TGTC pre-processing is applied to the corresponding baseline algorithm.

2. **Codec ratio on a given representation**

$$\rho_c(X) = \frac{X}{C_c(X)} \quad (2)$$

where $C_c(X)$ is the compressed size produced by codec c on data representation X of a given image.

3. **Final compressed-size reduction for a matched pair**

$$R_{\text{final},c} = \left(1 - \frac{C_c(X_{\text{TGTC}})}{C_c(X)}\right) \times 100 \quad (3)$$

where X_{TGTC} is the data representation of the pre-processed image with TGTC. This is the most important storage-facing metric because it directly compares the bytes that would be stored after compression.

4. **Tissue-retention metrics:** The fractions of retained tissue, discarded tissue, and removed background are captured to assess how much redundancy is in the image and how lossless the resultant compression actually is.
5. **Runtime metrics:** The time taken to compress, decompress, and the throughput (MB/s) are evaluated to help appreciate the impacts on real-world usage of any technique.

It is important to note the distinction between $p_c(X)$ and $R_{\text{final},c}$ is central to interpreting the results. A transformed image can be less compressible per byte than the original canvas while still yielding fewer final bytes overall because TGTC has already removed a substantial fraction of the canvas.

3.5 Tissue Retention Audit

In the absence of available pathologist-annotated ground truth masks for the slides used in this experiment, it was decided to utilise a classical thresholding method in the pipeline and audit it with a newer state of the art (SOTA) method. Ceachi et al (2025) introduced Double-Pass, an innovative unsupervised and annotation-free hybrid method for tissue detection for WSI's [8]. This approach fuses two complementary strategies: *FilterGrays*, which utilises sharpening and colour-based artifact rejection, and *DownKMeans*, which performs clustering on downsampled RGB thumbnails to maximise speed. Double Pass will be used to detect tissue loss across all slides in the cohort to ensure robustness.

The audit file uses the same parameters as the authors original paper [8], while the threshold assigned to determine a PASS in the audit is set at 0.5% for the fraction of discarded tissue over all tissue on the whole slide outside of the crop footprint, the largest component fraction, or the largest discarded tissue fragment.

The full benchmark was executed on Kelvin2 (NI-HPC), using an NVidia A100 GPU. The benchmarking used compared all algorithms (with and without TGTC) against the four compression codecs and the tiled lossless verification path.

4 Results

4.1 TGTC Reduces Raw WSI Bytes Across the Full 12-Slide Cohort

Table 1 provides an overview of the size of the files before (baselines) and after the TGTC pre-processing of the image. In particular, the % reduction provides an overview of the amount of redundant background information that can be discarded through tissue identification, cropping, and rotation. Across the 12 paired slides, TGTC reduces aggregate raw storage from 145.40 GB to 114.36 GB, corresponding to an overall reduction of 21.35%. On a per-slide basis, the mean raw reduction is 29.81% and the median is 27.33%. The largest reduction is observed for brain (69.91%), while ovary shows the smallest but still positive raw reduction (6.48%). This means TGTC reduces raw storage burden on every slide in the current cohort before any codec is applied. Figure 1 illustrates each stage of the TGTC process including the final rotation, shown at Fig. 1(d), to orientate the image before saving in full resolution.

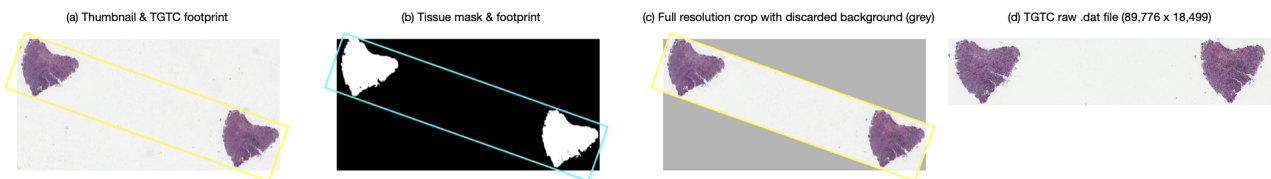


Figure 1: Representative TGTC process overlay for the lung slide. (a) The crop footprint is overlaid on the thumbnail and (b) tissue mask, (c) scaled to full resolution with discarded background shown in grey, and (d) the saved tissue is reserved after rotation.

Table 1: Paired baseline and TGTC .dat size and Raw Size Reduction (1) for the 12-slide HPC cohort.

Slide	Baseline (GB)	TGTC (GB)	Baseline Size	After TGTC	Size Reduction (%) (1)
Adrenal Gland	20.04	18.09	79,679 x 83,835	74,859 x 80,563	9.72
Bladder	21.00	15.39	10,1591 x 68,905	90,349 x 56,763	26.74
Brain	1.02	0.31	12,000 x 28,417	4,340 x 23,643	69.91
Breast	29.46	24.67	124,712 x 78,731	115,881 x 70,968	16.24
Cervix	2.34	1.92	24,940 x 31,295	20,877 x 30,717	17.84
Colon	10.61	6.04	55,040 x 64,256	43,432 x 46,372	43.05
Heart	4.61	3.08	32,671 x 47,076	21,036 x 48,856	33.18
Kidney	7.93	6.88	48,921 x 54,063	34,689 x 66,154	13.23
Lung	10.68	4.98	84,413 x 42,180	89,776 x 18,499	53.36
Ovary	28.79	26.93	109,559 x 87,606	107,792 x 83,270	6.48
Pancreas	5.97	4.30	49,880 x 39,867	33,348 x 42,986	27.91
Skin	2.94	1.76	27,888 x 35,169	21,925 x 26,790	40.11

4.2 Double-Pass Retention Audit find 99.94% Tissue-Preservation

While the double-pass retention audit shows there is some loss within the TGTC method, Table 2 shows all 12 audited slides PASS the audit and are safely inside the thresholds set, with mean retained tissue fraction 99.94%. The mean thumbnail-level removed area is 34.51% and the median is 35.31%, with only a very small fraction of this classified as discarded tissue and the vast majority being discarded background (See Figure 2). Despite positive results, the current audit cannot support a tissue-preserving losslessness claim for TGTC, however, it does provides strong evidence that TGTC achieves substantial data reduction while maintaining exceptionally high tissue retention, supporting its suitability for practical digital pathology workflows. Note: the ‘Area Removed (%)’ column from Table 2 and the ‘Size Reduction (%)’ column from Table 1 differ because while the former counts only discarded background pixels, excluding those pixels within the minimum bounding rectangle that lie outside of the original image, the latter measures the total pixel difference between the original and resized images.

Table 2: TGTC tissue-retention audit summary for the 12-slide HPC cohort.

Slide	Area Removed (%)	Retained Tissue (%)	Status
Adrenal Gland	10.83	99.62	PASS
Bladder	26.67	100.00	PASS
Brain	69.92	100.00	PASS
Breast	15.81	99.91	PASS
Cervix	26.62	100.00	PASS
Colon	46.69	100.00	PASS
Heart	38.19	100.00	PASS
Kidney	35.24	100.00	PASS
Lung	54.93	99.93	PASS
Ovary	11.19	100.00	PASS
Pancreas	35.37	99.81	PASS
Skin	42.61	100.00	PASS

4.3 Final Compressed Size Usually Improves Even With Less Redundant Data to Compress

When considering each compression method in combination and without the TGTC preprocessing block, we find that for each method, TGTC reduces the final compressed size in 40/48 cases (seen in Table 3), with mean improvement 10.39% and median improvement 6.55%. This is also strengthened with a small upward trend observed in the compression ratios, shown in Figure 3. Much of that work is to do with the removal of redundant background data during the preprocessing stage, as we often know that a transformed image contains fewer total bytes, but its remaining pixels are tissue-dense and therefore often less redundant than a large tissue-plus-background canvas for compression. Despite this, 40/48 files showed a positive improvement when TGTC is applied before compression. From a storage perspective, the final compressed size of a cohort matters more than individual slide representation where very small increases in size are negligible over full datasets. This is

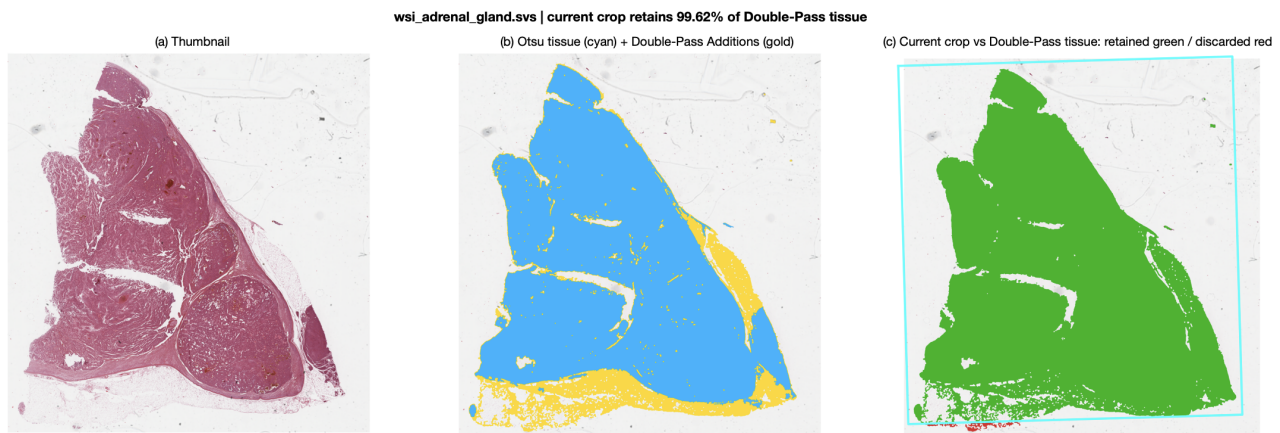


Figure 2: Double-Pass Audit for the Adrenal Gland slide - shows (a) original thumbnail, (b) Otsu's tissue detection (in blue) with additional tissue detected by Double-Pass shown in gold and (c) tissue that was erroneously cropped by TGTC shown in red.

Table 3: Method-level summary over the 12-slide paired benchmark. Mean final reduction (%) is the average of the final compressed-size reduction for a matched pair ($R_{\text{final},c}$) for the same slide and codec. (3)

Method	Wins	Baseline Median Ratio ($\uparrow$)	TGTC Median Ratio ($\uparrow$)	Mean Final Reduction (%) ($\uparrow$)	Baseline Median Comp. Time (s $\downarrow$)	TGTC Median Comp. Time (s $\downarrow$)
JPEG-XL (tiled)	12/12	4.65	4.96	10.65	1629.41	1339.83
LZW (tiled)	12/12	1.79	1.90	11.39	82.35	53.51
WISE (Custom) (tiled)	10/12	3.16	3.31	10.82	63.33	40.55
ZStandard (tiled)	6/12	2.41	2.50	8.70	24.70	15.30

in addition to the storage savings already achieved on file size when preprocessing is applied. The results show that this storage-centric view is the right one for TGTC.

Method-level behaviour is also informative. LZW and JPEG-XL improve final compressed size on all 12 slides. WISE improves on 10 of 12 slides, and Zstandard improves on 6 of 12 slides. Zstandard nonetheless remains the fastest codec family in terms of compression throughput, illustrating a clear compression-ratio-versus-runtime trade-off.

4.4 Lost Tissue Data are Small and Localised

Table 4: Cases where TGTC increased final compressed size slightly.

Slide	Method	Final Reduction (%)
Ovary	ZStandard (tiled)	-2.78
Skin	ZStandard (tiled)	-2.05
Adrenal Gland	ZStandard (tiled)	-0.87
Ovary	WISE (Custom) (tiled)	-0.79
Colon	ZStandard (tiled)	-0.58
Skin	WISE (Custom) (tiled)	-0.45
Bladder	ZStandard (tiled)	-0.44
Breast	ZStandard (tiled)	-0.17

There are eight negative paired comparisons in the full benchmark (Table 4), all relatively small in magnitude. Six occur with Zstandard and two with WISE. The nature of the Zstandard and WISE algorithms may mean that there is an optimal orientation of the image to best utilise the style of compression which does not necessarily mean reducing the volume of information to be compressed. The worst compression is seen in the ovary sample using Zstandard where final compressed size increases by 2.78%. These negative results are important to highlight as they show that TGTC is not uniformly beneficial for every slide-codec combination.

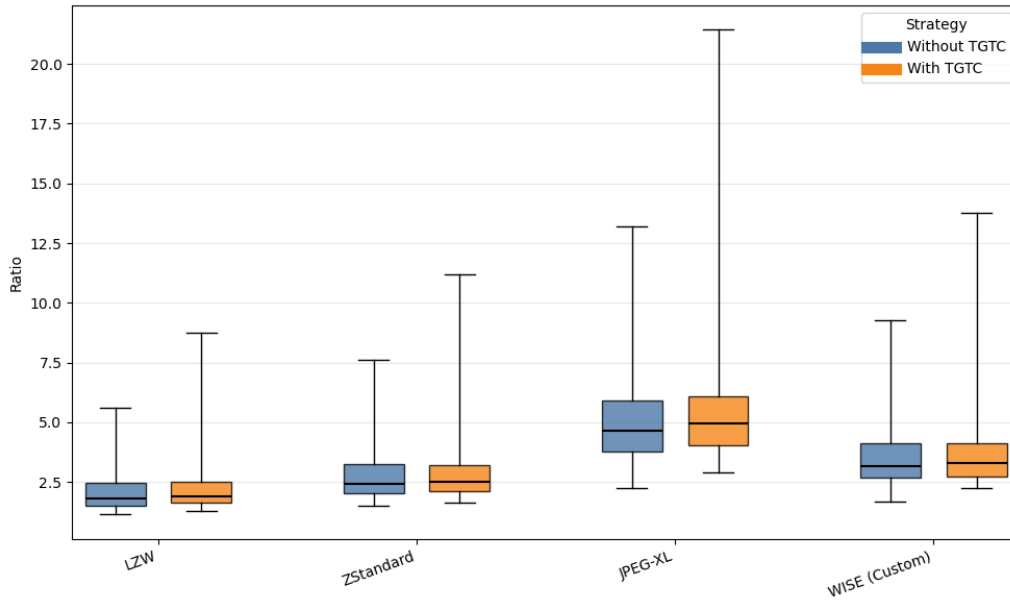


Figure 3: Box-and-whisker comparison of codec ratios for baseline and TGTC-transformed inputs across the 12-slide cohort in tiled format. TGTC generally shifts ratio trend upwards to improve final compressed size overall, with marginal increases in the median ratio shown in every compression method and similar patterns within the interquartile range with slight upward trends.

At the same time, the overall pattern remains favourable: all raw pairs shrink, all slides pass the retention audit, and most final compressed results improve.

4.5 Runtime Explains the Practical Limit on Dataset Size

The HPC benchmark clarifies why WSI compression studies are costly to scale. Median baseline compression times are 24.7s for Zstandard, 82.4s for LZW, 63.3s for WISE, and 1629.4s for JPEG-XL. On TGTC inputs, the corresponding medians fall respectively to 15.3s, 53.5s, 40.6s, and 1339.8s for each of the compression methods. Median baseline compression throughput is 464.5MB/s for Zstandard, 143.3MB/s for WISE, 108.4MB/s for LZW, and 4.3MB/s for JPEG-XL.

These numbers make the practical trade-off visible. JPEG-XL gives the highest compression ratios, but it is far slower than the other methods. Zstandard is consistently the fastest, but its storage benefit after TGTC is less reliable than LZW or JPEG-XL. WISE occupies a middle region: it is much faster than JPEG-XL while typically retaining a larger storage gain than Zstandard.

5 Discussion

This study provides encouraging evidence that TGTC can reduce whole-slide image storage requirements while maintaining a high level of tissue preservation. Four key findings emerge from the evaluation. First, **TGTC consistently reduced raw background burden** across the dataset, with every slide demonstrating a positive reduction in uncropped file size and an overall storage reduction of 21.35%. Second, **tissue-retention analysis showed a high degree of preservation**, with all 12 slides passing the SOTA Double-Pass tissue audit and achieving a mean retained tissue fraction of 99.94%. Third, **the effectiveness of TGTC is influenced by the choice of compression codec**. Storage improvements were observed in 40/48 paired comparisons across four codecs, with LZW and JPEG-XL producing file-size reductions for every slide in the cohort, although the magnitude of benefit remained codec dependent. Finally, **computational cost remains a practical consideration**. Even with serial HPC execution, benchmarking a 12-slide cohort across four codecs required substantial compute resources, particularly for JPEG-XL, which may limit larger-scale evaluations.

Despite these promising findings, several limitations should be acknowledged. Tissue-preservation was assessed using thumbnail-level audit masks rather than full-resolution pathological review, and future studies should incorporate pathologist validation to establish clinical relevance. While TGTC generally improved storage efficiency, results remain dependent on codec selection, compression settings, and tile configuration. The Double-Pass Tissue Audit also revealed minor loss of faint tissue structures along some tissue boundaries following cropping and rotation. Although retention remained well within acceptable thresholds, clinical significance of these losses are unknown and warrants further investigation. Refinement of the Otsu thresholding stage may help improve tissue identification and reduce the risk of inadvertent tissue exclusion. Finally, conclusions are based on a relatively small cohort of 12 WSI's. Future work should evaluate TGTC on larger, more diverse datasets to assess the generalisability of observed storage and tissue-retention performance.

6 Conclusion

This study demonstrates that Thumbnail-Guided Tissue Compaction (TGTC) is a practical preprocessing stage for WSI storage pipelines. As shown in Section 4.1, TGTC reduced aggregate raw storage requirements from 145.40GB to 114.36GB across the 12-slide cohort. In digital pathology environments, such reductions could lower storage costs, reduce transmission burdens, and improve the scalability of growing slide archives.

The compression benchmarking results in Section 4.3 show that TGTC improved the final compressed file size in 40 of 48 paired slide-codec comparisons, indicating that TGTC can complement existing compression approaches without requiring changes to downstream workflows. Tissue-retention analysis (Section 4.2) further showed that all audited slides passed the Double-Pass Tissue Audit, with a mean retained tissue fraction of 99.94%, suggesting that meaningful storage savings can be achieved while preserving tissue content under the current audit methodology. Among the evaluated codecs, LZW and JPEG-XL produced the most consistent benefits when combined with TGTC. JPEG-XL achieved the strongest compression performance, making it attractive for archival storage, while LZW offered a more favourable balance between storage reduction and processing time for active use.

Overall, the results suggest that TGTC offers a promising route towards reducing the growing storage burden of digital pathology while maintaining high levels of tissue retention. Future work should validate tissue preservation using higher-resolution and task-based assessments and evaluate performance across larger and more diverse datasets.

References

- [1] X. Matias-Guiu, J. Temprana-Salvador, P. G. Lopez, S. F. Kammerer-Jacquet, N. Rioux-Leclercq, D. Clark, C. M. Schürch, F. Fend, S. Mattern, D. Snead, N. Fusco, E. Guerini-Rocco, F. Rojo, M. Brevet, M. S. Tellez, A. D. Tos, T. di Maio, S. Ramírez-Peinado, E. Sheppard, H. Bannister, A. Gkiokas, M. Arpaia, O. B. Dhia, and N. Martino, "Implementing digital pathology: qualitative and financial insights from eight leading european laboratories," *Virchows Archiv*, 2025.
- [2] T. A. Welch, "A technique for high-performance data compression," *Computer*, vol. 17, no. 6, pp. 8–19, 1984.
- [3] Y. Collet and M. Kucherawy, "RFC 8878: Zstandard Compression and the 'application/zstd' Media Type," IETF, Tech. Rep., 2021. [Online]. Available: <https://www.rfc-editor.org/info/rfc8878>
- [4] J. Alakuijala, R. van Asseldonk, S. Boukott, M. Bruse, I.-M. Comsa, M. Firsching, T. Fischbacher, S. Gomez, E. Kliuchnikov, R. Obryk, K. Potempa, A. Rhatushnyak, J. Sneyers, Z. Szabadka, L. Vandevenne, L. Versari, and J. Wassenberg, "JPEG XL next-generation image compression architecture and coding tools," in *Applications of Digital Image Processing XLIII*. SPIE, 2019. [Online]. Available: <https://research.google/pubs/jpeg-xl-next-generation-image-compression-architecture-and-coding-tools/>
- [5] Y. Mao, J. Wang, N. Guan, and C. J. Xue, "Wise: A framework for gigapixel whole-slide-image lossless compression," 2025. [Online]. Available: <https://arxiv.org/abs/2503.18074>
- [6] N. C. I. USA, "<https://www.cancer.gov/tcga>."
- [7] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [8] B. Ceachi, F. Muresan, M. Trascau, and A. M. Florea, "Efficient tissue detection in whole-slide images using classical and hybrid methods: Benchmark on tcga cancer cohorts," *Cancers*, vol. 17, no. 17, 2025. [Online]. Available: <https://www.mdpi.com/2072-6694/17/17/2918>

Body Mass Index (BMI) estimation from a single frontal face image using a Bayesian-ViT architecture

Carlos Naranjo, Francisca Pérez, Rodrigo Wagner and Claudia Arellano

School of Business, Universidad Adolfo Ibañez, Santiago, Chile

Corresponding author: claudia.arellano@uai.cl

Abstract

Body Mass Index (BMI) is an attribute that can be used for a number of applications ranging from surveillance, identification, image retrieval systems and healthcare, amongst others. Healthcare applications of BMI have gained special importance for detecting people at high risk of suffering malnutrition, obesity, diabetes or cardiovascular problems. This work presents a Bayesian-ViT architecture to estimate BMI from a single face image. The model was trained using a benchmark dataset from literature (VisualBMI) and compared with state of the art algorithms. Results show that the proposed method is comparable to reported methods achieving an overall MAE of 4.434. The Bayesian fashion of the model allows uncertainty to be accounted for. Therefore, results can be selected according to the predicted variance. Using uncertainty, the model achieve MAE values between 2.560 and 3.668 when selecting the best 10% and 75% of the data respectively.

Keywords: Bayesian-ViT, Body Mass Index (BMI), Uncertainty estimation

1 Introduction

Body Mass Index (BMI) is a metric that uses a person's height and weight to estimate their body fat. It is calculated as weight in kilograms divided by the square of height in meters ($kg \div m^2$). The resulting value is usually classified in several categories from underweight to obese. The world health organization proposes the following classes: Normal weight ($BMI \leq 25$), Overweight ($25 < BMI \leq 30$) and Obese I ($30 < BMI \leq 35$) and Obese II ($BMI > 35$). This is a basic indicator that is broadly used to categorize people's health. It is specially useful to detect potential health risks related to weight such as malnutrition, obesity, cardio-vascular diseases or diabetes.

The fast development of computer vision techniques and deep learning in recent years have allowed the use of single face images to estimate BMI. This is mainly due to the correlation between facial geometry and Body Mass. Such techniques provide automatic and non-invasive tools that become very useful for detecting people at high risk of health disorders due to weight such as malnutrition or obesity. Obesity, in particular, has tripled worldwide since 1975, with 13% of adults currently obese and 39% overweight globally. This is directly linked to serious conditions including type 2 diabetes, cardiovascular disease, several cancers, sleep apnea, osteoarthritis, depression and mental health disorders amongst others.

This work, combines state of the art Vision Transformer architecture with Bayesian inference, to estimate BMI from single face images. It includes uncertainty measurements to analyse the results of the Bayesian framework and to improve estimation results. The remaining of the paper is organised as follows: Section 2 presents the most relevant methods for estimating BMI from images and the benchmark dataset used the most in the literature. Section 3 describe the Bayesian ViT architecture implemented while Section 4 reports the results obtained and compares them with the state of the art. Finally, Section 5, reports the conclusion and future work.

2 State of the Art

Computer vision techniques, such as Deep learning, have allowed BMI to be estimated directly from images promoting non invasive systems than can be used remotely. Several methods have been reported, mainly differing on the input data, the architecture and the dataset used. The most commonly used input data are 2D and 3D full-body images [Manichand et al., 2025, Pentakota et al., 2024], multi-view images/videos [Kim et al., 2023, Xiang et al., 2025] and single frontal facial image. The latter being the focus of this paper.

Facial images have been studied in the literature as a key source of information including age, gender, ethnicity, identity, blood pressure, stress level, alcohol consumption, genetic conditions, and weight amongst others [Agbo-Ajala and Viriri, 2021, Dantcheva et al., 2018, Pal et al., 2025]. Therefore, it is not surprising that BMI can also be estimated from face images since it has demonstrated a correlation with facial geometry. Kocabet *et al.* [Kocabey et al., 2017] were some of the first to introduce BMI estimation from a single face image. They proposed a regression method based on the 50-layers ResNet-architecture and suggested that facial images contain discriminatory information pertaining to height, weight and BMI, comparable to that of full-body images and videos.

Since then, several works have been reported in this field (See Table 1). The most well known benchmark datasets in the area are MORPH-II [Ricanek and Tesafaye, 2006], VisualBMI [Kocabey et al., 2017], VIP-Attribute and FIW-BMI [Jiang et al., 2019] and Illinois DOC dataset [Fisher, 2019]. Table 1 shows the best results obtained from each data set. The differences of MAE between datasets are mainly due to bias in race, gender or data imbalance. Siddiqui et al.[Siddiqui et al., 2022] have shown that black males obtained the lowest MAE across different CNN-based architectures. This results shown in table Table 1 where the database with the best MAE results is Morph-II which corresponds to images from the African-American community (with a majority of male subjects). The same study has shown that white females, on the other hand, obtained the worst MAE amongst all the gender-race groups .

Another factor that influence the global BMI estimation results reported in the literature is the imbalance of data regardless of the BMI category (Underweight, Normal, overweight and obese). Most databases do not include the Underweight category and datasets like VIP-attributes tends to over represent only a certain type of population. Since such databases were created from celebrity images which in general tend to have lower BMI than the general population.

In this work, a Bayesian architecture based on Transformer models is proposed for the estimation of BMI from single facial images. The Transformer architecture was first introduced by Dosovitskiy *et al.* [Dosovitskiy et al., 2020] and since then, it has been applied to a wide range of problems in computer vision.

Bayesian Neural Networks [Abdar et al., 2021], on the other hand, is a popular approach to model uncertainty. Given a set of inputs $\{x_1, x_2, x_3, \dots, x_n\}$ with their outputs $\{y_1, y_2, y_3, \dots, y_n\}$ it is possible to model a function f such that $y = f(\mathbf{x})$. A prior distribution over the space of functions $p(f)$ can be used as prior belief. The likelihood can then be defined as $p(y|f, \mathbf{x})$ and using the Bayes rule, the posterior found $p(f|\mathbf{x}, y)$ given the input dataset $\{\mathbf{x}, y\}$. The output is then predicted for a new input x^* by integrating over all possible functions f . Since the posterior is, in general, intractable, variational inference is used to approximate it with a tractable distribution $q(\mathbf{w})$ by minimizing the Kullback-Leibler (KL) divergence. Bayesian modelling has proven to be an useful tool for estimating uncertainty in a wide range of fields ranging from medicine [Ngartera et al., 2024] to agronomy [Arellano et al., 2025]

Paper	MAE	Dataset
[Kocabey et al., 2017]	5.16	VisualBMI
[Haritosh et al., 2019]	3.80	
[Yousaf et al., 2021]	4.99	
[Siddiqui et al., 2022]	5.02	
[Aarotale et al., 2023]	5.13	
[Jiang et al., 2020]	2.61	Morph-II
[Yousaf et al., 2021]	1.72	VIP attributes
[Jiang et al., 2019]	3.70	FIW-BMI
[Sidhpura et al., 2022]	3.22	Illinois DOC

Table 1: Summary of the state of the art techniques for BMI estimation from single face image

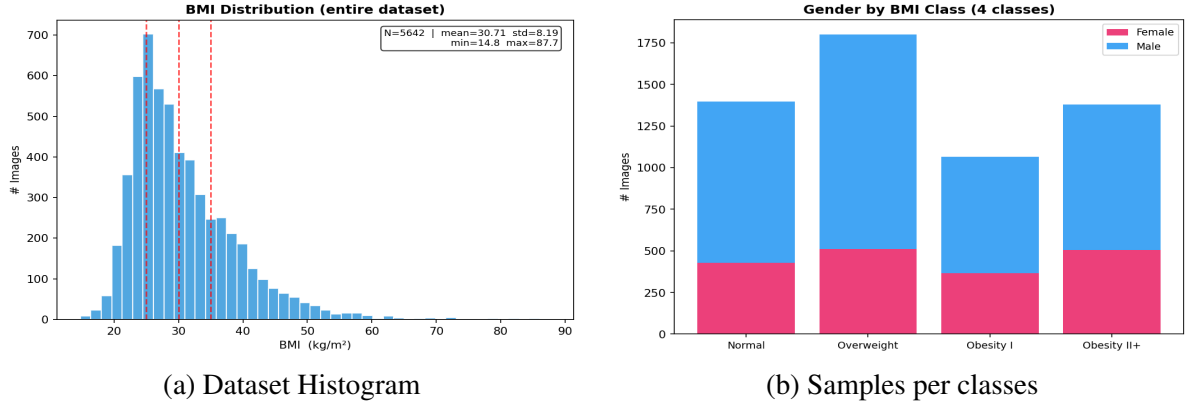


Figure 1: Distribution of the data sets used. In (a) the histogram of the BMI is reported while in (b) The images for each class according to the World Health Organization (WHO) and displayed by gender.

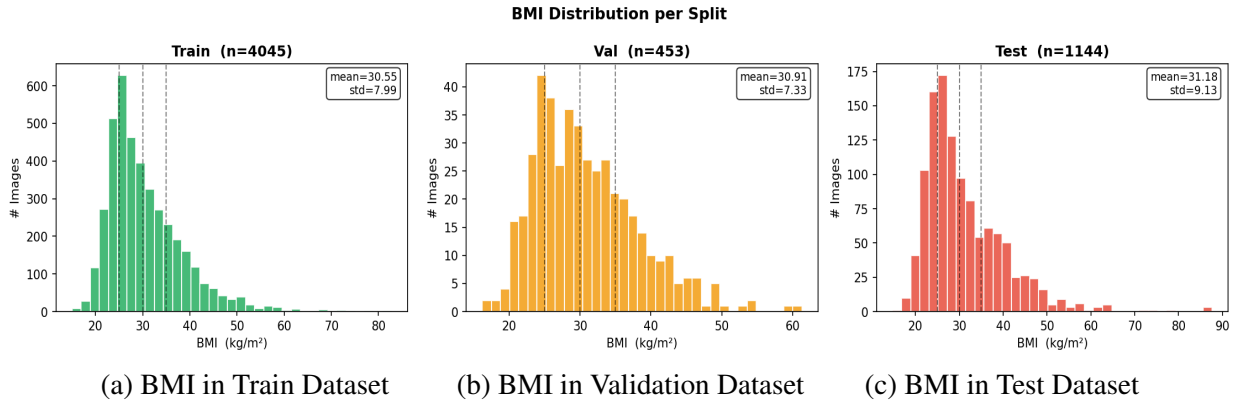


Figure 2: BMI Distribution of the datasets used for training (a), validation (b) and testing (c) . The number of images, the mean and the standard deviation of the BMI distribution is displayed in each figure.

3 Proposed Method

This paper proposes a Bayesian-ViT architecture to estimate BMI from single frontal images. The model has been trained and tested using the benchmark VisualBMI dataset [Jiang et al., 2019]. The dataset and the implementation details of the model are described as follows.

3.1 Dataset

The original VisualBMI data set was collected from the Reddit social media platform. Users reported images before and after weight changes along with their height and weight both before and after. The BMI was then computed for each subject from the reported height and weight values. This data set was originally composed of full body images. For this reason, a Multi-task Cascaded Convolutional Network (MTCNN) was used to detect and crop-out faces.

A total of 5,897 annotated images were available in the public Kaggle release of VisualBMI. After cropping faces and eliminating duplicates, only 5,642 images were considered. Figure 1(a) shows the BMI distribution of the dataset while Figure 1(b) shows the BMI distribution per class and gender. Where the classes are defined according to the World Health organization as: Normal, Overweight, Obesity I and Obesity II+.

The dataset was split in three sets of images: training (4,045), validation (453) and test (1,144). Two considerations were taken into account when splitting the dataset: (1) To preserve the BMI distribution along datasets and (2) to ensure that the datasets are person-level disjoint. Figure 2 shows the distribution of the

Method	Normal	Overweight	Obesity I	Obesity II	Male	Female	Overall
ViT	3.592	2.516	3.275	7.818	4.037	5.098	4.375
Bayesian-ViT	3.361	2.497	3.400	8.196	4.073	5.209	4.434

Table 2: MAE value obtained for each method (ViT and Bayesian-ViT) computed by class (Normal, Overweight, Obesity I and Obesity II) and gender (Male and Female). The final column corresponds to the MAE values computed over all images in the testing dataset for each method.

resulting datasets while Figure 3 shows an example of images taken from the dataset.

3.2 Bayesian-ViT Model

The proposed Bayesian-ViT Model was built using a ViT-B/16 Vision Transformer as backbone [Dosovitskiy et al., 2020]. This model uses a 16×16 patch size and it was first pre-trained using the ImageNet-21k dataset (14M images, 21,000 classes). An additional fine-tuning process with the ImageNet-1k dataset (1.28M images, 1,000 classes) was also performed using the method described by [Steiner et al., 2021]. Two variational layers were added (at the top) to make the regression model Bayesian. These two layers had a double number of parameters with respect to a non variational layer since the parameters to be estimated were distributions. Therefore, each parameters was defined by its mean and standard deviation. The estimation output also became a predictive distribution $p(\hat{y} | \mathbf{x}) = \mathcal{N}(\mu^*, \sigma^{2*})$ of the BMI with mean μ^* and standard deviation σ^* . The variance (σ^{2*}), therefore, acted as a metric of uncertainty, where small variance will indicate a more trusted result. In addition to the variance, the entropy was computed as follows:

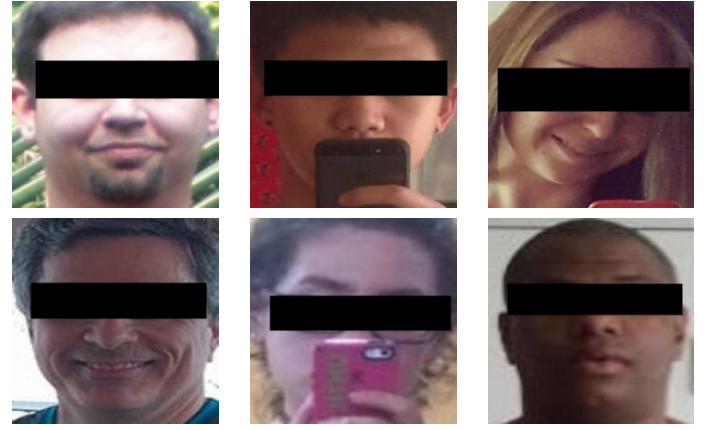


Figure 3: Random samples of facial images from the VisualBMI dataset.

$$\sigma^{2*} = \frac{1}{T} \sum_{t=1}^T (\hat{y}^{(t)} - \mu^*)^2; \quad H[p(\hat{y} | \mathbf{x})] = \frac{1}{2} \ln(2\pi e \sigma^{2*}) \quad (1)$$

Where T corresponded to the number of stochastic forward passes performed in the model (set to 30 in all the experiments). Lower entropy indicated more confident predictions. Those metrics were used to analyse the uncertainty of the predictions. The following section (Section 4) presents the experiments performed and the analysis of the performance of the model where the uncertainty was used to select the best BMI estimation results.

4 Experiments and Results

Experiments performed and the results of BMI estimation using ViT architecture with and without the Bayesian components are presented in section 4.1. Section 4.2, on the other hand, reports the performance of the model when uncertainty is used to select the best results. A comparison of the results with state of the art techniques is reported in Section 4.3.

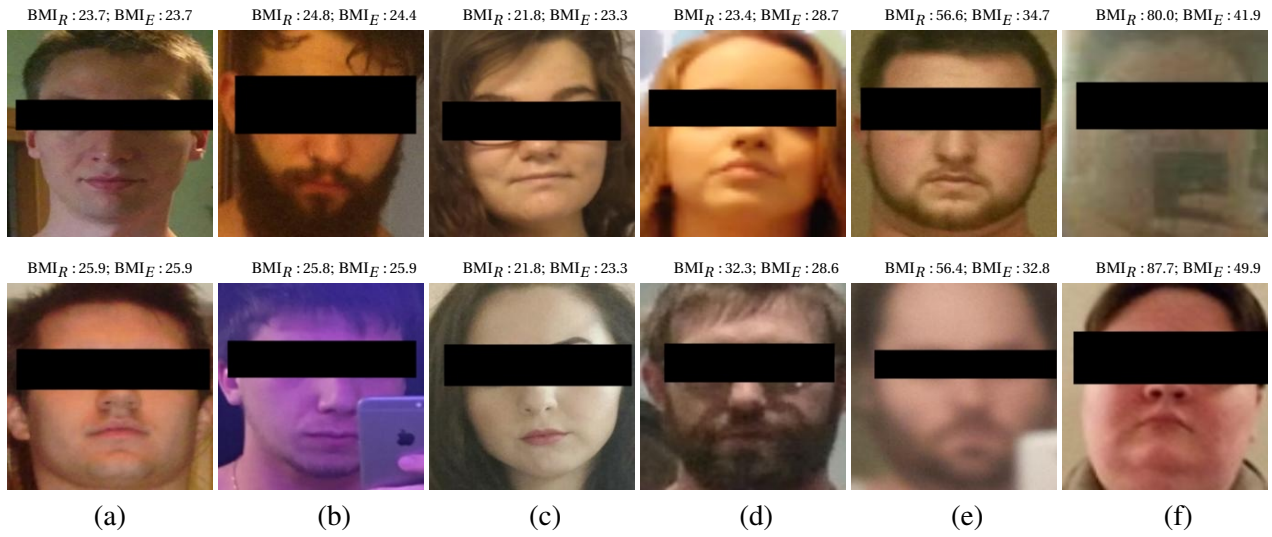


Figure 4: Samples of facial images from the VisualBMI dataset with both their ground truth (BMI_R) and estimated BMI value (BMI_E) respectively. Column (a) shows examples where BMI was perfectly estimated while the last column (g) shows examples with high error between the real and estimated BMI values.

4.1 Bayesian vs Non Bayesian ViT

Both ViT architectures (with and without Bayesian components) were trained using the same hyper-parameters which including batch size equals to 8, AdamW optimizer, *WeightedRandomSampler* functions to balance data according to gender and image resolution set to 224×224 pixels. An ensemble of 4 identical models were considered in each case (trained using different seeds).

Table 2 shows the BMI values estimated for each model and classified according to BMI categories (Normal, overweight, Obesity I and Obesity II) and Gender (Male and female). Results show that deterministic ViT outperforms the Bayesian variant slightly with MAE values of 4.375 and 4.434 for ViT and Bayesian-ViT respectively (only $+0.0595 \text{ kg/m}^2$ of difference). The main contribution of the Bayesian approach is the uncertainty-based selective prediction.

In both cases BMI estimates for female subjects had worse performance than BMI for males. Such results are not surprising since the database had more male subjects than female. The test dataset in particular, contains 68% male and only 32% female subjects. Figure 4 shows several examples of facial images form VisualBMI dataset with their respective BMI ground truth value (BMI_R) and the estimated value (BMI_R) by the proposed Bayesian ViT architecture.

4.2 Uncertainty analysis

Results in Table 2 have shown that adding a Bayesian top to the ViT backbone does not improve the results of the deterministic ViT Model. However, the advantage of using the Bayesian model relies on the possibility of predicting estimate uncertainty. Therefore, the results can be classified according to how certain the model is of the BMI prediction. Two metrics were evaluated to measure uncertainty: variance and entropy (Equation 1).

Table 3 shows the MAE obtained when selecting BMI results according to variance. Predictions with lower variance have less uncertainty and are, therefore, more trustworthy. The test data were classified in percentiles (10,25,50,75 and 100) according to its predicted variance and computed MAE. The results show smaller MAE for the classes Normal and Overweight while Obese I and II present higher MAE values. This result confirms that these two classes are more difficult to predict since their distribution is not uniform. Therefore, the average error in these classes tends to be higher. An example of this tendency is shown in the MAE computed for the Obesity class II which correspond to 10.465 (the highest in the whole dataset). This results correspond to

Method	Normal	Overweight	Obesity I	Obesity II	Male	Female	σ^{2*}
Bayesian-ViT (10%)	1.927	2.577	7.435	10.462	2.579	2.483	2.560
Bayesian-ViT (25%)	2.125	2.149	5.699	10.382	2.743	3.079	2.814
Bayesian-ViT (50%)	2.718	1.874	4.430	9.785	2.941	3.546	3.100
Bayesian-ViT (75%)	3.112	2.129	3.430	9.040	3.420	4.257	3.668
Bayesian-ViT (100%)	3.361	2.497	3.400	8.196	4.073	5.209	4.434

Table 3: MAE obtained for each method computed by class and gender when using variance as discriminator for uncertainty. The overall MAE computed is displayed at the last column.

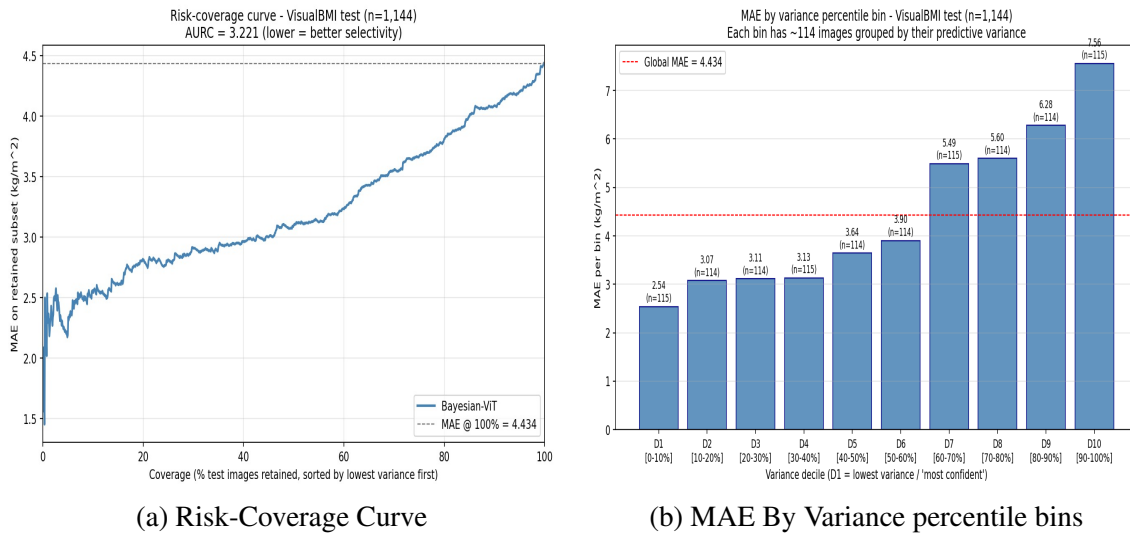


Figure 6: Figure shows the Risk coverage Curve in (a) where the MAE is computed as a function of the number of test images that are selected. Figure (b) shows the MAE as a function of the variance expressed in percentiles

only two samples where the BMI was poorly estimated. However, given the spread of the BMI values in this category, the MAE tends to be bigger and has a greater influence in the overall error. Figure 5 shows the specific images corresponding to this error in the Obesity Class with their respective BMI (real and estimated).

Figure 6 (a) shows the Risk Coverage Curve to evaluate how well an uncertainty measurement (in this case the variance) can identify unreliable predictions. The plot shows the increase in the MAE when more samples with less certainty are being added (x-axis). When all data is considered, the overall MAE is 4.434 as reported in Table 3.

The plot in Figure 6 (b) shows the MAE of the estimated BMI by variance percentile. As can be shown, the MAE increases when the variance increases showing the relation of the predicted variance with the error of the prediction and therefore, demonstrate the capability of this estimator as discriminator for better estimation results. Entropy has also been shown to be a good discriminator for uncertainty as is shown in column 9 of Table 3. In this case, the MAE computed is similar to the MAE achieved by the variance metric.

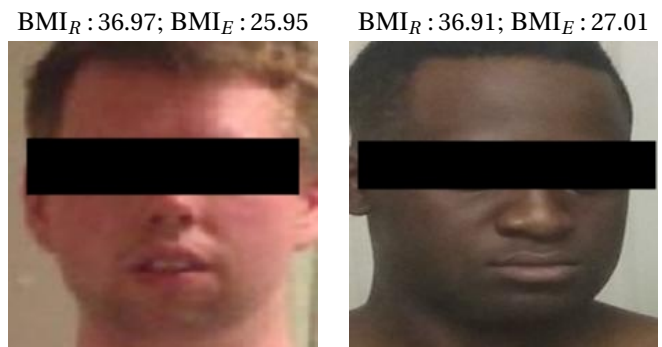


Figure 5: This two images correspond to the Obesity II class with a MAE of 10.462 reported in Table 3 .

4.3 Comparison with the state of the art

The MAE of the estimated BMI for the test dataset is compared to the state of the art in Table 4. Only methods using the same database (visualBMI) are reported. Results show that the proposed method gives better results than most previous methods. There are three main differences between the proposed method and previous work: First, it is one of the first methods reporting the use of a transformer-based architecture where attention head is used instead of convolutional neural networks. Secondly, a Bayesian approach is added to quantify the uncertainty of the prediction and make the algorithm able to select only the most trustworthy results. Finally, the Bayesian ViT architecture was pre-trained using standard images (ImageNet-21k and ImageNet-1k). Therefore, no additional face datasets such as VGGFace, MS-Celeb or CASIA-WebFace were needed.

5 Conclusion and Future Work

A Bayesian-ViT Model to estimate the Body Mass Index (BMI) in the benchmark VisualBMI dataset has been presented. These results are comparable to the state of the art with the advantage that the certainty of the prediction can be quantified. The overall MAE in the ViT and Bayesian ViT was 4.375 and 4.434 respectively.

When selecting only the best predictions (with less uncertainty) the MAE is considerable reduced (up to 2.560 when considering the best 10% of the data and 3.668 when selecting 75%. BMI estimation using a single face image is still a challenging problem and there are several issues that still need to be addressed such as an imbalance in the datasets. For future work, the use of a classification model combined with regression to better estimate obesity classes will be explored. The use of ensemble Bayesian model and metrics for performing the ensemble will also be studied.

Paper	MAE
[Kocabey et al., 2017]	5.160
[Siddiqui et al., 2020]	5.020
[Aarotale et al., 2023]	5.130
[Yousaf et al., 2021]	4.990
Proposed ViT	4.375
Proposed Bayesian ViT	4.434
[Haritosh et al., 2019]	3.800
Proposed Bayesian ViT (75%)	3.668
Proposed Bayesian ViT (50%)	3.100
Proposed Bayesian ViT (25%)	2.814
Proposed Bayesian ViT (10%)	2.560

Table 4: Comparison of the proposed method with state of the art BMI estimation algorithms using single facial images and the VisualBMI dataset

References

- [Aarotale et al., 2023] Aarotale, P. N., Hill, T. J., and Rattani, A. (2023). Patchbmi-net: Lightweight facial patch-based ensemble for bmi prediction. *2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 4022–4028.
- [Abdar et al., 2021] Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., Fieguth, P., Cao, X., Khosravi, A., Acharya, U. R., Makarencov, V., and Nahavandi, S. (2021). A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297.
- [Agbo-Ajala and Viriri, 2021] Agbo-Ajala, O. and Viriri, S. (2021). Deep learning approach for facial age classification: a survey of the state-of-the-art. *Artificial Intelligence Review*, 54.
- [Arellano et al., 2025] Arellano, C., Sagredo, K., Muoz, C., and Govan, J. (2025). Bayesian ensemble model with detection of potential misclassification of wax bloom in blueberry images. *Agronomy*, 15(4).
- [Dantcheva et al., 2018] Dantcheva, A., Brémond, F., and Bilinski, P. T. (2018). Show me your face and i will tell you your height, weight and body mass index. *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3555–3560.

- [Dosovitskiy et al., 2020] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.
- [Fisher, 2019] Fisher, D. J. (2019). Illinois DOC labeled faces dataset. <https://www.kaggle.com/davidjfisher/illinois-doc-labeled-faces-dataset>.
- [Haritosh et al., 2019] Haritosh, A., Gupta, A., Chahal, E. S., Misra, A., and Chandra, S. (2019). A novel method to estimate height, weight and body mass index from face images. In *2019 Twelfth International Conference on Contemporary Computing (IC3)*, pages 1–6.
- [Jiang et al., 2020] Jiang, M., Guo, G., and Mu, G. (2020). Visual bmi estimation from face images using a label distribution based method. *Comput. Vis. Image Underst.*, 197-198:102985.
- [Jiang et al., 2019] Jiang, M., Shang, Y., and Guo, G. (2019). On visual bmi analysis from facial images. *Image and Vision Computing*, 89:183–196.
- [Kim et al., 2023] Kim, S., Lee, K., and Lee, E. C. (2023). Multi-view body image-based prediction of body mass index and various body part sizes. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 6034–6041.
- [Kocabey et al., 2017] Kocabey, E., Camurcu, M., Ofli, F., Aytar, Y., Marin, J., Torralba, A., and Weber, I. (2017). Face-to-bmi: Using computer vision to infer body mass index on social media. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, pages 572–575.
- [Manichand et al., 2025] Manichand, F. R., Deuber, R., Jakob, R., Swerling, S., Rosen, J., Fleisch, E., and Langer, P. (2025). Digital scale: Open-source on-device bmi estimation from smartphone camera images trained on a large-scale real-world dataset. In *AAAI Conference on Artificial Intelligence*.
- [Ngartera et al., 2024] Ngartera, L., Issaka, M. A., and Nadarajah, S. (2024). Application of bayesian neural networks in healthcare: Three case studies. *Machine Learning and Knowledge Extraction*, 6(4):2639–2658.
- [Pal et al., 2025] Pal, B., Huang, S., and Chellappa, R. (2025). A quantitative evaluation of the expressivity of bmi, pose and gender in body embeddings for recognition and identification. *2025 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–10.
- [Pentakota et al., 2024] Pentakota, S. S., Priya, A. L., Reddy, S. S., and Umarani, J. (2024). Body mass index (bmi) estimation from multi-view facial images. *2024 IEEE International Conference on Computer Vision and Machine Intelligence (CVMi)*, pages 1–6.
- [Ricanek and Tesafaye, 2006] Ricanek, K. and Tesafaye, T. (2006). Morph: a longitudinal image database of normal adult age-progression. In *7th International Conference on Automatic Face and Gesture Recognition (FG06)*, pages 341–345.
- [Siddiqui et al., 2020] Siddiqui, H., Rattani, A., Kisku, D. R., and Dean, T. (2020). AI-based bmi inference from facial images: An application to weight monitoring. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1101–1105.
- [Siddiqui et al., 2022] Siddiqui, H., Rattani, A., Ricanek, K., and Hill, T. J. (2022). An examination of bias of facial analysis based bmi prediction models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2925–2934.
- [Sidhpura et al., 2022] Sidhpura, J., Veerhare, R., Shah, P., and Dholay, S. (2022). Face to bmi: A deep learning based approach for computing bmi from face. In *2022 International Conference on Innovative Trends in Information Technology (ICITIIT)*, pages 1–6.
- [Steiner et al., 2021] Steiner, A., Kolesnikov, A., Zhai, X., Wightman, R., Uszkoreit, J., and Beyer, L. (2021). How to train your vit? data, augmentation, and regularization in vision transformers. *Trans. Mach. Learn. Res.*, 2022.
- [Xiang et al., 2025] Xiang, C., Liu, B., Zhao, L., and Zheng, X. (2025). Three-dimension deep model for body mass index estimation from facial image sequences with different poses. *Journal of Visual Communication and Image Representation*, 107:104381.
- [Yousaf et al., 2021] Yousaf, N., Hussein, S., and Sultani, W. (2021). Estimation of bmi from facial images using semantic segmentation based region-aware pooling. *Computers in biology and medicine*, 133:104392.

Are Action Units Priors or Pathways? A Diagnostic Study of Temporal Valence-Arousal Estimation

Riju Das^{1,2} and Soumyabrata Dev^{1,2,3}

¹*The ADAPT Research Centre, Dublin, Ireland*

²*School of Computer Science, University College Dublin, Dublin, Ireland*

³*School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland*

Abstract

Facial Action Units and continuous valence-arousal are complementary but distinct representations of facial affect. While temporal valence-arousal estimation is well studied, the role of Action Units within this task remains unclear: should they be omitted, used as auxiliary training priors, supplied as predicted features, or used as the dominant pathway for valence-arousal prediction? This question is addressed through a systematic architectural study on the overlapping valence-arousal and Action Unit subset of the Aff-Wild2 dataset. The five AU-routing configurations share the same ResNet-50 backbone, Transformer temporal encoder, and training protocol; separate controls vary geometry and temporal-window length. In the single-run comparison, prediction without Action Unit supervision reaches a mean concordance correlation coefficient of 0.3339. Action Unit feature fusion improves this to 0.3472 with concatenation and 0.3643 with gating, while Action Unit-mediated prediction reaches 0.3688. The strongest configuration, auxiliary Action Unit supervision with direct temporal prediction and no MediaPipe geometry, reaches 0.3887 for a single run and 0.3975 ± 0.0133 across five seeds. Frame-level pathway diagnostics extend the analysis beyond the overall concordance correlation coefficient evaluation by comparing routing behaviour across distance from the neutral valence-arousal point, temporal stability, and facial Action Unit activation regimes. These findings indicate that Action Units are most effective as representation-shaping priors rather than as universal valence-arousal pathways.

Keywords: Action Units, Valence-Arousal, Temporal Affect Recognition, Multi-Task Learning, Auxiliary Supervision

1 Introduction

The study of human emotion involves multiple complementary representations. Action Units (AUs), anatomically defined facial muscle movements, provide a low-level, interpretable description of expression mechanics. In contrast, dimensional models using valence (positive-negative) and arousal (low-high activation) capture global affective state. Temporal VA prediction from facial video has been extensively studied, with recent systems employing recurrent, convolutional, transformer-based, and audio-visual temporal encoders. Thus, the central question is not whether temporal modeling benefits VA prediction, but how interpretable AU structure should be incorporated into temporal VA systems.

Existing approaches typically predict AUs and VA independently or use AUs as auxiliary supervision, but seldom examine whether VA prediction should be routed through estimated AUs. This design is intuitive because AUs describe facial muscle activations. However, accurate VA estimation may also depend on temporal context, facial appearance, expression intensity, head motion, and subtle cues outside the limited supervised AU set. Requiring predicted AUs to form a mandatory intermediate representation may therefore improve interpretability while discarding information needed for accurate VA estimation. This study frames AU integration

as a routing problem: whether AUs shape training, enter the VA regressor as predicted features, or become the dominant intermediate representation. This is an architectural prediction question, not a causal analysis of whether facial actions determine affect, which would require interventions and causal assumptions unavailable from observational annotations. On the Aff-Wild2 VA+AU overlap split, controlled architectural comparisons cover VA-only prediction, direct VA prediction with auxiliary AU supervision, AU feature fusion, and AU-mediated prediction. Compact MediaPipe [Lugaresi et al., 2019] geometry and a shorter temporal window serve as controls, while diagnostics compare aggregate and frame-level errors across affect and facial-action regimes.

The key contributions of the paper are:

- AU supervision improves temporal VA estimation without requiring predicted AUs in the inference route: every AU-aware variant exceeds VA-only prediction, while AU-aux direct is strongest in the controlled architectural comparison ¹.
- The supervised 12-AU probability vector is informative but incomplete as a universal VA route. Concatenation, gating, and mediation improve on VA-only prediction yet remain below AU-aux direct, indicating that the shared temporal representation retains additional VA-relevant information.
- The value of AU routing is condition-dependent. Five-seed evaluation and frame-level diagnostics show that AU-aux direct is strongest near neutral and during stable affect, whereas mediation retains selective value for upper-face activation regimes.

2 Related Work

Continuous affect recognition is commonly formulated through the widely used two-dimensional valence-arousal model [Russell, 1980], whereas facial expression analysis often relies on the Facial Action Coding System (FACS) [Ekman and Friesen, 1978]. Large-scale in-the-wild benchmark datasets such as Aff-Wild2 [Kollias and Zafeiriou, 2018] and ABAW [Kollias, 2022b] support evaluation across VA estimation, AU detection, and expression recognition under naturalistic conditions. These datasets introduce substantial variation in pose, illumination, occlusion, identity, and recording environment, making them important testbeds for modern affect recognition models.

Temporal modeling plays a central role in VA estimation from facial video. Recurrent networks, temporal smoothing, causal sequence models, transformers, and audio-visual fusion have all been used to improve over frame-level baselines [Meng et al., 2022, Zhang et al., 2021]. In these approaches, performance gains are primarily attributed to improved temporal representation learning, and the value of temporal context for VA prediction is well established. In contrast, the objective here is to hold the temporal backbone fixed and study the AU-to-VA routing mechanism as the primary experimental variable.

Multi-task learning across AUs, expressions, and VA is motivated by the intuition that lower-level facial signals can regularize higher-level affect prediction [Kollias, 2022a, Das et al., 2024]. ABAW systems and related multi-task approaches [Kollias, 2022b, Kollias, 2022a] often benefit from joint AU and VA supervision, making AU co-supervision a common design component. However, prior studies rarely disentangle three qualitatively different uses of AUs: auxiliary supervision that shapes encoder representations during training, predicted AU features that augment the VA head at inference, and enforced intermediate pathways that require VA to be inferred through AU activations. This distinction has direct architectural implications, because each strategy makes a different assumption about whether the supervised AU space is sufficiently expressive to support continuous VA prediction.

Geometric and landmark features offer a complementary interpretable prior. Compact facial and behavioral descriptors, including landmarks, pose, gaze, and AU features, have been used in related affective-state recognition settings [Das and Dev, 2025]; tools such as MediaPipe [Lugaresi et al., 2019] further enable lightweight

¹Code is publicly available at <https://github.com/rijju-das/AU-Routing-VA> to support reproducibility and further research.

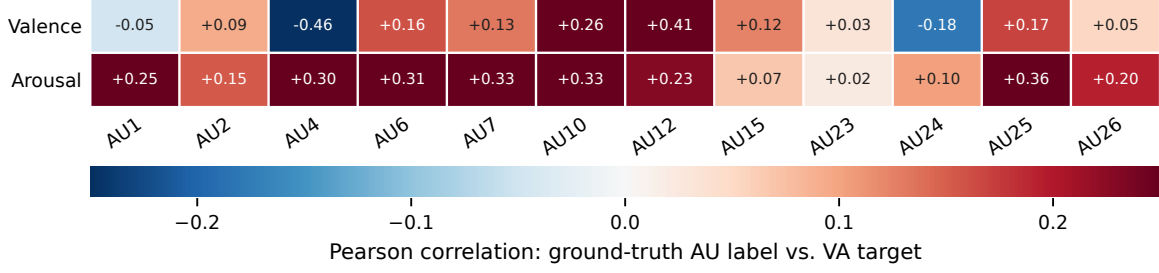


Figure 1: Frame-level Pearson correlations between each binary ground-truth AU label (columns) and the continuous valence and arousal annotations (rows) on the Aff-Wild2 overlap training set. Each cell is an AU-to-VA correlation computed across paired annotated frames.

facial geometry extraction. However, their value as fused priors for strong RGB-based temporal VA models remains unclear. Such features may provide explicit facial structure, but they may also be redundant with learned appearance representations or introduce noise when landmark detection is unreliable. MediaPipe geometry is therefore included as a controlled ablation rather than assumed to improve VA prediction.

3 Diagnostic Framework

3.1 Data and Setup

All experiments are conducted on the Aff-Wild2 VA+AU overlap subset, where each training sample contains both continuous valence-arousal (VA) annotations and binary labels for 12 facial Action Units: AU1, AU2, AU4, AU6, AU7, AU10, AU12, AU15, AU23, AU24, AU25, and AU26. The split contains 295 training videos and 65 validation videos. Unless otherwise stated, videos are sampled as 16-frame windows with a stride of 8. Frames are resized to 112×112 and normalized using ImageNet statistics. A compact temporal-context ablation uses 8-frame windows while keeping the remaining preprocessing and model configuration unchanged. Figure 1 reports the frame-level Pearson correlation between each binary ground-truth AU label and each continuous VA target (AU-to-VA correlations) on the overlapping training set. Their heterogeneous signs and absolute values, ranging from 0.02 to 0.46, show that individual AUs carry related but incomplete linear information about VA. The figure therefore motivates an empirical test of AU routing; it does not establish whether the combined AU vector is sufficient for VA estimation.

3.2 AU Routing Variants

All model variants use the same shared visual-temporal encoder. Given a window of T RGB frames, a ResNet-50 backbone pretrained on ImageNet extracts per-frame visual features. The resulting feature sequence is passed to a two-layer vanilla self-attention Transformer encoder [Vaswani et al., 2017], producing frame-wise temporal representations $\mathbf{f}_t \in \mathbb{R}^{256}$. Across the routing comparisons, the encoder architecture and representation dimensionality remain fixed, while the use of AU information varies. The AU head maps this representation to 12 frame-level activation probabilities, $\hat{\mathbf{u}}_t = \sigma(h_{\text{AU}}(\mathbf{f}_t)) \in [0, 1]^{12}$, while the direct VA head predicts a two-dimensional valence-arousal coordinate, $\hat{\mathbf{y}}_t^{\text{direct}} = \tanh(h_{\text{VA}}(\mathbf{f}_t)) \in [-1, 1]^2$. The h . terms denote independently learned two-layer feed-forward heads. Figure 2 summarizes how the shared temporal representation and predicted AUs are used across the five comparison variants.

The variants span a continuum from no AU use to AU-dominant routing. **Pure VA-only** ignores AU labels and predicts VA directly from $\mathbf{f}_t$. **AU-aux direct** is a multi-task configuration where the AU and VA losses jointly shape the shared encoder during training, while VA is predicted directly from $\mathbf{f}_t$ rather than from $\hat{\mathbf{u}}_t$. Unlike AU-aux direct, the remaining three AU-aware variants explicitly introduce predicted AUs into VA estimation. In **AU concatenation**, the VA head receives both the temporal representation and predicted AU

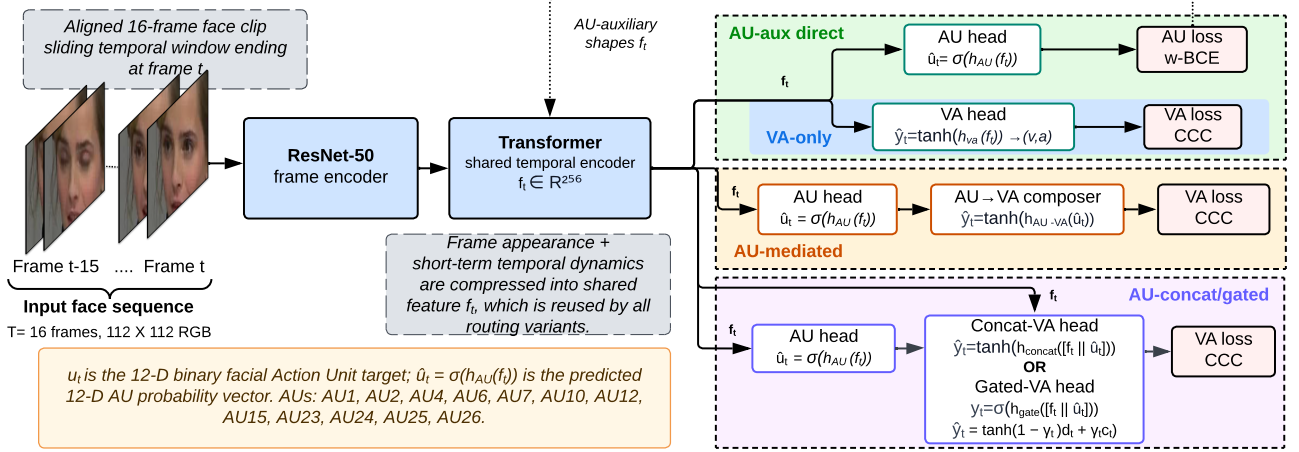


Figure 2: Five comparison variants sharing the same ResNet-50+Transformer backbone ($\mathbf{f}_t \in \mathbb{R}^{256}$, $T=16$ frames). The blue branch represents the pure VA-only baseline; adding the auxiliary AU head yields AU-aux direct. AU-mediated VA estimates VA primarily from predicted AUs, while AU concatenation and AU-gated fusion introduce them explicitly at the VA head. Repeated loss connections and the scalar residual path are omitted for visual clarity.

probabilities:

$$\hat{\mathbf{y}}_t^{\text{concat}} = \tanh(h_{\text{concat}}([\mathbf{f}_t; \hat{\mathbf{u}}_t])). \quad (1)$$

In **AU-gated fusion**, a two-dimensional direct VA score $\mathbf{d}_t \in \mathbb{R}^2$ derived from $\mathbf{f}_t$ and an AU-derived VA score $\mathbf{c}_t \in \mathbb{R}^2$ derived from $\hat{\mathbf{u}}_t$ are combined using a frame-level, per-dimension gate $\gamma_t \in [0, 1]^2$:

$$\gamma_t = \sigma(h_{\text{gate}}([\mathbf{f}_t; \hat{\mathbf{u}}_t])), \quad \hat{\mathbf{y}}_t^{\text{gated}} = \tanh((1 - \gamma_t)\mathbf{d}_t + \gamma_t\mathbf{c}_t). \quad (2)$$

In **AU-mediated VA**, the prediction passes primarily through the predicted AU probabilities:

$$\hat{\mathbf{y}}_t^{\text{AU}} = \tanh(h_{\text{AU2VA}}(\hat{\mathbf{u}}_t) + \alpha r_t), \quad (3)$$

where $r_t = \frac{1}{256} \sum_{j=1}^{256} f_{t,j}$ is a scalar temporal-feature summary broadcast to both VA dimensions, and α is a learnable scalar initialized to 0.01. This produces an AU-structured route with a low-dimensional residual rather than a strictly AU-only pathway.

4 Experiments and Results

4.1 Training and Evaluation Setup

Training uses AdamW for 20 epochs (learning rate 5×10^{-5} ; weight decay 10^{-5}), cosine scheduling, unit-norm gradient clipping, and batches of 8 accumulated over four steps (effective batch 32). ResNet-50 is frozen for 10 epochs and then unfrozen with the learning rate doubled. For $d \in \{v, a\}$, the concordance correlation coefficient (CCC) is

$$\rho_c^{(d)} = \frac{2 \text{cov}(\mathbf{y}^{(d)}, \hat{\mathbf{y}}^{(d)})}{\text{var}(\mathbf{y}^{(d)}) + \text{var}(\hat{\mathbf{y}}^{(d)}) + (\mu_{\mathbf{y}^{(d)}} - \mu_{\hat{\mathbf{y}}^{(d)}})^2 + \epsilon}, \quad (4)$$

where $\mathbf{y}^{(d)}$ and $\hat{\mathbf{y}}^{(d)}$ are valid batch labels and predictions; cov, var, and μ are population moments; and $\epsilon = 10^{-8}$ provides numerical stability. The VA loss is $\mathcal{L}_{\text{VA}} = (1 - \rho_c^{(v)}) + 1.5(1 - \rho_c^{(a)})$, mildly upweighting arousal. AU detection uses weighted binary cross-entropy to address class imbalance, and AU-aware models optimize the sum of the AU and VA losses with unit task weights. The pure VA-only ablation sets the AU loss weight to zero. The best epoch for each run is selected using validation mean CCC, computed as the average of valence CCC and arousal CCC. AU quality is reported with macro F1 at threshold 0.5.

Geometry is a secondary control, not part of the AU-routing mechanism. This geometry is extracted using MediaPipe to test whether explicit facial shape and pose cues complement the learned RGB-temporal representation or alter the relative behaviour of the routing variants. Only “+ MediaPipe” variants receive a precomputed 40-D vector: 32 normalized coordinates from 16 landmarks, three pose proxies, four bounding-box values, and one validity flag. A learned gate adds its projected embedding residually to the RGB-temporal representation, and disk caching fixes these inputs across runs.

4.2 AU Role, Robustness, and Geometry Ablations

Table 1 compares the tested AU-integration strategies and the two control ablations for geometry and temporal context, forming a clear performance ladder. Pure VA-only prediction is weakest, reaching 0.3339 mean CCC. Adding predicted AUs as explicit VA features improves performance to 0.3472 with concatenation and 0.3643 with gating. AU-mediated VA reaches 0.3688. The best single run is AU-aux direct, reaching 0.3887.

Group	Variant	Mean CCC	CCC _v	CCC _a	AU F1
AU role	Pure VA-only, no AU loss	0.3339	0.2066	0.4613	-
	AU concatenation, no MediaPipe	0.3472	0.2269	0.4675	0.4440
	AU-gated fusion, no MediaPipe	0.3643	0.2851	0.4435	0.4478
	AU-mediated VA, no MediaPipe	0.3688	0.2870	0.4506	0.4334
	AU-aux direct, no MediaPipe	0.3887	0.2800	0.4973	0.4393
Geometry	AU-aux direct + MediaPipe	0.3574	0.2717	0.4430	0.4428
	AU-mediated VA + MediaPipe	0.3470	0.2462	0.4477	0.4323
Temporal	AU-aux direct + MediaPipe, $T = 8$	0.3456	0.2920	0.3993	0.4529

Table 1: Main validation results on the Aff-Wild2 VA+AU overlap split, grouped by the tested factor. All variants use 16-frame windows unless explicitly marked otherwise. AU-aware variants improve over pure VA-only prediction, and the strongest result is AU-aux direct.

Together, Table 1 and Figure 3 support the central claim. Every AU-aware variant improves over pure VA-only prediction, but the strongest use of AUs is not hard mediation. AU concatenation, AU-gated fusion, and AU-mediated VA progressively improve on VA-only prediction but remain below AU-aux direct, which exceeds AU-mediated VA without MediaPipe by +0.0199 CCC. Thus, AU supervision improves temporal representations, which retain VA-relevant information beyond predicted AU probabilities.

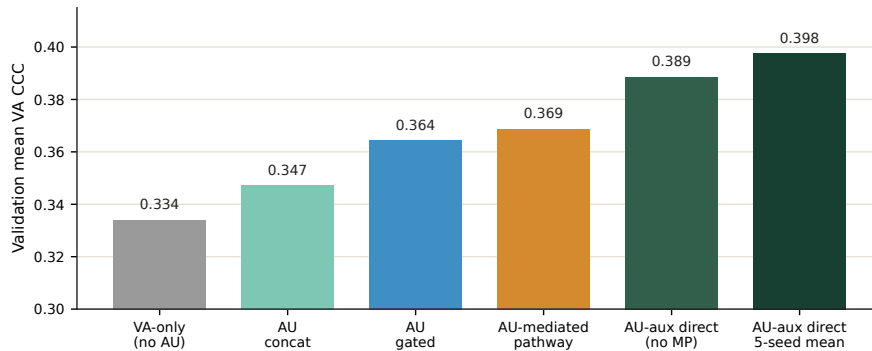


Figure 3: AU-role ablation ladder. The best single run is AU-aux direct at 0.3887 mean CCC; explicit AU routes improve over VA-only prediction but remain weaker. MP denotes MediaPipe.

Because single runs can overstate differences, Table 2 reports five-seed robustness for AU-aux direct and the matching MediaPipe variant. AU-aux direct, no MediaPipe obtains 0.3975 ± 0.0133 mean CCC, while AU-aux direct + MediaPipe obtains 0.3764 ± 0.0131 . Across the same five matched seeds summarized in Table 2, the no-MediaPipe model wins in 4 out of 5 paired comparisons with a mean paired gain of +0.0211 CCC. Thus Table 2 shows that the main conclusion is not driven by a single favourable run.

The geometry and temporal rows in Table 1 show a negative but informative result. Compact MediaPipe geometry reduces VA performance for both routing conditions: AU-aux direct drops from 0.3887 to 0.3574

Variant	Mean CCC	CCC _v	CCC _a	AU F1
AU-aux direct, no MediaPipe, 5 seeds	0.3975 ± 0.0133	0.3273 ± 0.0338	0.4678 ± 0.0180	0.4458 ± 0.0063
AU-aux direct + MediaPipe, 5 seeds	0.3764 ± 0.0131	0.2925 ± 0.0166	0.4603 ± 0.0302	0.4376 ± 0.0059

Table 2: Five-seed validation robustness. AU-aux direct is stronger on average; over matched seeds, the no-MediaPipe model wins 4/5 paired comparisons with a mean paired gain of +0.0211 CCC.

mean CCC, and AU-mediated VA drops from 0.3688 to 0.3470. The temporal-window row further shows that the 8-frame MediaPipe variant is weaker than the corresponding 16-frame MediaPipe model for VA (0.3456 vs. 0.3574), although it yields the highest AU F1 (0.4529). This suggests that VA benefits from longer temporal context, while AU detection may be more local and frame-proximal. More importantly, the geometry experiment cautions against assuming that interpretable priors automatically improve a strong RGB-temporal encoder.

4.3 Pathway Diagnostics

Aggregate CCC identifies the winner, but not the mechanism. To test whether each AU route helps in different affect regimes, validation outputs are aligned by temporal-window position and their frame-level errors are compared between pathways. For models *A* and *B*, listed before and after “vs.” in Table 3, Base CCC and Comp. CCC report their respective aggregate scores. ΔMAE is $\text{MAE}_B - \text{MAE}_A$, so positive values favour baseline *A*; win rate is the fraction of aligned predictions for which *A* has lower VA error. Ground-truth distance from neutral is $\delta_t = \sqrt{v_t^2 + a_t^2}$ and is grouped into near-neutral ($\delta_t < 0.15$), low ($0.15 \leq \delta_t < 0.35$), medium ($0.35 \leq \delta_t < 0.60$), and high ($\delta_t \geq 0.60$) bins. Table 3 summarizes the comparisons.

Comparison	Base CCC	Comp. CCC	ΔMAE	Win Rate
AU-aux direct vs. AU-mediated VA, no MP	0.3871	0.3675	+0.0012	0.503
AU-aux direct vs. AU-mediated VA, MP	0.3561	0.3461	+0.0327	0.624
AU-aux direct, no MP vs. MP	0.3871	0.3561	+0.0164	0.542
AU-mediated VA, no MP vs. MP	0.3675	0.3461	+0.0480	0.638
AU-aux direct, MP, $T = 16$ vs. $T = 8$	0.3365	0.3223	+0.0040	0.515

Table 3: Diagnostic comparisons; MP denotes MediaPipe and T the temporal-window length in frames. Positive ΔMAE favours the baseline. The first four use 545,327 aligned valid VA predictions from overlapping windows; the final comparison aligns VA predictions at the shared starting frame of each matched window.

To show whether the aggregate difference varies across conditions, the first comparison in Table 3 is further decomposed across affect and AU regimes in Figure 4. Although the overall difference between AU-aux direct and AU-mediated VA without MediaPipe is small (+0.0012), substantially larger differences emerge within individual subgroups. In this decomposition, AU-aux direct is especially favourable for negative/low-arousal frames ($\Delta\text{MAE} = +0.0342$), near-neutral frames (+0.0244), frames at low VA distance (+0.0141), and temporally stable windows (+0.0122). Here, temporal stability denotes a mean ground-truth VA standard deviation below 0.05 within a window. These results suggest that the supervised AU activations may not fully characterize VA under these conditions. In contrast, the AU-mediated model remains favourable for some upper-face patterns, including upper-face-only frames, where at least one upper-face AU is active and lower-face AUs are inactive (−0.0333), and eye/cheek-only frames (−0.0153). AU mediation therefore remains informative in selected facial-action regimes, although it is not the strongest overall route for VA estimation.

5 Discussion

The results separate two roles for AUs that are often conflated in multi-task affect models: supervision and routing. AU supervision is useful because it encourages the shared temporal representation to encode facial-action structure. However, using the predicted AU vector $\hat{\mathbf{u}}_t$ as the primary information route for VA is more restrictive. The AU-aux direct model keeps VA prediction conditioned on the full temporal representation while

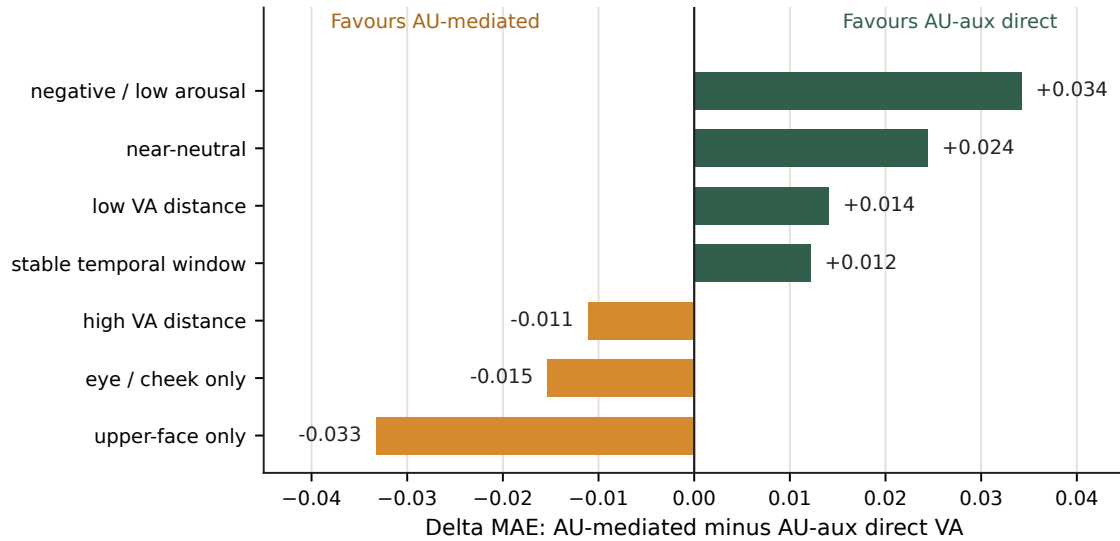


Figure 4: Grouped pathway diagnostics for AU-aux direct versus AU-mediated VA without MediaPipe. Positive Δ MAE favours AU-aux direct; negative values favour AU-mediated routing.

still using AU labels to shape that representation during training. This design preserves access to cues that fall outside the 12 supervised AUs, including expression intensity, facial appearance, person-specific dynamics, and sequence-level context. The concatenation and gated routing variants occupy an intermediate position between priors and pathways. They demonstrate that predicted AU probabilities can serve as useful VA features, particularly when a learned gate modulates the contribution of the AU-derived estimate. That neither variant surpasses AU-aux direct suggests that, in the tested setting, the expressiveness of the AU probability vector may be a stronger limitation than the choice of fusion mechanism. Once AUs have shaped the encoder, providing them explicitly at the VA head offers diminishing returns. These findings concern information flow within the trained models; they do not establish that AUs cause, mediate, or determine human affect, and the observed differences across affect and AU subgroups should be interpreted as associations rather than causal effects.

For the geometry control, the result applies only to the tested compact descriptor, not to facial geometry generally. The descriptor reduces VA performance in both routing conditions, and the no-MediaPipe model leads in four of five paired runs (Table 2). Static coordinates may be redundant with learned RGB shape cues or noisy under imperfect landmark fits. Richer dynamic geometry, including landmark velocities, distances, and symmetry, was not evaluated in this work. Therefore, the supported conclusion is limited to this descriptor and fusion setting.

The study is also limited to the Aff-Wild2 VA+AU overlap subset. Paired frame annotations enable controlled comparison, but the hierarchy may depend on the dataset’s annotation protocol, subject and recording distribution, label reliability, and 12-AU inventory. AU type and count are fixed, so the experiments cannot attribute the benefit to particular units or determine whether a larger inventory would change the ranking. The evidence is therefore within-dataset rather than universal. Replication, cross-dataset transfer, and AU-subset ablations are needed to test generalization. The diagnostic asymmetry motivates hybrid architectures that apply AU-mediated routing selectively for high-confidence upper-face activations, while retaining direct prediction when AU evidence is weak or ambiguous.

6 Conclusion

This work tests AUs as auxiliary supervision, predicted features, and AU-dominant pathways for temporal VA estimation. Under controlled comparisons using the same ResNet-50 and Transformer on the Aff-Wild2 VA+AU overlap split, every AU-aware variant exceeds pure VA-only prediction (0.3339 mean CCC). AU-aux direct is strongest at 0.3887 for a single run and 0.3975 ± 0.0133 across five seeds, ahead of concatenation

(0.3472), gated fusion (0.3643), and mediation (0.3688). Diagnostics show its largest advantages near neutral and within temporally stable windows, while mediation retains selective value for upper-face activations. Within the evaluated dataset and 12-AU inventory, AUs are therefore valuable but conditional: they provide interpretable facial-action evidence yet do not form a complete route for contextual or weakly expressed affect. The findings support AUs as representation-shaping priors rather than universal VA pathways and motivate selective routing when AU evidence is reliable.

Acknowledgments

This research was conducted with the financial support of the Research Ireland Centre under Grant Agreement No. 13/RC/2106_P2 at the ADAPT RI Centre at University College Dublin. ADAPT, the RI Centre for AI-Driven Digital Content Technology, is funded by Research Ireland through the RI Centres Programme.

References

- [Das and Dev, 2025] Das, R. and Dev, S. (2025). Optimizing student engagement detection using facial and behavioral features. *Neural Computing and Applications*, 37(23):19063–19085.
- [Das et al., 2024] Das, R., Dhang, S., and Dev, S. (2024). Integrated approach for simultaneous estimation of emotional states and valence-arousal levels in facial images. In *IET Conference Proceedings CP887*, volume 2024, pages 258–265. IET.
- [Ekman and Friesen, 1978] Ekman, P. and Friesen, W. V. (1978). Facial action coding system. *Environmental Psychology & Nonverbal Behavior*.
- [Kollias, 2022a] Kollias, D. (2022a). Abaw: learning from synthetic data & multi-task learning challenges. In *European Conference on Computer Vision*, pages 157–172. Springer.
- [Kollias, 2022b] Kollias, D. (2022b). Abaw: Valence-arousal estimation, expression recognition, action unit detection & multi-task learning challenges. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2328–2336.
- [Kollias and Zafeiriou, 2018] Kollias, D. and Zafeiriou, S. (2018). Aff-wild2: Extending the aff-wild database for affect recognition. *arXiv preprint arXiv:1811.07770*.
- [Lugaresi et al., 2019] Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.-L., Yong, M. G., Lee, J., et al. (2019). Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.
- [Meng et al., 2022] Meng, L., Liu, Y., Liu, X., Huang, Z., Jiang, W., Zhang, T., Liu, C., and Jin, Q. (2022). Valence and arousal estimation based on multimodal temporal-aware features for videos in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2345–2352.
- [Russell, 1980] Russell, J. A. (1980). A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161.
- [Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- [Zhang et al., 2021] Zhang, S., Ding, Y., Wei, Z., and Guan, C. (2021). Continuous emotion recognition with audio-visual leader-follower attentive fusion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3567–3574.

Parameter-Efficient Adaptation of V-JEPA 2 for Egocentric Action Anticipation

Ayoub Sahlaoui¹, Salaheddin Alakkari², and Younes Dhassi¹

¹*Department of Computer Science, Sidi Mohamed Ben Abdellah University*

²*Department of Computer Science and Information Systems, University of Limerick*

Abstract

Egocentric action anticipation requires predicting an action before it begins, given only the preceding visual context. We test whether low-rank adaptation (LoRA) can parameter-efficiently specialize V-JEPA 2, a self-supervised video encoder trained with a feature-prediction objective. On a controlled 22-participant subset of EPIC-KITCHENS-100 with a shared attentive probe, we compare frozen probing against LoRA at ranks $r \in \{4, 8, 16, 32\}$ and a matched-budget upper-layer variant. Full-layer LoRA with $r = 4$ reaches the best action mean-class Recall@5, raising it from 16.39% to 18.67% with 0.39M trainable parameters (0.13% of the backbone). Higher ranks bring no further gain, and concentrating the same adapter budget in the upper six layers hurts performance. These results show that small, broadly distributed low-rank updates can adapt V-JEPA 2 without full fine-tuning, while trading extra training memory and time for a compact stored adapter.

Keywords: Action Anticipation, Parameter-Efficient Fine-Tuning, LoRA, V-JEPA, Egocentric Video

1 Introduction

Action anticipation asks a model to predict what a person will do before the action begins. Unlike recognition, the target action itself is unobserved. Egocentric video makes this harder: object interactions are fine-grained and class frequencies are heavily skewed. Anticipation matters for assistive systems and embodied agents that must act before an event has completed.

Large-scale self-supervised video encoders are a strong starting point for this problem. V-JEPA 2 [Assran et al., 2025] is a recent large-scale self-supervised video encoder pre-trained with a feature-prediction objective. Rather than reconstructing pixels, a joint-embedding predictive architecture (JEPA) learns to predict latent representations of masked content. JEPA models are a natural fit for anticipation, since both pre-training and downstream evaluation require reasoning about unobserved information.

Full fine-tuning of a large video backbone is expensive and risks overwriting useful pre-trained representations. Parameter-efficient fine-tuning (PEFT) sidesteps this by learning a small number of task-specific parameters while keeping the pre-trained weights fixed. Low-rank adaptation (LoRA) [Hu et al., 2022] is now a standard tool for adapting large foundation models, but it has been studied mainly on language models and reconstruction- or contrastively-trained visual encoders. Whether the low-rank update hypothesis behind LoRA holds for representations learned through feature prediction, as in a JEPA video model, has not been established empirically. To our knowledge, this is the first systematic evaluation of LoRA adaptation of V-JEPA 2 for egocentric action anticipation. We treat this as an empirical question rather than a claim of methodological novelty: our aim is a practical, evidence-based recipe for adapting V-JEPA 2.

Our study yields three findings:

- LoRA adaptation of V-JEPA 2 improves egocentric action anticipation over frozen probing, raising action mR@5 from 16.39% to 18.67% with only 0.39M adapter parameters (0.13% of the backbone).

- The downstream update has low intrinsic rank: sweeping $r \in \{4, 8, 16, 32\}$ shows that $r = 4$ is already optimal and higher ranks bring no gain while beginning to overfit.
- Where adapters are placed matters more than how large they are: under a fixed 0.39M budget, spreading small adapters across all 24 encoder layers beats concentrating them in the upper six layers by 1.30 pp.

2 Related Work

Egocentric action anticipation. EPIC-KITCHENS introduced large-scale egocentric video understanding in everyday kitchens [Damen et al., 2018], and EPIC-KITCHENS-100 expanded the dataset and evaluation protocol [Damen et al., 2022]. Action anticipation predicts a future verb–noun action before its onset. Prior work has explored attention-based modality fusion for this setting [Furnari and Farinella, 2019]. We focus on adapting a self-supervised video encoder under a controlled anticipation protocol.

Joint-embedding predictive architectures. JEPA models learn by predicting latent representations of masked content rather than reconstructing pixels [Assran et al., 2023]. V-JEPA 2 [Assran et al., 2025] scales this idea to internet-scale video and shows strong transfer to video understanding and anticipation; V-JEPA 2.1 later reports further gains through denser self-supervision [Mur-Labadia et al., 2026]. We study how much a JEPA video encoder can be specialized with few trainable parameters when the egocentric training set is small.

Video representation learning. Video encoders have benefited from supervised pre-training on action datasets such as Kinetics [Kay et al., 2017], contrastive learning [Chen et al., 2020, He et al., 2020, Radford et al., 2021], and masked reconstruction [He et al., 2022, Tong et al., 2022]. JEPA models differ from reconstruction-based approaches by predicting latent targets rather than pixels.

Parameter-efficient fine-tuning. Full fine-tuning a 307M-parameter encoder is costly on a small task-specific subset. PEFT methods reduce the trainable parameter count, with adapter layers [Houlsby et al., 2019], prefix tuning [Li and Liang, 2021], and LoRA [Hu et al., 2022] being common examples. LoRA keeps the original weights fixed and learns a low-rank residual update. Our focus is empirical: whether this low-rank update is sufficient for a feature-prediction video encoder, and how rank and layer placement affect anticipation.

Attentive probing. Frozen feature evaluation relies on probes to measure representation quality. We adopt the attentive probing design from V-JEPA 2 [Assran et al., 2025] and the prior V-JEPA work [Bardes et al., 2024] and train a shared transformer-based probe across all experiments. The probe serves as a fixed evaluation head; it is part of our setup, not a contribution.

3 Method

3.1 V-JEPA 2 Backbone

We use the V-JEPA 2 ViT-Large encoder `facebook/vjepa2-vitl-fpc64-256` [Assran et al., 2025]. The backbone has 307M parameters, 24 transformer layers, a 1024-dimensional embedding, and 16 attention heads. It uses three-dimensional rotary positional embeddings (RoPE) for spatiotemporal position encoding. V-JEPA 2 was pre-trained by Meta on approximately 22M videos using a joint-embedding feature-prediction objective. During pre-training, a context encoder predicts latent targets from an exponential-moving-average target encoder under high-ratio block masking. We adapt the released context encoder in all downstream experiments.

Table 1: Architecture and training configuration.

Component	Configuration
Backbone	V-JEPA 2 ViT-Large
Backbone size	307M parameters
Encoder	24 layers, 1024 dim., 16 heads
Position encoding	3D RoPE
LoRA targets	Query and value projections
LoRA ranks	4, 8, 16, 32
LoRA scaling	$\alpha = 2r$
Attentive probe	4 blocks, 3 query tokens, 67.4M params
Input	32 frames, 8 FPS, 256 ² px
Context / gap	4 s / 1 s
Optimizer	AdamW, batch size 8
Learning rate	10^{-4} (baseline) / 5×10^{-5} (LoRA)

3.2 LoRA Adaptation

For a frozen pre-trained projection matrix $W_0 \in \mathbb{R}^{d \times k}$, LoRA [Hu et al., 2022] learns a low-rank residual update:

$$W = W_0 + \Delta W, \quad \Delta W = BA, \quad (1)$$

where $B \in \mathbb{R}^{d \times r}$, $A \in \mathbb{R}^{r \times k}$, and $r \ll \min(d, k)$. Only A and B are updated during training. The adapter output is scaled by α/r before addition, where we set $\alpha = 2r$ so that the effective scaling is constant across all rank configurations.

Using the Hugging Face PEFT library, we inject LoRA adapters into the query and value projections of self-attention. Our main variants apply adapters to all 24 encoder layers. We conduct a rank sensitivity study with $r \in \{4, 8, 16, 32\}$, corresponding to 0.39M, 0.79M, 1.57M, and 3.15M trainable adapter parameters. To study layer placement under a fixed budget, we also apply rank-16 adapters only to layers 18–23. This upper-layer configuration contains 0.39M adapter parameters, matching full-layer LoRA with $r = 4$.

3.3 Attentive Probe

All experiments use the same attentive probe, trained from scratch. The probe has four pre-LayerNorm transformer blocks, 16 attention heads, an MLP ratio of 4.0, and zero dropout. Three learnable cross-attention query tokens summarize the video representation for verb, noun, and action classification. The probe totals 67.4M trainable parameters. For the frozen baseline, only the probe is trained. For LoRA variants, the probe and low-rank adapter matrices are trained jointly while the original backbone weights remain frozen. Table 1 summarizes the full architecture and training configuration.

4 Experimental Setup

4.1 Dataset and Subset Construction

We train and evaluate on a 22-participant subset of EPIC-KITCHENS-100 [Damen et al., 2022], comprising approximately 56,311 training clips and 6,720 validation clips from the official train and validation CSVs. The full dataset spans 37 participants; our subset was determined only by frame-storage capacity, and all experiments use the same subset. We therefore report internally controlled comparisons and discuss full-dataset V-JEPA results only as context in Section 6.

Frames are pre-extracted JPEG images obtained from the official EPIC-KITCHENS-100 distribution hosted by the University of Bristol [Damen et al., 2022]. The probe predicts the full training vocabulary: 97 verbs, 300 nouns, and 3,568 verb–noun actions. Mean-class recall is computed over validation-present classes, including 895 actions in our subset.

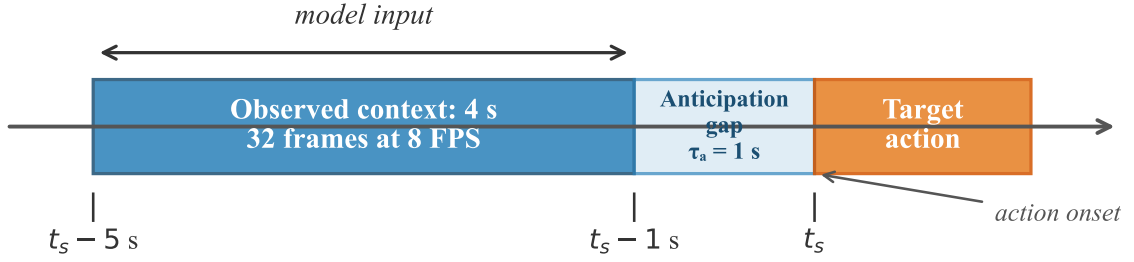


Figure 1: Temporal anticipation protocol. The model receives a four-second context window ending one second before the target action begins.

4.2 Temporal Anticipation Protocol

For an action starting at time t_s , the observation ends one second before its onset:

$$t_{\text{end}} = t_s - \tau_a, \quad \tau_a = 1.0 \text{ s}. \quad (2)$$

We sample a four-second context window ending at t_{end} , as illustrated in Figure 1. Each input therefore contains 32 frames at 8 FPS, linearly sampled from source video recorded at 60 FPS. The model sees none of the anticipated action.

The one-second gap follows the standard EPIC-KITCHENS-100 anticipation protocol [Damen et al., 2022]. The four-second, 8 FPS window gives V-JEPA 2 its expected 32-frame input, covers immediate pre-action context, and avoids near-duplicate frames from the 60 FPS source videos. These temporal settings were fixed before the LoRA sweep and held constant across all variants. Frames are processed at 256×256 resolution with center cropping and the default normalization from the Hugging Face `AutoVideoProcessor`. We use no additional data augmentation.

4.3 Loss Function

We optimize focal loss [Lin et al., 2017] independently for verbs, nouns, and actions. For ground-truth class probability p_t , focal loss is:

$$\mathcal{L}_{\text{focal}}(p_t) = -(1 - p_t)^\gamma \log(p_t), \quad (3)$$

with $\gamma = 2.0$. The complete objective gives equal weight to each prediction head:

$$\mathcal{L} = \mathcal{L}_{\text{verb}} + \mathcal{L}_{\text{noun}} + \mathcal{L}_{\text{action}}. \quad (4)$$

We use `ignore_index=-1` for unseen action classes.

4.4 Optimisation

We train with AdamW [Loshchilov and Hutter, 2019] using $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, and weight decay 0.05. The baseline learning rate is 10^{-4} ; all LoRA variants use 5×10^{-5} . A linear warmup covers the first 5% of steps, followed by cosine decay to zero. We clip the gradient norm to 1.0 and train with a batch size of eight using `bfloat16` mixed precision.

Experiments run on two NVIDIA A100 80 GB GPUs with PyTorch 2.6 and PEFT 0.19.

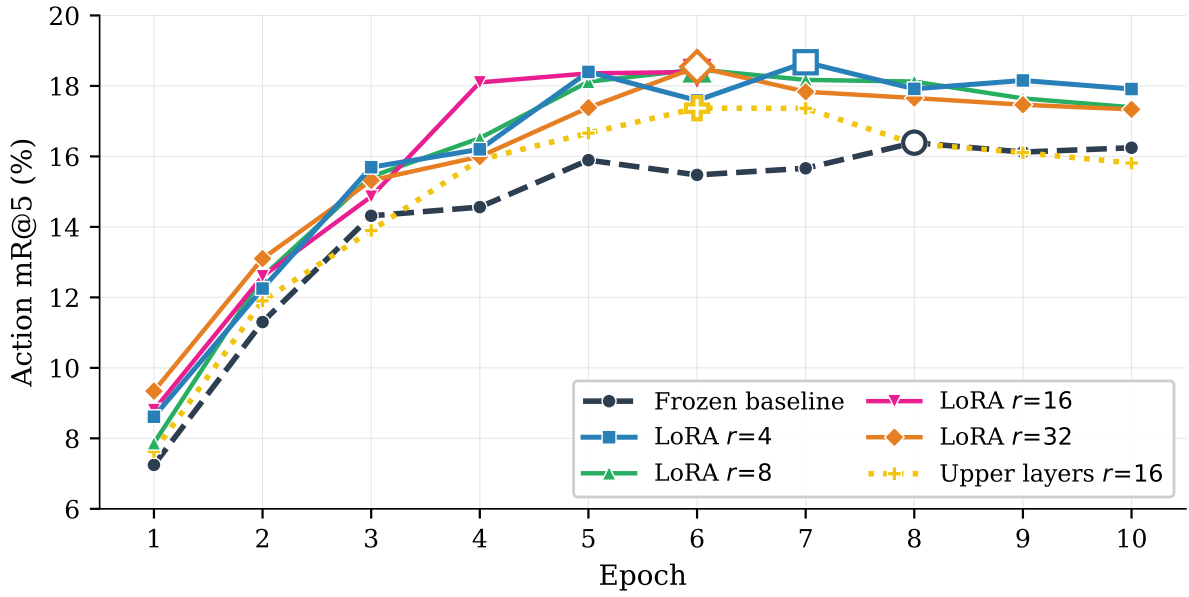


Figure 2: Validation action mR@5 over training epochs. Low-rank full-layer adaptation consistently improves on frozen probing. Increasing rank beyond $r = 4$ does not improve the best action score.

4.5 Evaluation Metrics

Our primary metric is mean-class Recall@5 (mR@5), following EPIC-KITCHENS evaluation practice. Mean-class recall suits the long-tailed label distribution because it weights each class equally instead of letting frequent actions dominate. We report verb, noun, and action mR@5 separately.

5 Results

5.1 Main Results

Table 2: Best validation results selected by action mean-class Recall@5 (mR@5). Adapter parameters exclude the shared 67.4M-parameter attentive probe. Bold denotes the best value in each metric column.

Configuration	Adapter Params.	Best Epoch	Verb mR@5	Noun mR@5	Action mR@5
Frozen baseline	0	8	27.50	29.84	16.39
LoRA $r = 4$	0.39M	7	28.79	32.64	18.67
LoRA $r = 8$	0.79M	6	26.41	33.26	18.47
LoRA $r = 16$	1.57M	7	26.46	32.02	18.44
LoRA $r = 32$	3.15M	6	27.04	33.13	18.54
LoRA upper ($r = 16$)	0.39M	6	25.67	31.51	17.37

Table 2 compares frozen probing against each LoRA configuration. All best-epoch selections are made by action mR@5, the primary evaluation metric. Every LoRA variant improves noun and action mR@5 over the frozen baseline. Full-layer LoRA with $r = 4$ achieves the best result, raising action mR@5 from 16.39% to 18.67% (+2.28 pp absolute, +13.9% relative) with 0.39M adapter parameters (0.13% of the backbone).

5.2 Rank Sensitivity

Figure 2 shows validation action mR@5 across epochs. The full-layer LoRA variants cluster tightly at their best checkpoints: 18.67% for $r = 4$, 18.47% for $r = 8$, 18.44% for $r = 16$, and 18.54% for $r = 32$. Increasing

rank gives no monotonic benefit. This suggests that small adapter matrices are sufficient for this downstream update. LoRA $r = 32$ peaks at epoch 6 and drops to 17.34% by epoch 10 while training loss keeps falling, indicating overfitting.

Action and noun mR@5 improve across all LoRA variants. Verb mR@5 is mixed: most variants score below the baseline (26.41–27.04 vs. 27.50), though LoRA $r = 4$ reaches 28.79. Verb-relevant motion features appear to be well captured by the frozen encoder; LoRA helps most with noun and action composition.

5.3 Matched-Budget Layer Placement

Full-layer LoRA with $r = 4$ and upper-layer-only LoRA with $r = 16$ each add 0.39M adapter parameters. Performance differs: distributing small adapters across all 24 layers yields 18.67% action mR@5, while adapting only layers 18–23 yields 17.37%. The 1.30 pp gap shows that broad layer coverage matters more than concentrating the same budget in the upper layers.

5.4 Efficiency

Table 3: Training efficiency over ten epochs. Hours are summed from epoch-level logs.

Configuration	Peak GB	Total Hours
Frozen baseline	13.80	21.99
LoRA $r = 4$	52.93	39.58
LoRA $r = 8$	53.24	34.84
LoRA $r = 16$	53.31	35.88
LoRA $r = 32$	53.17	46.96
LoRA upper	32.42	35.75

Table 3 clarifies that efficiency here means trainable parameter count, not lower compute or memory. Compared with frozen probing, LoRA raises peak memory from 13.8 GB to about 53 GB and total training time from 22.0 to 34.8–47.0 hours because gradients and activations must pass through adapted encoder layers. The upper-layer variant reduces peak memory to 32.4 GB by adapting only six layers. Thus LoRA’s benefit is a compact adapter and unchanged backbone weights, not faster or lighter training than frozen probing; the timing numbers are descriptive rather than a hardware benchmark.

6 Limitations

We train on a 22-participant subset of EPIC-KITCHENS-100, so the results reflect internal comparisons among adaptation strategies and cannot be compared directly to full-dataset benchmarks.

Relation to reported V-JEPA results. Our best action score (18.67% mR@5) is below the EPIC-KITCHENS-100 figures reported for V-JEPA 2 (39.7%) [Assran et al., 2025] and V-JEPA 2.1 (40.8%) [Mur-Labadia et al., 2026], but these are not like-for-like comparisons. We use 22 of 37 participants, compute mean-class recall over the 895 action classes present in our validation subset, and train a 0.39M-parameter LoRA module plus probe rather than reproducing the full benchmark recipe. The results should therefore be read as an internally controlled comparison of PEFT strategies, not as a re-benchmarking of V-JEPA.

Comparison to full fine-tuning. We do not evaluate full fine-tuning of V-JEPA 2, whose activation-memory cost exceeded our two-GPU budget. Future work should evaluate the full 37-participant split, compare against full fine-tuning, and measure variance across seeds and augmentation settings.

7 Conclusion

We applied parameter-efficient adaptation to V-JEPA 2 for egocentric action anticipation. Low-rank updates to the query and value projections improve action mR@5 over frozen probing across all tested ranks. Rank $r = 4$ performs best, adding 0.39M parameters and raising action mR@5 by 2.28 percentage points. Distributing adapters across the full encoder beats concentrating the same budget in the upper six layers. These results indicate that low-rank adaptation is viable for JEPA video representations and can steer V-JEPA 2 toward egocentric anticipation with few trainable parameters. The strongest gains occur on noun and action prediction, suggesting that the frozen feature-prediction representation already encodes useful pre-action structure and benefits from light task specialization.

Future work includes evaluation on the full 37-participant EPIC-KITCHENS-100 split, a controlled comparison against full fine-tuning, and joint adaptation of the predictor and encoder.

References

- [Assran et al., 2025] Assran, M., Bardes, A., Fan, D., Garrido, Q., Howes, R., et al. (2025). V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*.
- [Assran et al., 2023] Assran, M., Duval, Q., Misra, I., Bojanowski, P., Vincent, P., Rabbat, M., LeCun, Y., and Ballas, N. (2023). Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- [Bardes et al., 2024] Bardes, A., Garrido, Q., Ponce, J., Chen, X., Rabbat, M., LeCun, Y., Assran, M., and Ballas, N. (2024). Revisiting feature prediction for learning visual representations from video. In *European Conference on Computer Vision*.
- [Chen et al., 2020] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*.
- [Damen et al., 2018] Damen, D., Doughty, H., Farinella, G. M., Fidler, S., Furnari, A., et al. (2018). Scaling egocentric vision: The EPIC-KITCHENS dataset. *European Conference on Computer Vision*.
- [Damen et al., 2022] Damen, D., Doughty, H., Farinella, G. M., Furnari, A., Ma, J., et al. (2022). Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *International Journal of Computer Vision*.
- [Furnari and Farinella, 2019] Furnari, A. and Farinella, G. M. (2019). What would you expect? anticipating egocentric actions with rolling-unrolling LSTMs and modality attention. In *International Conference on Computer Vision*.
- [He et al., 2022] He, K., Chen, X., Xie, S., Li, Y., Dollar, P., and Girshick, R. (2022). Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- [He et al., 2020] He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- [Houlsby et al., 2019] Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., de Laroussilhe, Q., Gesmundo, A., Attariyan, M., and Gelly, S. (2019). Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*.

- [Hu et al., 2022] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2022). LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.
- [Kay et al., 2017] Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., et al. (2017). The Kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.
- [Li and Liang, 2021] Li, X. L. and Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. In *Annual Meeting of the Association for Computational Linguistics*.
- [Lin et al., 2017] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollar, P. (2017). Focal loss for dense object detection. In *International Conference on Computer Vision*.
- [Loshchilov and Hutter, 2019] Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *International Conference on Learning Representations*.
- [Mur-Labadia et al., 2026] Mur-Labadia, L., Muckley, M., Bar, A., Assran, M., Sinha, K., Rabbat, M., LeCun, Y., Ballas, N., and Bardes, A. (2026). V-JEPA 2.1: Unlocking dense features in video self-supervised learning. *arXiv preprint arXiv:2603.14482*.
- [Radford et al., 2021] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., et al. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*.
- [Tong et al., 2022] Tong, Z., Song, Y., Wang, J., and Wang, L. (2022). VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*.

IMVIP 2026

IRISH MACHINE VISION AND IMAGE
PROCESSING CONFERENCE



26th August – 28th August 2026

Maynooth University,
Maynooth, Co. Kildare, Ireland.