

Deep Learning Approaches for Robust Image-Based Localisation of Mobile Platforms

Eduardo Andres Avila Herrera

A dissertation submitted for the degree of

Doctor of Philosophy



Department of Computer Science

National University of Ireland Maynooth

Head of Department: Dr Aidan Mooney

Supervisors: Prof. Timothy McCarthy & Prof. John McDonald

Maynooth, Co. Kildare, Ireland

Abstract

Image-Based Localisation (IBL) provides a novel approach for estimating the geo-location of a mobile platform by matching a query image to a pre-collected catalogue of georeferenced scenes. This technique faces significant challenges, such as viewpoint variation and perceptual aliasing, particularly when matching scenes captured from orthogonal viewpoints. Previous research has demonstrated the use of image-based localisation to recover the geolocation of mobile platforms, primarily focusing on improving recall metrics [142, 54, 106, 124, 155, 38]. This thesis investigates the development, implementation, and improvement of an IBL approach, capable of operating in near-real time, for both aerial and ground level platforms. Combining neural networks, temporal filtering and efficient data structures, we develop a localisation technique that improves recall and robustness, which we validate on both mobile platforms.

This thesis makes three contributions. First, it introduces the Cross-View Shared Transformer (CVST) network, which improves recall and retrieval accuracy for ground-level platforms operating with both panoramic and limited field of view (FoV) images. Second, it provides a novel localisation technique that enables an Unmanned Aerial Vehicle (UAV) to operate in rural environments under restricted GPS conditions. Third, it enhances the robustness of cross-view localisation in ground-level platforms by integrating the CVST with a particle filter and a 3D KD-tree data structure to achieve near-real time performance. Further improvement in robustness is achieved through a mathematical formulation of the particle filter likelihood that incorporates knowledge of the road network, thereby improving the localisation accuracy of the system.

Acknowledgements

Reaching the end of this degree has been a long and demanding journey, one that would not have been possible without the support, patience, and encouragement of many people around me.

First and foremost, I would like to express my deepest gratitude to my wife, who knows this research inside and out, and to my parents and brother for their patience and support during this period. Their belief in me and constant motivation allowed me to keep going forward. Every chapter in this thesis is as much theirs as it is mine.

I am sincerely thankful to my supervisors, Prof. John McDonald and Prof. Tim McCarthy, for their insightful feedback, continuous guidance, and for always challenging me to think more deeply about my research. Their thoughtful comments and careful reviews shaped this work into what it has become.

I owe special thanks to my colleagues — Daire Walsh and Dr. Paul Lewis — for the countless discussions, code reviews, and insightful comments on my thesis.

I would also like to acknowledge Dr Richard Whooley, whose mentorship has been a steady source of inspiration and direction since my time in Phasefocus. Your advice extended far beyond my professional career and has had a lasting impact on how I think.

My thanks also go to Ann-Marie Burke and Sharon Comerford, whose support and patience have kept the group running smoothly. Your help behind the scenes has made a world of difference.

Finally, I want to extend a special thank you to my friends Dr Sean P. O’Kane and Dr Aidan Magee for their thoughtful review comments on this thesis, code reviews, and all

the motivational conversations. You've welcomed me to this country and made me feel at home.

I dedicate my publications to Rosa Jimenez.

To all of you — thank you for being part of this journey.

Contents

Abstract	ii
List of Publications	vi
List of Acronyms	vii
List of Figures	x
List of Tables	xv
1 Introduction	1
1.1 Background to Robotic Platforms and Localisation Methods	2
1.2 Contemporary Research	12
1.3 Gaps and Shortcomings	15
1.4 Research Questions	17
1.5 Aims and Objectives	20
1.6 Contributions	21
1.7 Thesis Structure	22
2 Image-based Localization State-of-the-Art	24
2.1 Introduction	24
2.2 Related Work and Foundational Concepts for IBL	29
2.3 Image-Based Localisation (IBL)	45
2.4 Ground View IBL	47
2.5 Aerial View IBL	55
2.6 Cross-View IBL	61

2.7	Discussion	78
2.8	Conclusion	81
3	Cross-View Image Based Localization with Transformer Network	84
3.1	Introduction	84
3.2	State of the Art	87
3.3	Methodology	91
3.4	Experimental Setup and Results	102
3.5	Discussion and Conclusion	106
4	Aerial Image Based Localization	108
4.1	Introduction	108
4.2	State of the Art	111
4.3	Methodology	114
4.4	Experimental Setup	119
4.5	Results and Discussion	122
4.6	Conclusion	126
5	Temporal Filtering in Image-Based Localisation	127
5.1	Introduction	127
5.2	State of The Art	129
5.3	Methodology	130
5.4	Results	139
5.5	Conclusion	150
6	Conclusion	151
6.1	Summary of Findings	151
6.2	Answering the Research Questions	153
6.3	Contributions	154
6.4	Limitations of IBL Methodology	155
6.5	Future Work	156
6.6	Conclusion	156

List of Publications

The following peer-reviewed publications arose from the research presented in this thesis:

1. E. A. Avila Herrera, T. McCarthy, and J. McDonald, “Cross View Shared Transformer (CVST): A Shared Weight Siamese Network for Limited FoV Image Based Localisation,” in *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*, 2024, pp. 1261–1268. (*Forms the basis of Chapter 3.*)
2. E. A. Avila Herrera, A. Magee, T. McCarthy, and J. McDonald, “Aerial Image Based Localisation via Learned Embeddings,” in *Irish Machine Vision and Image Processing Conference (IMVIP), IET Conference Proceedings*, vol. 2024, no. 10, 2024, pp. 291–298. (*Forms the basis of Chapter 4.*)

List of Acronyms

List of Technical Acronyms	
Acronym	Definition
ABT	Adaptive Belief Tree
AI	Artificial Intelligence
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
BoW	Bag of Words also found as Bag of visual words
CGAN	Conditional Adversarial Network
CNN	Convolutional Neural Network
CTONN	Cross Time Orientation Normalisation Network
CV	Computer Vision
CVM-Net	Cross-View Matching Network
DELG	Deep Local and Global Features
DS	Data Set
DSM	Dynamics Similarity Matching
ESA	European Space Agency
FAST	Features from Accelerated Segment Test
FCN	Fully Convolutional Network
FoV	Field of View
GAMa	Ground-video to Aerial-image Matching
GAN	Generative Adversarial Network
<i>Continued on next page</i>	

List of Technical Acronyms (continued)	
Acronym	Definition
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HCF	Hand Crafted Features
IBL	Image Based Localisation
ICP	Iterative Closest Point
IP	Internet Protocol
LAN	Local Area Network
LCM	Location Classification Methodology
LF	Learned Features
mAP	Mean Average Precision
MCMC	Markov Chain Monte Carlo
ML	Machine Learning
NASA	National Aeronautics and Space Administration
NLP	Natural Language Processing
ONN	Orientation Normalisation Network
ORB	Oriented FAST and Rotated BRIEF
OS	Operating System
PID	Proportional Integral and Derivative
POMDP	Partially Observable Markov Decision Process
PPT	Perspective Projection Transformation
ReLU	Rectified Linear Unit
RFID	Radio Frequency Identification
SAFA	Spatial-Aware Feature Aggregation
SfM	Structure from Motion
SIFT	Scale-Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
<i>Continued on next page</i>	

List of Technical Acronyms (continued)	
Acronym	Definition
SURF	Speeded up Robust Features
UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
USAM	Unit Subtraction Attention Module
VIO	Visual Inertial Odometry
VL	Visual Localisation
VPR	Visual Place Recognition
WAN	Wide Area Network
WSN	Wireless Sensor Networks

List of Figures

1.1	Examples of different robot configurations: Wheeled (top left) [132], Gating (top right) [17], Flying (bottom left) [98] and Marine (bottom right) [4] . . .	4
1.2	Quadrotor platform equipped with optical and LiDAR units	5
1.3	UGV platform equipped with optical and LiDAR units	7
1.4	Overall image-based localisation framework developed in this thesis. The offline stage (top) pre-encodes satellite and database imagery with the CVST encoder and stores the embeddings in a 3D KD-tree; the online stage (bottom) encodes the query image, and a particle filter queries the KD-tree and applies a road-network likelihood to estimate the platform pose. The research questions (RQ1-RQ5) are annotated on the stage each one addresses.	18
2.1	Images of Atlas (left) [107], Spot (top right) [17] and Ingenuity (bottom right) [98]	25
2.2	Different platforms capture data from different viewpoints. a) Satellite imagery viewpoint. b) Ground or horizontal viewpoint. c) UAV oblique downwards viewpoint. d) Aerial or overhead viewpoint.	30
2.3	Different platforms capture data from different viewpoints. a) Satellite imagery. b) Ground Vehicle horizontal viewpoint. c) Aerial downwards viewpoint. d) Aerial oblique downwards viewpoint. e) Ground oblique upwards viewpoint	31
2.4	Timeline of the most relevant datasets for IBL publicly available to date. .	33
2.5	Example of image matching features using the SIFT algorithm with Flann matching.	40

2.6	Voronoi cells demonstrating the cluster centre closed for a single cluster k . a) An ideal cluster that represents the two feature sets in blue and red. b) A cluster that does not represent the data in an optimal space. c) The adapted cluster centre fits the data in a geometry in which the difference between the residuals is maximum.	44
2.7	Framework for ground-level VL showing the general components needed to implement an IBL system.	46
2.8	A tree that maps IBL research by methodology, showing the evolution of the different algorithms.	78
3.1	Viewpoint of a ground-level image captured in Toronto.	85
3.2	Satellite image of Europe showing the location of Karlsruhe, Germany. Location where the dataset is collected.	92
3.3	Satellite images corresponding to the ground-level images collected in the KITTI dataset.	93
3.4	Road network of Karlsruhe, Germany.	94
3.5	Normally distributed samples collected over the Karlsruhe road network to generate negative satellite image pairs. In white satellite image samples from the KITTI Data Set locations. In black stochastic samples of satellite images.	95
3.6	The white markers show the collection locations of the KITTI data set. The black markers show normally distributed samples collected over the Karlsruhe road network to generate negative satellite image pairs.	96
3.7	The CVST Architecture employs a Siamese Network with identical branches represented in the image by a single branch network. Sec. 3.3.3 provides a detailed explanation of the architecture.	97
3.8	Example of mini batch of 8 triplets. The top row shows the ground level image also referred to as the anchor. The middle row of images represent the positive overhead matches. The bottom row of images show negative pairs. Data Attribution: Imagery 2023 CNES / Airbus, GeoBasis-DE/BKG, GeoContent, MaxarTechnologies, Maps Data: 2023 Google	100

3.9	Fraction of query ground level images for which the corresponding overhead view is in the top-K%. shared weights (sw), non-shared weights (nsw), Weighted Soft Margin Loss (sl), pretrained ImageNet-1K weights (^o)	102
3.10	Example of query ground level images. Highlighted in blue the ground truth pair and in green the correct match. Data Attribution: Imagery 2023 CNES / Airbus, GeoBasis-DE/BKG, GeoContent, MaxarTechnologies, Maps Data: 2023 Google	105
3.11	Visualisation of the ninth head attention map extracted from CVST. The attention map of each view is projected to each computed image.	106
4.1	Localisation Work Flow. In Blue, the offline process of encoding the satellite image collection. In red, the iterative process of encoding the live drone feed and the localisation algorithm.	110
4.2	Raster Image extracted	114
4.3	Sample grid locations shown in red and in white the random trajectory followed by the UAV	115
4.4	Recall results for different architectures extracted from the grid samples and test trajectories.	120
4.5	Recall metrics computed for the cross-time dataset.	120
4.6	Attention Maps Visualisation of aerial views at different time and lighting conditions.	121
4.8	Example of successful aerial images queries in a cross-time urban environment. The green box depicts the correct match and in blue the ground truth pair. Data Attribution: Imagery 2023 CNES / Airbus, GeoBasis-DE/BKG, GeoContent, MaxarTechnologies, Maps Data: 2023 Google.	123
4.7	Example of successful aerial images queries in a rural environment exposed to lighting variations. The green box depicts the correct match and in blue the ground truth pair. Data Attribution: Imagery 2023 CNES / Airbus, GeoBasis-DE/BKG, GeoContent, MaxarTechnologies, Maps Data: 2023 Google.	123

4.9	Examples of the IBL system successfully estimating the location of the UAV. The green particle shows the ground truth of the UAV, and the red show the top 10 position estimates.	124
4.10	Sample of frames from localising a drone over an extended trajectory in a rural environment. See https://youtu.be/p8IRbvpWZF0 for the complete sequence.	125
4.11	Examples of failure in estimating position due to ambiguous vegetation present in a rural scene. The green particle shows the ground truth of the UAV, and the red show the top 10 position estimates.	125
5.1	Workflow used in both the online and offline stages of the IBL System. In red, the online sequential algorithm encodes the ground level image, then the particle filter queries the KD-tree to retrieve the corresponding embeddings which are compared. Finally, the particle filter computes an estimate of the vehicles pose. In blue, the offline pre-encoding process employed to generate a 3D KD-tree structure.	131
5.2	Closest perpendicular distance of a particle to the road.	134
5.3	Visual representation of a 3D KD-tree. The x and y axis represent the coordinates of the satellite images, and the z axis represents the orientation of sampling.	136
5.4	Sampling methodology used to pre-process the KD-Tree databases. Samples captured over the road network are shown in black, while non-road network samples are shown in white.	139
5.5	Particle Filter converging to the correct position when implemented with VGG16.	141
5.6	Comparison of position error between the pure IBL system and the road-assisted IBL system.	142
5.7	Comparison of the standard deviation between the pure IBL system and the road-assisted IBL position.	142
5.8	Position-error comparison between the pure IBL system and the road-assisted IBL system, for the VGG16+SAFA network.	143

5.9	Standard-deviation comparison between the pure IBL system and the road-assisted IBL system, for the VGG16+SAFA network.	144
5.10	Particle Filter converging to the correct position when implemented with VGG16 + SAFA.	144
5.11	Position-error comparison between the pure IBL system and the road-assisted IBL system, for the CVST network.	145
5.12	Standard-deviation comparison between the pure IBL system and the road-assisted IBL system, for the CVST network.	145
5.13	Particle Filter converging to the correct position when implemented with CVST. Video of the CVST network applied to a pure IBL system with 1500 particles can be found in the link: https://youtu.be/_0pakpzS_Is . Video of the CVST network applied to a road assisted IBL system with 700 particles can be found in the link: https://youtu.be/ySoQ5QMKhbI? . . .	146
5.14	Comparison between architectures for the pure IBL system.	146
5.15	Performance of CVST at different numbers of particles.	148
5.16	Performance of CVST at different numbers of particles.	149
5.17	Comparison between architectures for the road-assisted IBL system. . . .	150

List of Tables

1.1	Summary of the feature-extraction approaches to image-based localisation discussed in this section, showing examples of representative works. These methods are reviewed in detail in Chapter 2.	14
1.2	Summary of the imaging-viewpoint and temporal-filtering characteristics of recent IBL systems, showing examples of representative works.	16
2.1	A table classifying the datasets up to date based on collection perspective and aimed purpose. Cross View (CV), Oblique aerial view (OAV), Multi view point (MV)	38
2.2	Classification of methodologies for ground-view IBL based on the captured viewpoint; also identifies the feature-extraction technique as handcrafted features (HCF) or learned features (LF).	48
2.3	Methodologies implemented for aerial VL, classified by viewpoint, with the feature-extraction technique identified as handcrafted features (HCF) or learned features (LF).	56
2.4	A representative selection of methodologies for cross-view localisation and their performance on the CVUSA dataset.	62
3.1	Results on the 5K test set. † denotes that the model was trained using weighted soft margin loss. ‡ represent Siamese networks that do not share weights between branches. ^o represents networks with weights pretrained on ImageNet-1K. * represents networks trained from scratch.	103

3.2	Results on the CVUSA test set. † denotes that the model was trained using soft margin loss. ‡ represent Siamese networks that do not share weights in between branches. ^o represents networks that have been pretrained with ImageNet-1K weights. * represents networks trained from scratch.	104
3.3	Network comparison of training parameters and inference time (I.T.) on a single NVidia RTX 3090 GPU containing 24 GB on board memory for fair comparison, percentages relative to CVST.	105
4.1	DJI Mavic 3 Enterprise Wide Camera Specifications	116
4.2	Results on the 5K test set. The symbol † denotes that the model was trained using soft margin loss. ^o represents networks that have been pretrained with ImageNet weights.	122
5.1	Near-real time performance of each network tested at different numbers of particles. The symbol † denotes that the model was trained using soft margin loss. ^o represents networks that have been pretrained with ImageNet weights. (RA) Road Assisted and (IBL) Pure IBL.	147

Chapter 1

Introduction

Image-Based Localisation (IBL), also known as **Visual Place Recognition (VPR)**, is the task of estimating the geographic position of a query image from a collection of georeferenced or geotagged images. Visual features are extracted from an image to enable comparison with catalogued images that contain known geolocation. These visual features can be extracted using different techniques, ranging from classic computer vision algorithms such as SIFT [86], FAST [109], or ORB [110] to the most recent methods that employ neural networks [72, 152, 80], where embeddings that can be compared in latent space are computed.

The main challenges faced by IBL are perceptual aliasing, viewpoint differences, seasonal/lighting variations, and computational inefficiencies. Perceptual aliasing is caused by architectural similarities across scenes in urban environments and by the feature sparsity in the repetitive scenery of rural environments, with a lack of distinct visual features. Geometric differences in viewpoints prevent visual features from being shared across images that capture the same scene. IBL relies on retrieving a positive match from an extensive catalogue of geotagged photos, where retrieval techniques can be computationally expensive.

IBL types are classified based on the viewpoints of the query image and the database. *This thesis focuses solely on aerial and cross-view IBL*, where each type presents unique challenges. Cross-view IBL is the task of geolocating a query image captured from a viewpoint that differs significantly—such as between ground-level and aerial imagery—from those in the reference database. When performing IBL, perceptual aliasing remains the

central challenge, resulting in considerable ambiguity in the matching process, and as a result, yields a high number of outliers in the pose estimates. The use of temporal filters, such as particle filters, works well in conjunction with neural networks and helps suppress the effect of such noise in cross-view scenarios. Particle filters can model the nonlinearities in the hidden distribution by incorporating odometry measurements into the process, thereby improving localisation [67, 55, 38]. However, the computational cost of particle filters can limit IBL’s speed, requiring alternative techniques to accelerate computations and enable near-real time operation.

This thesis bases the research on recent contributions to IBL systems [67, 54, 122, 55, 146, 155, 15, 79, 39]. These novel techniques implement different deep learning methodologies to geolocate a query image using a catalogue of georeferenced images as a map. *This thesis aims to improve upon these recent localisation systems by developing a neural network architecture capable of achieving higher recall rates and demonstrating its use on both aerial and ground-level platforms. In addition, improvements in the robustness of the localisation systems are achieved by incorporating a novel mathematical formulation that leverages knowledge of the road network to enhance particle filter estimation.* As an introduction, this thesis first provides background for the reader by describing the different mobile platform types commonly used in robotics and showing how distinct robot configurations affect the nature of localisation technologies. Second, the most common sensors required by mobile platforms to operate are introduced. Finally, the localisation problem is defined and the common localisation methodologies used in robotic systems are presented to introduce the reader to the Image-based localisation problem.

1.1 Background to Robotic Platforms and Localisation Methods

Unmanned vehicle technology has improved dramatically over the last few years, thanks to advances in sensors (LiDAR, IMUs, cameras) and novel developments in deep learning techniques. The potential to improve efficiency across various economic sectors by adopting autonomous vehicles has led to the rise of the robotics industry in areas such as construction

[70], mining [21] [117] [141] and transport. The ability of Unmanned Ground Vehicles (UGVs) and Unmanned Aerial Vehicles (UAVs) to navigate autonomously enables industries to perform logistics tasks with precision, safety, and efficiency. Autonomous vehicles are defined as platforms capable of performing complex tasks without human intervention, such as navigating in real-world environments. To achieve autonomy, robots must be programmed with complex perception algorithms that enable them to understand the environments in which they will operate. A key component for achieving autonomy is the ability to localise within an environment. Localisation is defined as the ability to identify a platform's pose with respect to a reference frame.

Localisation is a long-studied problem in the robotics and engineering field, with a wide variety of paradigms and techniques presented in the literature. Localisation techniques can vary based on the mobile platform configuration and the sensors integrated for perception. Before delving into the details of platform localisation, it is important to understand the differences across mobile platform configurations.

1.1.1 Mobile Platform Configurations

Mobile platforms may be categorised according to their operational environment, which determines the appropriate actuation mechanisms, sensing modalities, and localisation systems needed for accurate pose estimation. Robots can be broadly categorised into flying, wheeled, legged, marine or hybrid systems, as shown in Fig. 1.1.

Marine platforms may be grouped as either submergible or non-submergible. Platforms such as submarines fall in the submersible category. While floating platforms are non-submergible. Such platforms are not relevant to this thesis, where the main focus is on aerial and ground level platforms. Whilst the algorithms explored in this research are not expanded on beyond these platforms, they could become relevant to a wider set of platforms in the future.



Figure 1.1: Examples of different robot configurations: Wheeled (top left) [132], Gating (top right) [17], Flying (bottom left) [98] and Marine (bottom right) [4]

Ground and aerial robots can be classified as indoor or outdoor platforms based on their operational environments. The most common autonomous mobile platforms used in robotics are UGVs and UAVs. Among each type of robot, there are further subclassifications of UGVs and UAVs. UGV types can be divided into legged robots or wheel-based robots, while UAVs can be classified as fixed-wing platforms and multi-rotor platforms. Even though each type of robot is different, operates in distinct environments and offers different advantages, all of them share the need for localisation techniques to manoeuvre in a given environment.

Wheel-based UGVs are energy efficient and can withstand high load-carrying capacity for autonomous applications, making them desirable for industrial applications. UAVs also show high potential for autonomous aerial navigation and scene exploration, providing rapid means for the user to capture panoptic views. These systems have evolved to a point where they can navigate autonomously in unknown environments by utilising multi-modal

sensors, such as cameras, LiDAR, and navigation systems. These platforms regularly combine such sensors to digitally model and map their environment. A recently compelling example was shown by NASA where they demonstrated the ability of aerial autonomy with the deployment of the Ingenuity UAV for exploration on Mars [97].

1.1.2 Robotic Sensors

Sensors can be classified as exteroceptive or proprioceptive. Exteroceptive sensors measure the state of the external environment relative to the robot's current position; proprioceptive sensors measure the robot's internal state without interacting with the environment. The most common proprioceptive sensors are encoders, tachometers, gyroscopes, inertial measurement units (IMU), accelerometers, magnetometers and barometers, among many others. The most common exteroceptive sensors are optical sensors, laser scanners, radars, global navigation satellite systems (GNSS), sonar and ultrasound sensors, among others. All these sensors can be integrated in multiple combinations to collect rich multimodal data, allowing the robot to sense the environment and determine its position within it.

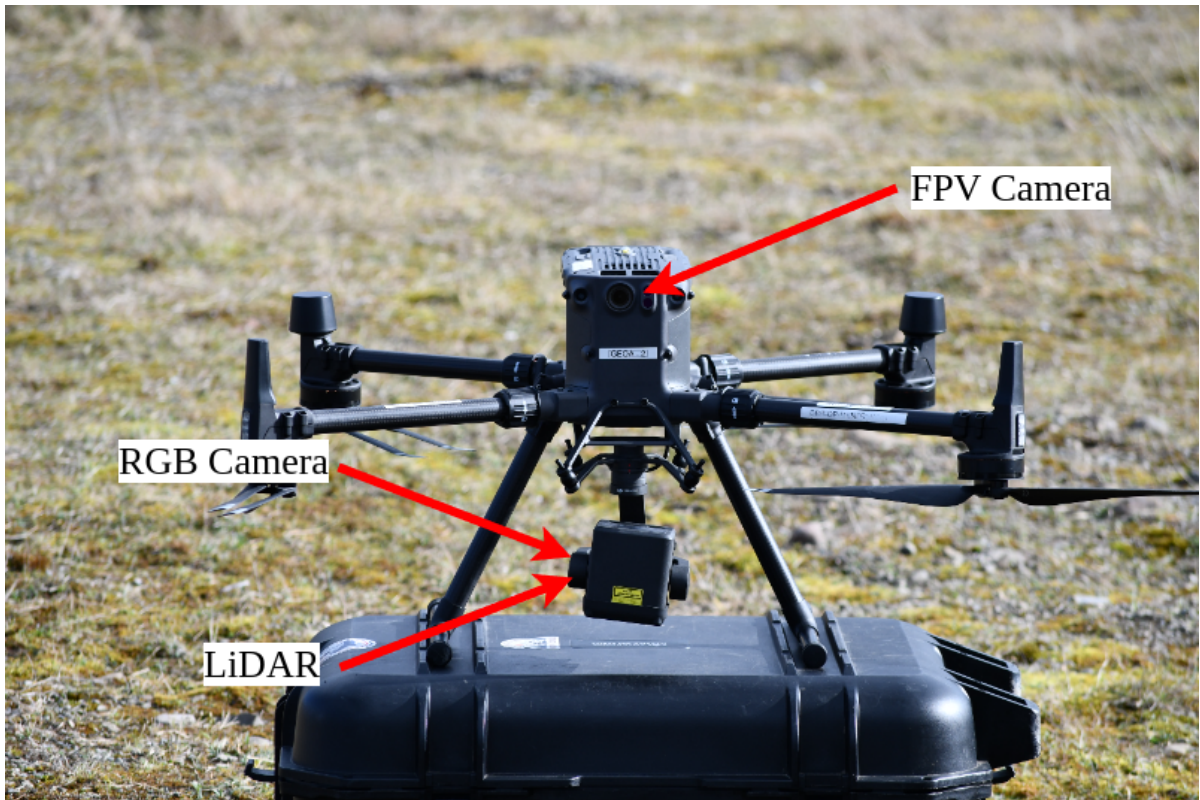


Figure 1.2: Quadrotor platform equipped with optical and LiDAR units

As an example, quadrotors normally compute and maintain an estimate of their pose from the following four sensors as a minimum set-up: a barometer to calculate altitude, GNSS to estimate global position, a magnetometer to measure heading and a gyroscope to estimate its frame orientation. As an example, a Quadrotor platform integrated with optical and LiDAR sensors is presented in Fig. 1.2. Each sensor has unique advantages and drawbacks, which need specific techniques to handle their noise. The barometer is susceptible to light intensity [99] and atmospheric pressure changes, which induce noise in its measurements. GNSS can be noisy in urban areas due to issues such as multipath effect in urban canyons, whereas magnetometer requires a drift correction [151].

UGVs are normally equipped with a similar set of sensors as aerial platforms. However, the mounting direction of their sensors significantly affects the viewpoint from which optical and LiDAR sensors collect data. The UGV’s optical and LiDAR sensors are mounted parallel to their operational plane, while UAVs mount these sensors orthogonal to the surface of the Earth. Additionally, larger UGVs can handle larger and heavier LiDAR units and cameras with larger lenses. An image illustrating the different sensors integrated to a UGV platform is provided in Fig. 1.3.

The large variety of available sensors has created a complex robotics design landscape involving different algorithms, electronic modules, power control systems, among many other factors. To improve robotic perception, researchers have developed a wide range of computer vision techniques that interpret data from optical sensors, typically tailored for particular tasks [116]. For instance, a calibrated optical sensor can be utilised to measure angular rates, distances, and, more recently, for localisation methods, among others [23, 67, 15, 6]. The versatility offered by optical sensors enables an additional level of redundancy, enhancing the robustness of platforms. Nonetheless, optical sensors do not replace and instead often complement specialised task sensors such as laser scanners or gyroscopes.

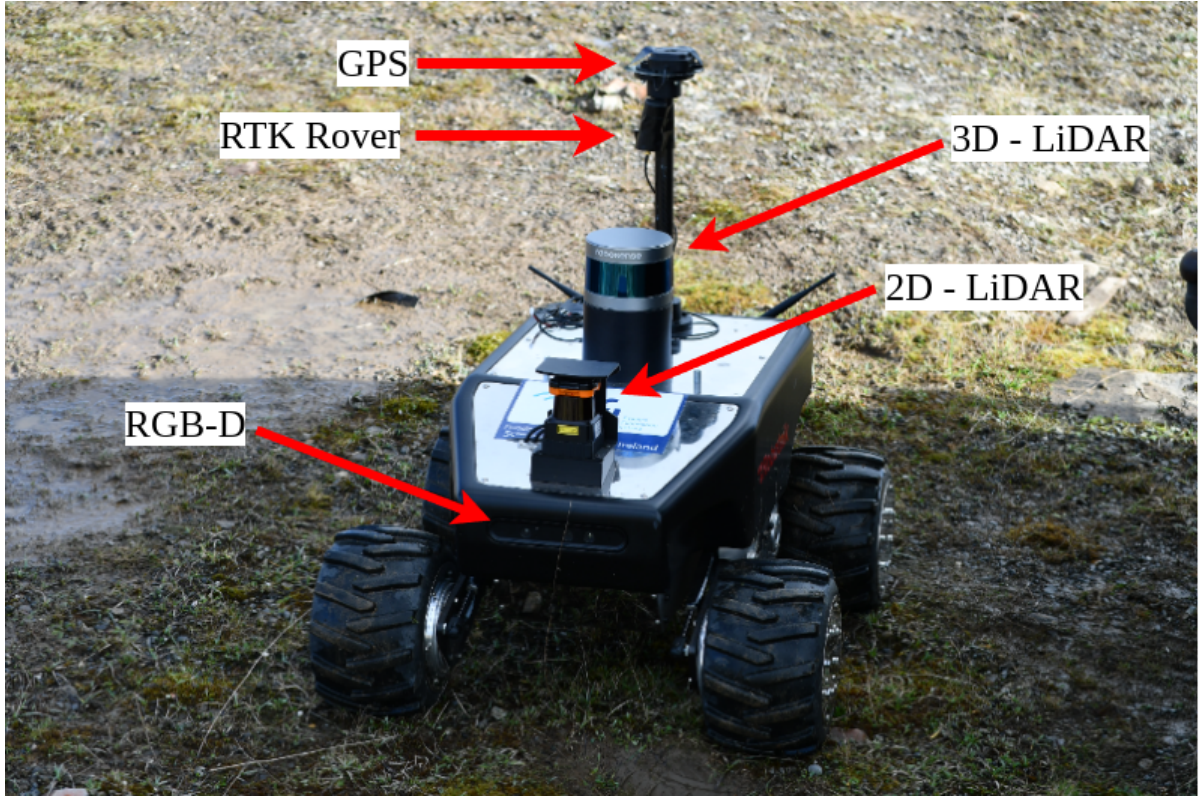


Figure 1.3: UGV platform equipped with optical and LiDAR units

1.1.3 Localisation Methods

Just as robotic platforms can be classified as ground, aerial, and marine platforms, localisation systems can be organised into similar categories. Localisation methods can vary depending on the environment and context within which the mobile platform will operate. Identifying the challenges and the physics of the operating environment is crucial for selecting a suitable localisation system. Furthermore, the type of mobile platform will limit the localisation technique employed. There are several localisation methods available today; the most common are the Global Navigation Satellite System (GNSS), Signal Strength-Based Localisation [11, 68, 46, 56], Dead Reckoning [42], Simultaneous Localisation and Mapping (SLAM) [159, 121, 96, 149, 141], Visual Inertial Odometry (VIO) [116], and Bayesian Filters [29, 129, 40], and Visual Place Recognition [152, 142, 67, 122, 106, 124, 123, 15, 155, 166, 164, 79, 38, 39].

The most common localisation system employed by outdoor platforms at present is GNSS, whereas indoor mobile platforms utilise Dead Reckoning [42], electronic beacons [18], vicon systems [91, 149], visual representations of the environment [60] paired with SLAM

algorithms [159, 121, 96, 149, 141] or VIO [116]. A brief description of the most prevalent techniques currently available is presented.

1.1.3.1 Global Navigation Satellite Systems (GNSS)

As mentioned earlier, the most widely used approaches to solving localisation are based on GNSS. Major players in the spatial industry have deployed satellite constellations, resulting in proprietary positioning systems such as Galileo in the EU, the Global Positioning System (GPS) in the USA, BeiDou in China, and the Global Navigation Satellite System (GLONASS) in Russia, along with other systems developed by other nations.

These systems allow a direct measurement of a platform's position in an inexpensive way. Also, being highly accessible, GNSS solutions are popular and easy to integrate into robots. Modern devices such as smartphones and cameras integrate chips capable of simultaneously connecting to as many as 17 satellites, enhancing the accuracy and robustness of position estimation through satellite signal trilateration. Despite its advantages, this technology has several weaknesses, such as its susceptibility to noise caused by interference, weather conditions, signal jamming, and multi-path effects due to the satellite signal reflecting off certain materials [19]. Furthermore, GNSS signals are vulnerable to spoofing, which involves the hacking of the signal, leading to a loss of positional accuracy for the platforms [126].

Various methodologies have been implemented in the past to enhance the accuracy and reliability of GNSS systems. Alternative solutions on the market are Differential Global Positioning System (DGPS), Post Processing Kinematic (PPK) and Real Time Kinematic (RTK). DGPS augments standard GPS systems with a network of ground-based reference stations with known fixed positions. The use of DGPS depends on the network broadcasting the deltas between their accurate positions and the noisy satellite-computed positions. Land-based DGPS phased out or superseded by newer technology in the USA, UK and Ireland. PPK is a position correction methodology applied post-mission, making it unsuitable for real-time applications. PPK is a position correction methodology in which GNSS readings are collected during the mission, and later corrected using data gathered from the nearest

ground base station. Due to GNSS readings being time dependant, the mission data can be accurately adjusted against the base station readings collected at the time of the mission, permitting precise position correction data in a post-processing step. Since PPK is applied after the mission, this constrains near-real time applications. RTK consists of at least one calibrated base station that collects satellite readings over time and computes an accurate position. Based on the corrected location, it provides any platform connected to the station with precise position measurements. RTK positioning requires the GNSS base station to maintain direct communication with the rover RTK module at all times, making the distance between the base and the rover crucial. All the solutions presented depend on a ground station close to the mobile vehicle for implementation. Furthermore, these solutions can significantly impact the overall cost and complexity of the system.

1.1.3.2 Signal Strength Based Localisation

Localisation systems based on measuring a device's signal strength are commonly seen in environments where a network of transceivers producing radio-frequency electromagnetic fields is available. A great example is the cellular network 4G or the Ultra-Wideband transceivers Decawave DWM1001. This technique relies on measuring the signal strength transmitted by the mobile device with respect to a minimum of three fixed transceivers. Based on the signal strength from each of the fixed transceivers and the specifications of the wave used for communication, the position of the device can be trilaterated [11, 68, 56]. This technique is common in indoor applications; however, it relies on the network always being available and precise positioning of the fixed infrastructure. Additional challenges are present such as estimating the electromagnetic noise induced into the signal and the multipath effect caused by reflections, similar to the GNSS case presented previously.

1.1.3.3 Dead Reckoning

Dead Reckoning utilises a history of velocity measurements along with the sampling times to maintain an estimate of the platform's position over time [42]. Given the noise associated with such sensors, the velocity estimates can be obtained from individual wheel encoders, a combination of tachometers and gyroscopes, among other sensor integrations. The use of

this localisation methodology as a main position estimator results in large trajectory drift. All velocity measurements sensed are susceptible to disturbances, hence sensors operate between a range of errors. These errors induced into the position estimates accumulated over time, causing the position estimate to drift away from the ground truth. In addition to the sensor error, the actuator also induces errors caused by mechanism backlash, wheel slip or stochastic events during motion adding to the drift error. This technique is normally implemented in combination with Bayesian filters to counteract the drift error.

1.1.3.4 Bayesian Filters

To overcome the challenges presented by GNSS, signal strength based methods and Dead Reckoning, mobile platforms typically combine both proprioceptive and exteroceptive sensors to estimate the current position based on prior measurements relative to a reference location in the unknown environment. These methods are commonly referred to in the literature as sensor fusion techniques [41], which typically utilise Kalman filters, particle filters [40] or smoothing to combine the outputs of different sensor technologies to estimate the platform's pose. Kalman and particle filters are an iterative process that rely on multiple iterations for the system to smooth the measurement error when predicting a more accurate estimate of position. This thesis focuses on the implementation of particle filters, which can model non-linear and non-Gaussian distributions by using stochastic sampling methods. Despite this ability, particle filters are computationally expensive algorithms that require a large number of iterations and are additionally sensitive to calibration procedures. Chapter 2 and Chapter 5 present the mathematical formulation of the algorithm and detail its application in the context of this thesis, respectively.

1.1.3.5 Visual Inertial Odometry (VIO)

VIO is a technique that leverages the optical and inertial sensors to estimate a platform's pose. By extracting invariant features from images and tracking them across successive frames, VIO fuses the visual information with the inertial measurements to compute an accurate and robust position estimate [116]. The most common image features extraction algorithms are SIFT [86, 65], SURF [13] and ORB [110], among many others. Two

configurations can be used when applying VIO, mono and stereo cameras. When a single optical sensor is used, the features extracted at time t ($\mathcal{F}_t$) are compared to the features extracted at time $t - 1$ ($\mathcal{F}_{t-1}$) to compute an estimate of odometry based on the induced visual motion of the features within the image frame and calibration of the sensor. Depending on the approach taken, this allows the estimation of an associated essential or fundamental matrix relating the two views, from which a linear and rotational velocity can be calculated. The stereo camera configuration leverages the ability to extract features from two sensors separated by a known rigid transformation. This allows for the use of epipolar geometry to calculate a depth estimate of the scene. The depth estimate paired with the invariant features is used to estimate the odometry of the sensor. To integrate and fuse the Visual odometry with the IMU data, an optimisation process such as bundle adjustment or a Multi State Kalman filter is implemented as a tightly coupled system. The combination of high-rate inertial estimations with visual measurements yields the robust position predictions of VIO.

1.1.3.6 Simultaneous Localisation and Mapping (SLAM)

Simultaneous Localisation and Mapping (SLAM) is a methodology that employs the perception of a robot navigating an environment to build a map while keeping track of the robot's location in reference to the constructed map. SLAM has become a popular solution for autonomous vehicles, as it allows it dispenses with the need for an a-priori, ther by permitin operation in a new unknown environment [21, 159, 96, 23]. There are many variants of SLAM algorithms which use LiDAR [159, 121], cameras or a combination of sensors [21, 96, 23] to map. When optical sensors are used, the technique is called visual SLAM (VSLAM) [96, 23]. SLAM technologies have improved the autonomy of robotic platforms by enabling robots to build local map representations that facilitate navigation tasks.

1.1.4 Localisation in Collaborative Robotics

The unique advantages and disadvantages offered by different platform configurations make collaboration between UGVs and UAVs desirable, as different platforms complement one

another. UGVs have a constrained field of view and can take considerable time to explore large areas. However, they are energy-efficient and possess a high carrying capacity. On the other hand, UAVs can cover large areas in less time, but their energy efficiency and dependency on weather conditions limit their operational time. Current research is pushing the boundaries of machine autonomy by combining both technologies into one combined heterogeneous autonomous systems. The integration of both platforms can enhance the mapping and localisation capabilities of the robots by leveraging the viewpoint of each platform [31, 74, 47].

Collaborative robotics requires the interaction between multiple platforms through sharing individual locations, local maps as well as the ability to physically interact in between them. The test of collaborative systems is commonly done under controlled indoor environments due to the ability to control the scene, as is the case of [30, 31, 74, 92]. However, with the new technological advancements and the popularity of robotic systems that can operate robustly in outdoor conditions, the implementation of collaborative robots in outdoor scenarios presents various challenges to be reliable in real world conditions. Dynamic scenes, changes in lighting conditions and drastic weather fluctuations affect negatively the localisation technologies and systems require new methods to account for these challenges. One of the main challenges that autonomous systems face when operating outdoors is maintaining robust and continuous localisation.

1.2 Contemporary Research

The rapidly evolving robotics industry demands robust localisation technologies for effective planning and navigation of mobile platforms. Recent research has demonstrated a more focused interest in robot collaboration in order to solve the localisation problem between platforms navigating in a shared physical environment. Combining both aerial and ground vehicles into one autonomous system offers synergistic advantages, thereby enhancing the perception of the environment with aerial and horizontal views. However, robotic collaborative tasks are typically deployed indoors and confined to local navigation references, such as ArUco markers [108] or local radio frequency systems [68, 46, 56]; Platforms that

operate outdoors heavily depend on GNSS for position estimation.

With transportation and navigation technology advancing, it is more common to see new robotic platforms collaborating with each other. As an example, researchers explore the potential for smart cities to collaborate with autonomous vehicles and for those vehicles to work alongside nearby robots [71, 154]. For many years, collaborative robotics has explored various methods for platform localisation, yet requiring more safe and robust approaches; these include employing visual tags, electronic beacons, Simultaneous Localisation and Mapping (SLAM), human-in-the-loop checkpoints or shared maps to localise the platforms, among other solutions [30, 31, 74, 47, 92, 100, 95]. This demand for new localisation technologies motivates the development of alternative solutions such as IBL.

This thesis focuses on the development, implementation, and improvement of IBL approaches capable of operating in near-real time for both aerial and ground-level platforms. Combining transformer based architectures, temporal filtering and efficient data structures, a localisation technique that improves recall and robustness is developed, which is validated on both mobile platforms. This thesis also investigates a novel particle filter algorithm that integrates knowledge of the road network to enhance the robustness of the estimated position. Recent IBL research focuses on techniques that employ deep learning approaches to enable the comparison of the query image in latent space [67, 54, 122, 106, 124, 123, 155, 79, 155, 164, 38, 39]. IBL feature-extraction methods have evolved from CNN-based systems, through Generative Adversarial Networks (GANs) and systems that perform geometric transformations of the query image prior to encoding it, to more recent Transformer-based methods. Table 1.1 summarises these approaches together with representative works that adopt each one. The most relevant contributions that influence the results of this theses are methods that incorporate IBL with limited FoV images and that integrate temporal filtering to the workflow. Kim *et al.* [67] present the first cross-view IBL system that works on sequential images in an urban scenario, where the UGV’s ground level image is localised against an aerial satellite map. The images are compared via deep learning techniques to compute a similarity metric used by a particle filter to localise the vehicle. Hu *et al.* [54] presents a system performing city level IBL employing 360° panoramic images and deep learning techniques improving

the recall metrics in cross-view IBL. In a subsequent work the authors demonstrate the integration of the architecture with a particle filter [55]. Bianchi *et al.* [15] demonstrates a teach and repeat approach operating a UAV in near-real time. Here, the quadrotor is capable of returning home based on a pre-encoded database of satellite images that allow the live camera feed to be compared in latent space. Wang *et al.* [146] presents an extensive study demonstrating improvements in recall metrics for both aerial and cross-view IBL by sectioning learned embeddings based on image regions. City wide IBL is demonstrated by Downes *et al.* [39], who presents successful near-real time operation of cross-view IBL for limited FoV images. Each of the examples presented, among many others, demonstrates the robotics community’s interest in exploring the possibility of mobile platforms navigating using visual features, just as humans and animals do. All the approaches presented are described in detail in Chapter 2 of the thesis, where the reader can find the complete state-of-the-art collected.

Feature-extraction approaches to image-based localisation		
Approach	Characteristic	Representative works
CNN-based	Convolutional encoders compared in latent space	[152, 81, 142, 67, 54, 122, 130, 55, 124, 123, 15, 79, 38, 146, 147, 33, 39]
GAN-based	Synthesise cross-view appearance to aid matching	[106]
Geometric transformation	Warp the query to the reference domain before encoding	[123, 79]
Transformer-based	Attention-based cross-view embedding	[155, 165, 164, 168, 160, 10]

Table 1.1: Summary of the feature-extraction approaches to image-based localisation discussed in this section, showing examples of representative works. These methods are reviewed in detail in Chapter 2.

To apply these deep learning techniques for localisation, data sets of georeferenced ground and aerial imagery collected simultaneously are needed to train the neural network models. Like any other process, the training results are highly dependent on the quality of the inputs, in this case the collected data. To operate deep learning models at high accuracy, data sets should be collected with RTK technology, which reduces location noise. Various groups around the world have collected data sets around different locations, such as KITTI [43], Oxford [88] [89], Four Seasons [150]. The history of datasets introduced is presented

in Chapter 2, showing their evolution.

Research has found that capturing different cross-time data in the same location is key for good performance in deep learning models [63]. As the physical scene changes over time, environmental conditions vary across samples, introducing different types of noise into the system. Collecting diverse scenarios enriches the data used to train deep learning models, resulting in more robust models that are less susceptible to noise. This allows robots to handle real world scenarios more robustly.

IBL systems are modular systems comprising a feature extraction process, a sampling technique, an image retrieval mechanism, and a temporal filter to mitigate noise. Each of these modules offers unique characteristics that allow the IBL system to operate. By methodically analysing each component of the IBL workflow, robustness and accuracy can be improved. Ablation studies help to identify architectures more suitable for IBL where improvements over the temporal filters help mitigate the perceptual aliasing noise and implementation of efficient data structures enable the systems to operate in near-real time.

IBL has the potential to become an alternative technique to GNSS methods. Not depending on external networks for localisation, IBL reduces spoofing and multipath effects, which are the primary drawbacks of GNSS technology. IBL has proven to be suitable for indoor environments and, more importantly, for outdoor operations [67, 54, 55, 15, 155, 164, 39].

Chapter 2 presents an extensive state-of-the-art review of image-based localisation and visual place recognition. Nonetheless, the previous examples demonstrate how IBL can be utilised to mitigate the disadvantages present in localisation systems discussed in Section 1.1.3.

1.3 Gaps and Shortcomings

A number of shortcomings in contemporary robot navigation research have been identified in this study. Mobile platforms that only rely on GNSS systems are susceptible to signal loss, jamming, multi path effects [19] or spoofing attacks [126]. The easy access

to satellite images and street view catalogues makes IBL systems an attractive alternative/complementary solution for global localisation systems to help mitigate the downsides of GNSS. Further developing IBL will enable mobile platforms to operate with an additional layer of redundancy for position estimation.

Although impressive results have been presented by recent IBL research [142, 67, 105, 7, 54, 122, 106, 55, 155, 79, 38, 147, 39], all the systems report being susceptible to perceptual aliasing, lighting conditions, seasonal variations and scene ambiguity. To address this, the majority of systems implement IBL with panoramic images, which are known to improve the recall metrics, while few systems test the techniques with limited field-of-view (FoV) images. Additionally, when systems attempt to localise a ground-level platform based on image sequences, a temporal filter is required regardless of the use of panoramic images. Table 1.2 summarises these viewpoint and filtering characteristics together with the works that exhibit them. The challenges presented by limited FoV leave room to explore further methodologies that improve the recall and robustness when employing limited FoV cross-view IBL.

Imaging viewpoint and temporal-filtering characteristics of IBL systems		
Characteristic	Note	Works
Panoramic imagery	Higher recall, but requires specialised 360° capture	[152, 122, 106, 123, 124, 55, 166, 79, 38, 147, 33, 77, 34]
Limited field-of-view imagery	More challenging, lower recall; few systems address it	[142, 67, 105, 39]
Recall improvement (panoramic)	Recall gains attributed to panoramic views	[142, 79, 155, 33, 147, 77]
Temporal filtering (image sequences)	Required to localise ground-level image sequences	[67, 55, 38, 39]

Table 1.2: Summary of the imaging-viewpoint and temporal-filtering characteristics of recent IBL systems, showing examples of representative works.

Systems that integrate Bayesian filters, such as particle filters, to mitigate noise caused by appearance ambiguity introduce computational inefficiencies into the overall IBL system [67, 55, 39]. Even the most efficient techniques require vast trajectories to converge to a correct position estimate [39], leaving room for improvement in the implementation of particle filter based approaches.

Current aerial localisation methodologies explore improving the recall for IBL aerial

localisation and test the UAV capabilities in “return to home” [15] tasks. However, this thesis identifies a gap between “return to home” tasks and solving the kidnapped-robot problem. To successfully achieve aerial IBL, the robot must be able to determine its location from an image captured at its instantaneous position, regardless of lighting conditions and topographic features. Current aerial systems do not address the kidnapped-robot problem for IBL-based aerial platforms.

Improving the robustness of IBL systems is essential in the further development and deployment of the technology. Perceptual aliasing is one of the main challenges in IBL, motivating research for further techniques that integrate additional information to mitigate this noise. Perceptual aliasing and recall noise can also be attributed to lighting variations during deployment, making datasets that incorporate lighting conditions essential for training models that generalise across lighting variations and improve system robustness. The IBL system should be robust and mitigate perceptual aliasing, changes in lighting conditions, scene dynamics across time and sparse visual features. The fundamental challenge for IBL in all platform types lies in learning discriminative visual representations that are robust to ambiguous visual patterns, generalisable across geographic scenes and resilient to photometric variations.

The purpose of the thesis is to improve IBL technology for both aerial and ground platforms. The work led by Kim, Hu, Bianchi and Downes *et al.* [67, 54, 55, 15, 39] are key foundations in this research.

1.4 Research Questions

The localisation approach developed in this thesis follows the overall framework shown in Fig. 1.4. It comprises an offline pre-processing stage, which encodes the satellite and database imagery with the Cross-View Shared Transformer (CVST) and stores the resulting embeddings in a 3D KD-tree, and an online stage, in which the query image is encoded and a particle filter queries the KD-tree and applies a road-network likelihood to estimate the platform pose in near-real time. Each of the research questions addressed below targets a specific stage of this framework, as annotated in the figure.

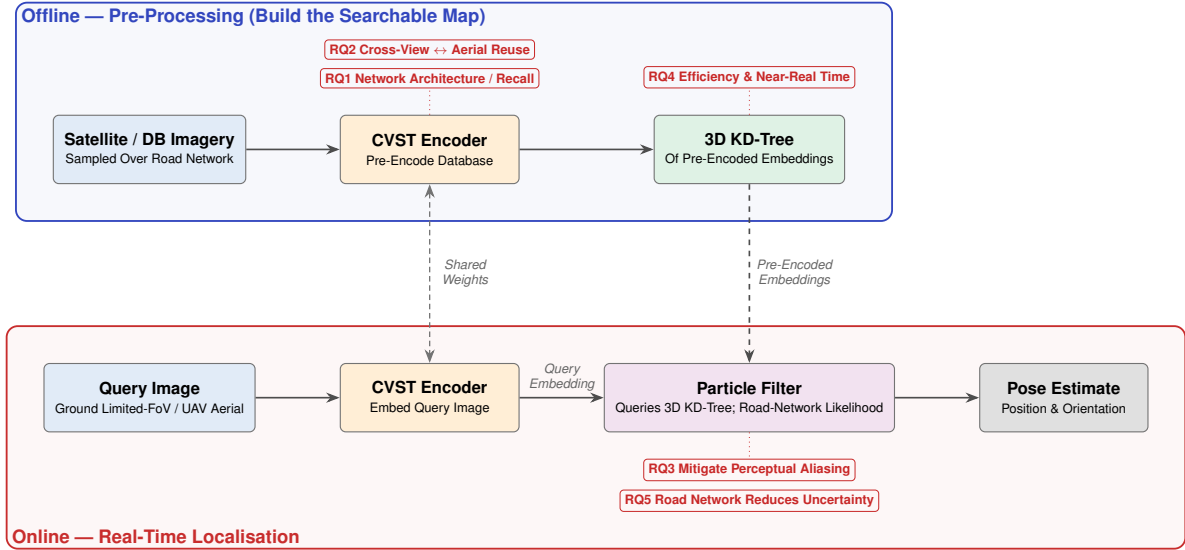


Figure 1.4: Overall image-based localisation framework developed in this thesis. The offline stage (top) pre-encodes satellite and database imagery with the CVST encoder and stores the embeddings in a 3D KD-tree; the online stage (bottom) encodes the query image, and a particle filter queries the KD-tree and applies a road-network likelihood to estimate the platform pose. The research questions (RQ1-RQ5) are annotated on the stage each one addresses.

This thesis is focused on addressing the following key research questions (RQ1–RQ5):

- **RQ1: What neural network architectures can improve recall for ground-level cross-view IBL?**

The use of panoramic images allows robotic systems to capture feature-rich images that work well when compared against extreme viewpoint images, such as satellite imagery [152, 55, 38]. In contrast, limited-field-of-view (FoV) images exhibit a significant drop in recall compared with those attained with panoramic images [142, 67]. The large difference in recall metrics makes cross-view IBL much more challenging [67, 39] as the system becomes more susceptible to perceptual aliasing. To improve performance in cross-view IBL, it is therefore essential to develop neural networks capable of handling comparisons across extreme viewpoints.

- **RQ2: Can neural network architectures be interchanged between cross-view and aerial IBL?**

UAVs are required to operate in both urban and rural environments under GNSS-denied conditions. The deployment of aerial platforms in real-world scenarios necessitates testing on cross-time datasets that capture lighting variations and seasonal

changes. UAVs are also susceptible to perceptual aliasing, especially in urban environments that lack distinct geometric landmarks. Such datasets are scarce, and few systems have been tested under kidnapped robot conditions [36, 15, 135], where variations in illumination and temporal changes are present.

- **RQ3: What techniques can be implemented to mitigate perceptual aliasing in IBL?**

Cross-View IBL is highly susceptible to perceptual aliasing, which introduces noise in the position estimates. To mitigate such noise, temporal filtering can be employed to improve the position estimate [67, 55, 39]. Temporal filters, such as particle filters, need a proposed probability density function (p.d.f.) to approximate the underlying non-linear distribution. Estimation is then performed through an iterative process that evaluates possible solutions and rejects those that are non-viable. The p.d.f. serves as a means for quantifying and measuring the quality of the proposed solutions. To reduce the uncertainty in the estimate, it is necessary to improve the formulation that quantifies the proposed solutions by integrating additional measurements.

- **RQ4: How can particle filters be integrated with deep learning techniques to mitigate visual feature ambiguity while remaining computationally efficient?**

Particle filters are computationally expensive algorithms that require a large number of particles to model the multimodal, non-linear distribution. In IBL applications, a particular challenge arises as each particle requires a separate inference of the neural network [67, 55], making near-real time operation infeasible with the current hardware. This makes it impossible to run the process in real time with the current hardware available. Precomputation of embeddings eliminates this bottleneck by removing the need for per-particle inference [15, 10, 39], reducing it to a search task. Integration of particle filters with efficient data structures allows for near-real time operation. To date, only two IBL systems have leveraged grid presampling structures and KD-trees [15, 39] to enable near-real time performance.

- **RQ5: Can additional estimation features, such as the road network, reduce position uncertainty for cross-view IBL systems?**

As mentioned in Section 1.3, all systems are susceptible to errors in position estimate caused by the presence of visual ambiguity when comparing images. Recent research has improved both particle filter formulations [67, 55, 39] and neural network architectures [142, 67, 122, 123, 106]. However, few approaches have leveraged knowledge of the road network topography to enhance localisation accuracy [53, 57].

1.5 Aims and Objectives

This research aims to explore the applications of image-to-image localisation in the field of robotics and proposes improvements over existing methods. This thesis presents both a UAV localisation system and a UGV cross-view framework that advance previous state-of-the-art through the implementation of a novel neural network (CVST). To answer the research questions formulated above, this thesis presents IBL systems with the following novel implementations that:

1. Improvements in image recall by introducing and comparing a novel Cross View Shared Transformer (CVST) network.
2. Present a thorough ablation study between state-of-the-art IBL architectures in limited field of view images.
3. Demonstrate a complete IBL system with the use of CVST for UAVs operating in urban and rural areas with light variations.
4. Improve the robustness of cross-view IBL for limited field of view images, introducing a novel likelihood formulation implemented in the particle filter
5. Present a complete IBL framework capable of performing in near-real time with integrated deep learning and temporal filters.

1.6 Contributions

The main contributions of this thesis are summarised below, each cross-referenced to the chapter in which it is developed and evaluated:

1. **Cross-View Shared Transformer (CVST):** a novel shared-weight transformer architecture for image-based localisation that compares limited field-of-view ground-level images against extreme viewpoints, improving retrieval recall over existing architectures for both cross-view and aerial views (Chapter 3; addresses RQ1 and RQ2).
2. **Aerial image-based localisation in urban and rural environments:** a complete framework that localises UAVs operating under real-world conditions, including cross-time and seasonal variation, by reusing the CVST embeddings; it is demonstrated on a purpose-captured rural dataset acquired in Ireland and on an urban cross-time satellite dataset (Chapter 4; addresses RQ2 and RQ3).
3. **Robust near-real time cross-view IBL:** a mathematical formulation that incorporates prior knowledge of the road network into the particle-filter likelihood to overcome perceptual aliasing, combined with a pre-encoded 3D KD-tree that enables near-real time operation (Chapter 5; addresses RQ3, RQ4 and RQ5).

These contributions are unified by three research themes that recur throughout the thesis:

- **Limited field-of-view localisation:** achieving reliable IBL from narrow field-of-view imagery rather than panoramic views.
- **Near-real time operation:** keeping the complete localisation pipeline computationally tractable through pre-encoded embeddings and efficient data structures.
- **Perceptual-aliasing mitigation:** improving robustness by combining learned visual features with temporal filtering and road-network priors.

1.7 Thesis Structure

Chapter 1 provides background information about localisation systems and mobile robotics platforms. It then outlines the motivation for investigating the specific topics addressed in this thesis, followed by a detailed statement of the research problems. Finally, the main research questions of the thesis are presented, followed by the overall thesis structure.

Chapter 2 reviews the current state-of-the-art techniques on image-based localisation. It presents a detailed analysis of research in visual place recognition, image-based localisation and the theoretical foundation for particle filters. The history of image-based localisation is discussed, and an overview of datasets introduced in recent years is provided. The methodologies are classified into classic computer vision methodologies and modern deep learning approaches. In addition to the chronological overview, a hierarchical taxonomy of the technological development in IBL techniques is presented. Furthermore, the mathematical formulation of particle filters is introduced and discussed.

Chapter 3 presents a cross-view IBL method, where the ground-level images from the KITTI dataset are used to train a neural network capable of comparing ground-level images with satellite images. Using a transformer architecture, a triplet soft-margin loss function, and paired ground-level and satellite imagery, the proposed Cross-View Spatial Transformer (CVST) model is trained and evaluated against several existing architectures, demonstrating superior performance on the image retrieval task for image-based localisation. Note: The research presented in Chapter 3 corresponds to the published work named “Cross View Shared Transformer (CVST) a shared weight siamese network for limited fov image based localisation.” presented in the International Conference on Intelligent Transportation Systems (IEEE ITSC 2024) [10]. In this chapter, Research question 1 and 3 are addressed through the development of a CVST.

Chapter 4 focuses on aerial image-based localisation in both urban and rural environments. This chapter describes in detail the methodologies employed to capture the datasets for both urban and rural scenarios. To capture the rural dataset, a UAV is deployed in Ireland to map the region where the localisation algorithm is tested. Additionally, the UAV is

navigated in random walks to collect the testing data. For the urban dataset, the KITTI dataset locations are leveraged, and the dataset is complemented with images captured from two different satellites at different times, producing a cross-time satellite dataset. Finally, the CVST network is tested for both scenarios with the recall@k metric, demonstrating superior performance in the aerial localisation task as well. Note: The research presented in Chapter 4 corresponds to the published work entitled “Aerial Image Based Localisation via Learned Embeddings” presented in Irish Machine Vision and Image Processing Conference Proceedings 2024 [9]. In this chapter, the research questions 2 and 3 are addressed through the implementation of CVST in both rural and urban environments for UAVs in near-real time operation. Leveraging pre-encoded embeddings ordered in a standard database.

Chapter 5 integrates a particle filter into the IBL process to mitigate the aliasing noise produced by the neural network. To improve the robustness of the IBL workflow, a novel likelihood metric that incorporates the knowledge of the road network is formulated. Combining image and road-network based likelihoods allows the system to converge earlier to the position estimate. Furthermore, this thesis presents the methodology used to enable near-real time operation by integrating an efficient data structure with the particle filter. Research questions 3, 4 and 5 are addressed in this chapter through the development of a near-real time IBL system.

Chapter 6 concludes the thesis by discussing the overall contributions, ethical implications, and potential future research directions arising from the developed techniques.

Chapter 2

Image-based Localization

State-of-the-Art

2.1 Introduction

Robotic systems have reached a maturity level where they can perform complex autonomous tasks. Systems such as Atlas, Spot or Ingenuity shown in Fig. 2.1, among others, are capable of performing complex motions and navigating rough terrain. Despite their high levels of dexterity and manoeuvrability, any mobile platform relies on a localisation system to navigate indoor or outdoor environments. Classic localisation methods rely on GNSS technology to obtain geolocation. However, GNSS is susceptible to multipath effects, signal jamming and weather conditions, amongst other factors. The causes of these issues are the dependence on multiple satellites to calculate the position based on triangulation. Meaning that any disturbance in the signal will cause inaccuracy in the platform's position, interfering with the control components of the machines.



Figure 2.1: Images of Atlas (left) [107], Spot (top right) [17] and Ingenuity (bottom right) [98]

An alternative method for navigation and localisation is Simultaneous Localisation and Mapping (SLAM). SLAM systems leverage multi sensor platforms to navigate over unknown environments and generate a local map of the area. Mobile platforms that employ SLAM operate with proprioceptive sensors such as inertial measurement unit (IMU), accelerometers and encoders, as well as with exteroceptive sensors like LiDAR, sonar and optical sensors. During motion, the platform leverages the advantages of each sensor to improve the scene understanding and keep track of its performed movements to navigate in the unknown environment.

SLAM algorithms have evolved to the point where Visual Inertial Odometry is implemented to improve the accuracy, as developed in ORB-SLAM 3 [23]. Despite the ability of SLAM systems to explore complex environments, they are not capable of geolocating the platform in a global coordinate system without GNSS. Visual SLAM is being primarily demonstrated in indoor environments or small places with controlled light conditions. Unfortunately,

visual SLAM is susceptible to lighting conditions as it mainly depends on the optical sensor. Because of its dependency on classic feature extraction methods, sensor saturation can cause the system to lose position or fail to converge during the map estimation. In addition, classical feature extraction methodologies used in SLAM can be computationally expensive.

Let's imagine a tourist exploring the city for their first time. After leaving the hotel, the individual spots the name of a well known shop in Dublin. Assuming that he might get lost during his city exploration, he takes note of the shop name and orientation to ensure he remembers his way back to the hotel. In addition, he marks this shop on a map given by the hotel staff.

After a few kilometres of walking, the tourist gets distracted and realises he is lost. In addition, he does not remember how many right or left turns were made during his walk. After looking around he finds himself next to a park with a big statue and in a big road junction. Making use of the map, the lost tourist tries to find a green area and the same road junction. Successfully, he is capable of matching the shape of both the junction and the park to the junctions and green areas present in the map, revealing his location. Now relieved that he knows the way back, he proceeds to return to the hotel.

The individual, unaware of his current location, performed a complex localisation workflow that enabled him to find his way back. First, he was required to extract features found in his current environment to match them against previously visited places. When no matches were found, as an alternative, he tried to match those extracted features against a map showing the overhead view of the city. From this example, it can be noted that all of the complex tasks required to be performed by a machine to localise itself based on an image, which is genuinely the same as locating the lost tourist.

Image-based localisation (IBL), also known as visual localisation (VL) or image place recognition (PR), is a long-studied problem in computer vision and robotics, with numerous techniques reported in the literature. These techniques will be analysed in depth throughout the chapter. **Visual localisation is defined as the task of estimating the geolocation of a query image based on its visual representation of the scene.** The VL is commonly formulated as an image retrieval problem, where the query image is somehow

compared against a database of geotagged images.

To geolocate a platform, the computer vision community has formulated a localisation solution based solely on captured images. Visual localisation attempts to exploit the optical sensor of mobile platforms as navigation systems. Like the lost tourist trying to locate itself in a city, the visual localisation challenge tries to mimic the process that a human follows when finding their position in an unknown environment. A lost human would try to match the features in their current environment, such as facades, shop signs, building colours, road network, among others, to match them against locations that have been previously visited. Alternatively, if a map is provided, the lost tourist would match the features of its surroundings against the provided map. Finding those matches would allow the lost individual to guess their location. Image-based localisation aims to reproduce this behaviour for robotic platforms.

Studying IBL is essential as it can provide an alternative navigation system for mobile platforms operating under GNSS denied environments. Companies such as Google, Microsoft, and Baidu, among others, have collected extensive publicly available collections of georeferenced street view and satellite images. Mobile platforms can leverage these geotagged image collections as a digital map. There are also vast image collections that lack geotags and might be of interest to classify by location. As an example, companies such as Facebook, Instagram or Snapchat, among others, have collected a large amount of images that have not necessarily been georeferenced. A secondary use case of VL is to geotag image collections with an unknown capture location.

Despite the progress made to date in visual localisation, several open research challenges remain, including robustness to viewpoint variations, dynamic scenery, seasonal / illumination changes, perceptual aliasing and real-time geo-localisation. Some methodologies apply classic computer vision techniques to extract features used to geolocate images against a geotagged database. This problem is commonly solved as an image matching task, where the query image is compared against a geotagged database to find its location. The computational cost of classic methodologies has driven research to look for alternative solutions. The introduction of Convolutional Neural Networks (CNNs), which revolutionised

computer vision in 2012, has moved research to explore Deep Learning techniques for image localisation.

The localisation problem in Robotics can be classified based on mobile platform characteristics, as each platform's configuration affects the suitability of localisation methodologies. The aim of this chapter is to explain the background of IBL and attempt to classify the different methodologies based on the nature of the mobile platform. The camera configuration on the mobile platform impacts the operational viewpoint, causing a divergence on the methodologies utilised for localisation of vehicles. As an example, localising an unmanned underwater vehicle (UUV) requires different equipment and algorithms to the ones utilised to localise an unmanned ground vehicle (UGV) or an unmanned aerial vehicle (UAV). For this reason, the localisation techniques are divided into two categories, Cross view localisation and aerial localisation. Cross view localisation consists of finding a geographic position of an image or sequence of images taken from a vehicle navigating on the surface of the earth. Aerial localisation is referred in this text as the problem of finding the geolocation of an aerial platform with the camera pointing in the direction of gravity, also called birds-eye view. However, there are different viewpoints for aerial localisation depending on the azimuth and pitch of the onboard camera; these types of viewpoints are explained in further sections.

2.2 Related Work and Foundational Concepts for IBL

Early research on IBL, motivated by the need for platform operation in GNSS-denied environments, focused on how geotagged image databases could be used to match a query image through handcrafted features. Treating the geolocalisation challenge as an image-matching and retrieval task, this work employed classical computer vision algorithms to estimate image location. [161, 114, 119, 156, 131, 113, 118].

This section uses the ‘Lost Tourist’ analogy explained in the introduction to focus on the workflow performed. It’s important to dissect the detailed steps the tourist follows to estimate a geolocation. The tourist’s first step involved gathering a collection of reference landmarks. As a second step, the tourist uses his current position in the environment to identify visible reference points. Third, he proceeds to match the reference points he can locate to the landmarks on a map. Likewise, the computer vision community has developed a suite of tools and algorithms to enable the computation of visual localisation. This section first explains the different mobile platforms used for image-based localisation, classifying them into various configurations and describing the sensors integrated to measure the environments. This is followed by an introduction to the datasets commonly used for IBL. Then, an overview of the feature extraction mechanisms used in the early days of IBL is presented. The following sections introduce the main tools needed to understand the further development of IBL.

2.2.1 Platforms for Data Capture

Before introducing the most commonly used datasets, it is first necessary to classify the different viewpoints from which imagery may be captured. Data for visual localisation can be collected in a variety of forms, each acquisition method exhibiting distinct characteristics. The most prevalent visual scene representations include horizontal pinhole camera images, horizontal panoramic images, ground-upwards oblique imagery, aerial downwards oblique imagery, vertical aerial imagery, and satellite imagery. Fig. 2.2 shows how different mobile platforms capture data and the available viewpoint geometries for data capture.

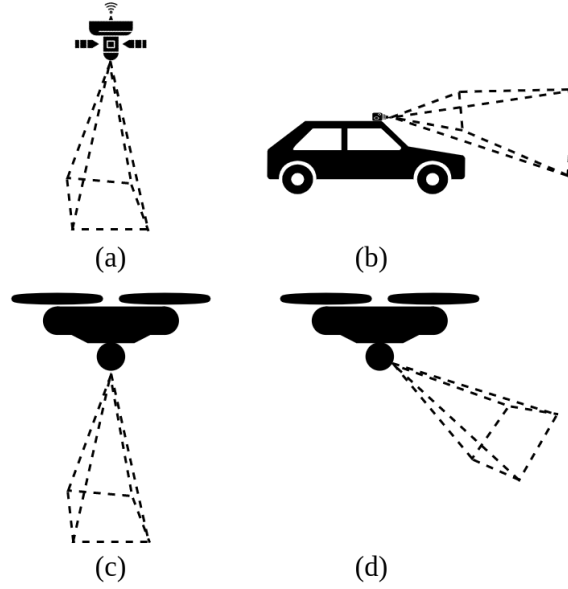


Figure 2.2: Different platforms capture data from different viewpoints. a) Satellite imagery viewpoint. b) Ground or horizontal viewpoint. c) UAV oblique downwards viewpoint. d) Aerial or overhead viewpoint.

Satellites can capture optical data over large areas at different resolutions. The resolution of the image will depend on the sensor quality installed in the satellite. Satellites are equipped with different types of optical detectors, allowing them to capture multispectral images, which capture measurements at different wavelengths that can not be perceived by the human eye. Other types of optical detectors in Satellites are digital camera frames that collect RGB and near infrared (NIR) data. They are also equipped with discrete scanner detectors, which employ a mirror to scan the area and capture RGB and NIR data. Among other types of detectors. Not all satellites contain multispectral or RGB data; this depends on the type of satellite used to collect measurements. These orbiting devices can collect data over large areas and at impressive high resolutions now days. However, the data will depend on the satellite's orbit trajectory. Meaning limited samples per day are allowed, because the satellite orbit will restrict the time and area at which the data will be captured. [62]

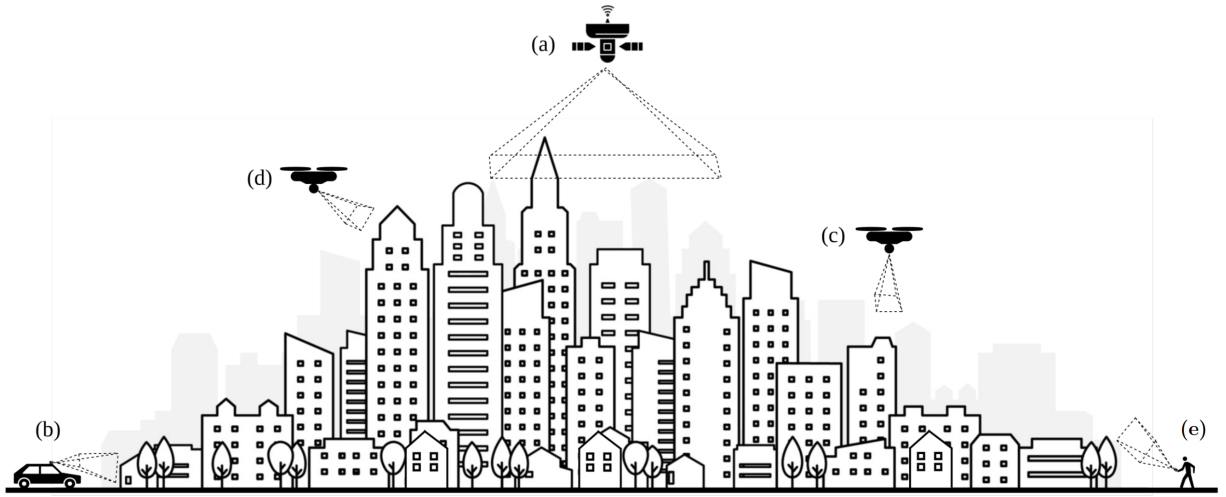


Figure 2.3: Different platforms capture data from different viewpoints. a) Satellite imagery. b) Ground Vehicle horizontal viewpoint. c) Aerial downwards viewpoint. d) Aerial oblique downwards viewpoint. e) Ground oblique upwards viewpoint

Aerial platforms capture data with a smaller spatial footprint than satellite imagery, as UAVs operate at much lower altitudes above the Earth’s surface as shown in Fig. 2.3. Multiple types of sensors can be mounted on the chassis of aerial platforms, making them more versatile for collecting multimodal data. LiDAR, thermal, and RGB data can be captured simultaneously on a quadcopter that can handle the equipment’s weight. With visual localisation becoming more popular over the years, this approach is becoming increasingly common. In addition, the data resolution is very high and is not dependent on a satellite’s operating orbit.

Ground platforms, like aerial platforms, can be equipped with multiple types of sensors, making them highly versatile for data capture. Furthermore, ground vehicles can operate over large areas for prolonged periods of time. A good example of this are the Google vehicles that capture RGB imagery of the cities. Thanks to their high load capacity, ground vehicles can carry multiple large and high resolution cameras, LiDAR sensors among other equipment. The downside of ground vehicles is shown when capturing GNSS data over urban areas. As discussed previously, the cluttered environments cause multipath effects, signal jamming, among others, that cause the geolocation measurements to be noisy.

2.2.2 IBL Open Datasets

The performance of Visual place recognition is partly attributed to the type of data used when formulating the retrieval problem. Several open, publicly accessible dataset repositories can be used in IBL research. The most popular type of data utilised for image-based localisation is horizontal images limited to the optical sensor field of view (FoV) or panoramic images which capture a 360° FoV. Although both are Red Green and Blue (RGB) images, the difference in FoV has a drastic impact on the accuracy performance. The increase in performance is given by the fact that panoramic images capture more features of the sampled location. Whilst an image limited by its field of view only captures a partial representation of the sample location, and is also constrained to its compass orientation.

The publication of large scale datasets has helped improve the performance and standardise the performance metrics of visual localisation algorithms. The first approach of the computer vision community is to capture a standard and generalised Dataset where VL algorithms can be tested and compared. However, in the attempt of standardising the data, each dataset released contains unique characteristics. During the last decade, multiple data collections were released across different countries with different capture methodologies. Among the most popular collections, Oxford buildings [103], Paris Buildings [1], St. Lucia [148], KITTI Dataset [43], Pittsburgh 30K [138], Tokyo 24/7 [137], CVUSA [152], Oxford Robo Car Data Set [88] and KITTI 360 Data Set [3] among others, are the most common datasets for visual localisation. Despite the multiple data options, there is not a standard method to capture a dataset and locations around the world are visually different, causing datasets to have their own visual characteristics, unique collection methodologies and use limitations. A timeline of the public available datasets is shown in Fig. 2.4.

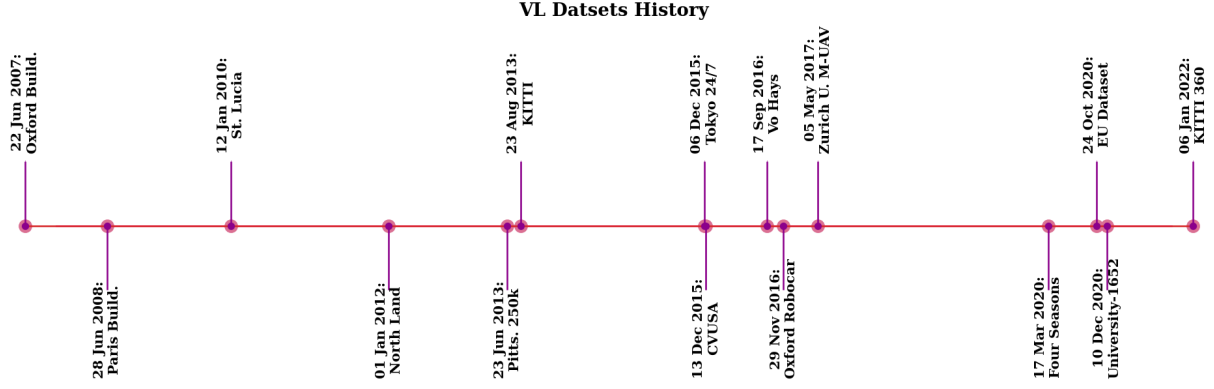


Figure 2.4: Timeline of the most relevant datasets for IBL publicly available to date.

The Oxford Buildings Dataset [103] is a dataset build for testing image retrieval tasks. The collection is composed of three different sets of Oxford city building images. A collection with approximately 5 thousand high resolution images crawled from Flickr. Two more sets of images were utilized during the testing of the retrieval algorithm, the second one with approximately 100 thousand images and the third one with one million images. Despite the large number of collected images, the dataset does not come with geolocation identifiers for the images.

Similar to the Oxford building dataset, the Paris Dataset [1] is composed of 6300 high resolution images. The images were collected from Flickr by querying text tags associated with famous landmarks. This dataset was collected as an augmentation to test for soft assignment methodology on image retrieval tasks. To test the performance of the retrieval algorithm in two different scenarios, both the Oxford building dataset and the Paris dataset were used to compute the retrieval metrics. As the Oxford building dataset, the new images presented in the Paris collection are not geotagged. This restricts the use of both the Oxford and Paris building datasets for image retrieval tasks, not allowing for measuring the accuracy with geolocation metrics.

Focused on Simultaneous localisation and mapping (SLAM) tasks, Warren *et al.* [148] released the St. Lucia dataset. The novel dataset contains geotagged video sequences collected with stereo cameras. Each camera was placed on the roof of a ground vehicle with 800mm distance between sensors. The system is capable of collecting and geotagging the images with the measurements received from the GNSS. The image sequences are

utilised to demonstrate a SLAM algorithm based on SIFT descriptors. In their work, they demonstrate the improvement in trajectory estimation when using bundle adjustment algorithms. Without knowing at the time, the authors have also developed one of the most common datasets used for image-based localisation. Although, this is one of the first datasets to contain a collection of geotagged images, the resolution of the images is lower compared to the ones used in the Oxford building dataset and the trajectory where data was collected was constrained to a small area.

North Land Data Set [128] is a collection of images taken from a train trajectory of $\approx 700\text{KM}$. The data is captured between an area near Trondheim and the city of Bodø. The trajectory was repeated four times during winter, spring, summer, and fall season. Making one of the largest multi seasonal datasets. GPS information is provided with the dataset, however the GPS measurements are very noisy and requires manual data cleaning to match them with the corresponding frames.

The KITTI Dataset [43] is one of the most popular datasets used in robotics. This dataset was captured using state-of-the-art LiDAR and optical sensors at approximately 100Hz. An engineered rig mounted on top of a Volkswagen vehicle held a Velodyne LiDAR scanner, two PointGray Flea2 gray scale cameras, two PointGray Flea2 colour cameras and one OXTS RT3003 inertial and GPS navigation systems. More than 6 hours of driving footage was collected over different areas in Karlsruhe, Germany. The dataset includes dense urban areas, highways, rural areas and residential areas around the city, making it one of the most complete datasets to date.

Recognising that urban environments contain numerous repetitive structures—such as windows, doors, and brick patterns, Torii *et al.* [138] exploit these architectural regularities to enhance performance in the image-retrieval task. To test their retrieval algorithm, a new dataset called “Pittsburgh 30K” is presented. The dataset consists of 254,064 perspective images computed from 10,586 Google Street View panorama images in Pittsburgh. The Google Street View panorama images contain a resolution of $6,656 \times 3,328$ pixels. A panorama image is comprised of 24 perspective images of 640×480 pixels that correspond to a 60 degrees horizontal FoV sensor. Each perspective image is calculated with two

yaw directions [4 deg, 26.5 deg] and 12 deg pitch directions [0 deg, 30 deg, . . . , 360 deg]. This dataset is a good example for localisation in cluttered urban environments, as it contains images with multiple building facades. Although the dataset allows testing for scenes with repeated structures, the collection of images is biased to a single urban location.

With visual localisation becoming more popular over the years, Torii *et al.* [137] introduce a new ground-level image dataset, Tokyo 247, designed to account for the substantial changes in urban appearance between day and night. The dataset is intended to evaluate the robustness of image-retrieval methods under varying lighting conditions. Images were collected at 125 distinct locations within a $1.6 \text{ km} \times 1.6 \text{ km}$ area of Tokyo. At each location, three images were captured in different viewing directions and at three separate times of day. All imagery was acquired using Apple iPhone 5s and Sony Xperia devices. Although the dataset comprises 1,125 images captured under diverse illumination conditions, it remains limited to a relatively small urban region in a single city.

In an attempt to introduce a cross view dataset, Workman *et al.* [152] proposed an extensive collection of street view images paired with corresponding satellite images called “Cross View USA” (CVUSA). Ground level images are collected from two main sources, Google Street View and Flickr. Ground level images are collected from two main sources, google street views and Flickr. The corresponding satellite images are collected from Bing Maps with a cover area of $800 \times 800 \text{ px}$ at different zoom levels. The centre of the satellite image corresponds to the matching street view image. The complete dataset consists of 1,036,804 street view images and 879,318 aerial images, covering almost the entire country. Although the dataset is large and covers a large extent of areas mixing urban, rural, wild and coastal scenery, the arbitrarily centring of the street view image in the corresponding satellite image only applies to panoramic images with a 360° FoV. While, images collected with a constrained field of view should be collected in front of the captured location and not on top of it. Capturing corresponding aerial images centred by the constrained FoV street view images causes that features in half of the satellite image to not correspond in between view points.

In their investigation of methods for determining the location and orientation of street-

view images, Vo and Hays introduce a new dataset for cross-view localisation, Vo and Hays introduce a new dataset for cross-view localisation [142]. This dataset comprises Google Street View imagery collected from eleven cities across the United States. For each panorama image, a random crop was performed, and the corresponding satellite image was extracted, generating more than one million matched pairs. However, the dataset maintains a similar issue to the one discussed for the CVUSA dataset. The corresponding satellite image is taken on top of the street view location, capturing features that are not shown in the street view image. In addition, to match the orientation between the street view image and the overhead image, a rotation transformation is computed to the satellite image causing corners with black areas. The black areas caused by the rotation introduce artefacts in the dataset that should not be present.

Oxford Robot Car Data Set developed by Maddern *et al.* [88], focuses on improving robustness for autonomous driving. With a similar set-up to the one presented by the KITTI dataset, the robot car dataset distinguishes itself by capturing data under different driving conditions. More than 20TB of images have been captured over 1000km during a one year period. The new dataset allows for research to test algorithms under multiple illumination conditions, structural changes, in order to develop long term driving autonomy. The vehicle is equipped with one colour stereo camera (Point Grey Bumblebee XB3), three fish eye lenses cameras (Point Grey Grasshopper2) to the rear and sides of the vehicle, three lidar scanners and one GNSS system. The data collected by the Oxford Robot Car makes one of the most complete datasets presented. The only downside of this dataset is that data is only collected over one urban environment. In addition, inaccurate GNSS measurements [150] were present during the data collection, which were later improved using real time kinetic (RTK) systems [89].

As demonstrated by previous research on data collection, numerous datasets have been captured for autonomous driving; however, comparatively few aerial datasets exist due to the logistical and legal constraints associated with operating aerial vehicles. To address this gap, Majdik *et al.* released the Zurich Urban Micro Aerial Vehicle (MAV) Dataset. [90]. A tethered drone is used to capture aerial oblique images for a 2 km urban circuit in the city of Zurich. A GoPro hero 4 camera is mounted to the Fotokite to capture the

imagery. The data is released to the public for research in SLAM and aerial localisation systems. Despite being one of the first urban aerial datasets, the surveyed area does not consider illumination nor seasonal changes.

To address the limitations of the Oxford RobotCar dataset, Wenzel *et al.* introduce “Four Seasons” [150]. This dataset is designed for long-term SLAM research and comprises multiple traversals of the same environment over extended periods, enabling evaluation under significant seasonal and environmental variation. To avoid GNSS inaccuracies, the Four Seasons system integrates RTK technology during data collection. During data collection, the system operates dual inertial sensors and stereo cameras separated by 30 cm. Multiple trajectories are collected around the Munich area, where rural, urban, residential and highway scenarios are captured over multiple weather and seasonal conditions. The Four Seasons dataset solves the accuracy problem by integrating the RTK system; however, it only uses greyscale sensors, creating one disadvantage over the Robot Car and Kitti Datasets that operate full RGB cameras.

As a successor of the successful KITTI dataset, Liao *et al.* [3] released the KITTI-360 Data Set. As a second attempt on developing a dataset focused for autonomous driving and SLAM algorithms, the platform integrates two perspective RGB cameras with a base distance of 60 cm, a Velodyne LiDAR scanner, a pair of lateral fish eye cameras and a second SICK LiDAR scanner. In addition to the data, segmentation annotations are provided with the dataset to support image and LiDAR segmentation research. However, unlike the Four Seasons dataset, no RTK system is employed during the data collection. Instead, to measure global position, a GPS is used for localisation measurements.

The last dataset analysed is the University-1652 introduced by Zheng *et al.* [163]. UAVs are not always operated with a downwards-looking camera; instead, a lot of aerial footage is captured at oblique viewpoints looking downwards, and ground-level images can also be collected with oblique viewpoints looking upwards. The University 1652 dataset is the first datasets that collect multiple viewpoints of the same location. The novel collection aims to bridge the gap of visual appearance by data that can be used to learn viewpoint invariant representations of locations or buildings. 1,652 university buildings are surveyed

around the world to capture multiple platform viewpoints, such as quadcopters, phones, and satellites, which are grouped into one set. For the aerial view satellite images are utilised, for the oblique downward views a simulation is generated using the Google Earth 3D models of the buildings. Finally, for the ground views, Google Street View images are collected for every building. Approximately 58 images are collected per building in different viewpoints. The novel dataset introduces a multi-view source for benchmarking cross-view matching algorithms. Despite the novelty of the data collection protocol, the dataset only includes buildings, not allowing one to test the overall robustness of the algorithms, but rather the ability to encode building features.

To classify the publicly available datasets, a table identifying different characteristics used for data collection is presented in Tab. 2.1. The characteristics chosen are: geolocation, stereo camera collection, LiDAR collection, viewpoint orientation, data resolution and purpose

Visual Dataset						
DS Name	Geotag	Stereo	LiDAR	Viewpoint	Resolution	Purpose
Oxford Build.	No	No	No	GV	1024×768	Image Retrieval
Paris Build.	No	No	No	GV	1024×768	Image Retrieval
St Lucia	Yes	Yes	No	GV	1280×960	SLAM, VL
North Land	No	No	No	GV	640×360	Seasonal VL
KITTI	Yes	Yes	Yes	GV	1241×376	SLAM, VL
Pittsburgh	Yes	No	No	GV	640×480	Image Retrieval
Tokyo 24/7	Yes	No	No	GV	3264×2448	Image Retrieval
CVUSA	Yes	No	No	CV	2448×3264	CVL
Vo and Hays	No	No	No	CV	354×354	CVL
Oxford RBC	Yes	Yes	Yes	GV	1280×960	SLAM, VL
Zurich MAV	Yes	No	No	OAV	1920×1080	SLAM, VL
4 Seasons	Yes	Yes	Yes	GV	800×400	SLAM, VL
KITTI 360	Yes	Yes	Yes	GV	1408×376	SLAM, VL
Uni.-1652	Yes	No	NO	MV		VL

Table 2.1: A table classifying the datasets up to date based on collection perspective and aimed purpose. Cross View (CV), Oblique aerial view (OAV), Multi view point (MV)

2.2.3 Feature Extraction Methodologies

The ability to extract repeatable features that allow image comparison is fundamental to tasks such as image matching, image stitching, visual localisation and SLAM algorithms. Feature extraction is one of the main components in computer vision as it opens the door for large image collection analysis. Looking at it from a different perspective, image features are the input data for multi image analysis.

The key component of visual localisation lies in the ability to extract image features that are comparable between different viewpoints. Multiple methodologies for image comparison have been developed by the computer vision community. Images can be compared on a pixel to pixel base, in the Fourier space or through more complex feature extraction techniques. These techniques can be divided into handcrafted features or learned features (LF). Among the most popular methodologies, SIFT, SURF, and ORB have excelled in the feature extraction task. However, these handcrafted techniques have been outperformed by new deep learning methodologies that learned features from large image collections.

2.2.4 Handcrafted Features

There are many types of image feature extraction methodologies that have been developed for the purpose of image analysis, 3D scene understanding and computer vision. Some of these features can be extracted by analysing images to look for corners. The most famous algorithm for corner detectors is the Harris detector, introduced by Harris *et al.* [48]. The first step of this detector consists of computing the horizontal and vertical derivatives of the image. In other words, it computes the Gaussian derivative of each pixel. Second, it calculates the second moment matrix M in a Gaussian window around each pixel. Third, uses the second moment matrix M to calculate the corner response function, defined as R . Then a threshold process is applied to the R matrix and finally it finds the local maxima of the response function, applying a non maxima suppression. This methodology returns rotational invariant features and is partially invariant to intensity.

One type of scale invariant feature detection algorithm is the Harris-Laplacian detector

presented by Mikolajczyk *et al.* [93]. Essentially the procedure finds local maxima of Harris corner detections in space and the Laplacian in Space.

On the other hand, Scale-Invariant Feature Transform (SIFT)[86] [85] finds the local maxima of Difference of Gaussian (DoG) in space and scale. Where the DoG is a pyramid of DoG at different octaves. Additionally, SIFT is capable of computing the best orientation for each key point region and a key point descriptor calculated with local image gradients provides additional information of the key point region.

Scale and rotational invariant features allow matching scenes in motion, making them very useful for visual SLAM, image matching and image retrieval tasks. In Figure 2.5, the capabilities of feature detection to match two images of the same scene taken at different illuminations are shown.

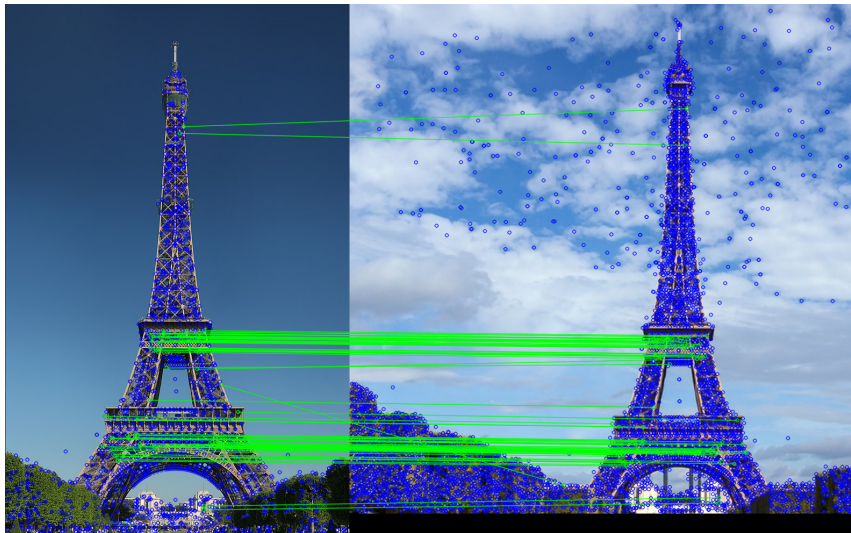


Figure 2.5: Example of image matching features using the SIFT algorithm with Flann matching.

An alternative corner detection algorithm is presented by Rosten *et al.* [109] called features from accelerated segment test (FAST). This algorithm uses a circular area around a potential corner. A corner is detected if a continuous section of the pixels are brighter than the centre plus a determined threshold or darker than the centre minus the same threshold. Although Fast is a computationally efficient algorithm, it fails under artificially generated images.

Speeded Up Robust Features (SURF) presented by Bay *et al.* [13] builds a feature extraction

algorithm that is robust to scale and rotational variations. Additionally, it incorporates the basis for Hessian detectors, making it more repeatable and faster than SIFT. The SURF algorithm outperforms SIFT in most of the cases, thanks to the fact that SURF integrates the gradient information in a sub-patch used for the computations of the descriptors.

Years after SURF was proposed, Binary Robust Independent Elementary Features (BRIEF) took the lead as an image feature descriptor. Proposed by Calonder *et al.* [22], the new algorithm identifies images features on a pixel by pixel basis. It employs Gaussian kernels to pre-smooth the analysed image patch, the image patch is then converted to a binary vector based on the intensity of the centre pixel. BRIEF computes the features faster than previous state-of-the-art detectors because it relies on a few intensity change tests to generate the binary vector for the patch. In addition, BRIEF generates more repeatable features that yield higher recognition rates as long as the rotation variations are in between ∓ 15 degrees.

Using FAST and BRIEF detectors as a base, Rublee *et al.* [110] purpose an open source feature extraction algorithm coined as Oriented (FAST) and Rotated BRIEF (ORB). The ORB algorithm introduces a new methodology that is invariant to rotation and is one order of magnitude faster than SURF and SIFT. An orientation metric is added to the FAST descriptor, and a learning step is introduced, which finds less correlated binary tests, improving the BRIEF descriptor. Despite its speed, on its downside, ORB is not scale invariant.

2.2.5 Feature Clustering

Handcrafted features are capable of representing images based on interest points that describe the scenery. They also allow comparing image similarity by comparing images in feature space. However, it is of interest to reduce the dimensionality of handcrafted features to increase speed in the algorithms and reduce memory consumption.

Bag of visual words (BoVW) presented by Sivic *et al.* [127] introduces a methodology that incorporates a dictionary or database of visual subsections of images called “visual words” (VW). The visual words can be extracted through numerous methodologies, among those

methodologies are defined grids or areas around handcrafted features such as SIFT, SURF or ORB. In the case of grids, each visual feature will be extracted either at a corner of the grid or the sub-section of the grid will be recorded as a visual word. For handcrafted features, both the key point and the descriptor parameters are taken into consideration to extract an area of interest around the detected key point. Once the areas of interest are extracted from each image, the number of occurrences of the visual words in an image is counted and each image is now represented by the histogram of VW in the particular image.

The extracted descriptors are then clustered by the k-nearest neighbours algorithm, which defines the grouping or cluster centres of different descriptors, defining the dictionary BoVW. A large set of images is used to generate the BoVW dictionary, as this increases the sampling subset. To find the similarity between images, the histogram representation of the images is compared by the cosine similarity, meaning images do not have to be compared neither in pixel nor feature base. Thus, reducing the dimensionality of the image representation and improving efficiency of the retrieval task. However, it is important to consider that although increasing the size of the BoVW increases accuracy, it impacts directly the retrieval inference time. In practice, not all the VW are good for matching purposes, as some VW might appear too often in most of the images. For example, visual words extracted from the sky, clouds, trees, plain concrete, among others, might occur repeatedly among a large portion of the sampled set of images used to generate the BoVW dictionary. The large occurrences make these VW not suitable for matching, as they do not allow discriminating in between images. To account for this, the term frequency inverse document frequency (TF-IDF) shown in Eq. 2.1 is employed. The TF-IDF re-weights each bin of every image histogram based on the amount of occurrences in the population, giving more importance to the VW allowing particular images to be distinguished.

$$t_{id} = \frac{n_{id}}{n_d} \log\left(\frac{N}{n_i}\right) \quad (2.1)$$

Where, t_{id} denotes the histogram bin of VW i for image d . n_{id} represents the occurrences of VW i in image d . n_d is the number of VW word occurrences in image d . n_i is the number

of images that contain the VW i and N is the total number of images. In order to find the n most similar images in the database given a query image, the most likely matches will be the n images with the smallest cosine or Euclidean distance between histograms. Although, this methodology performs well for matching images, finding a similarity metric between images and detecting objects, the algorithm is highly dependable on the visual dictionary. If a new type of object is required to be added to the BoVW the dictionary needs to be recomputed, and the inference time is highly dependable on the size of BoVW.

Jegou *et al.* [64] explore the image retrieval problem. Proposing a new feature representation called vector of locally aggregated descriptors (VLAD), Jegou *et al.* improve the capabilities of searching the most similar image in a database of 10 million images in 50ms. First, the dictionary of VW is learned in the same manner as BoVW. Where the dictionary is represented as shown in Eq. 2.2, k is the number of visual words also known as number of centroids, and c_i is the i^{th} visual word or cluster centroid. Each image descriptor x is associate to the nearest visual word by the k-nn algorithm $c_i = NN(x)$.

$$\mathcal{C} = \{c_1, \dots, c_k\} \quad (2.2)$$

As shown in Eq. 2.3 VLAD features uses the accumulation of the differences between the features $\mathbf{x}_i \in \mathbb{R}^{1 \times D}$ of dimension D assigned to cluster centroid $\mathbf{c}_k$ also called VW. Where, $a_k(\mathbf{x}_i) \in \{0, 1\}$ is a value zero or one representing the membership of the feature vector $\mathbf{x}_i$ to the cluster centre $\mathbf{c}_k$. k represents the number of cluster centres or VW. N is the total number of images and j is the dimension of the feature vector $\mathbf{x}_i$.

$$V_{i,j} = \sum_{i=1}^N a_k(\mathbf{x}_i) (\mathbf{x}_i(j) - \mathbf{c}_k(j)) \quad (2.3)$$

Using the accumulation of residuals between the image feature representations and the extracted VW the dimensionality of the representation is reduced and by consequence reducing inference time during retrieval. After experimentation, VLAD outperforms BoVW and Fisher vectors with a mAP =0.494 on the Holidays Data Set.

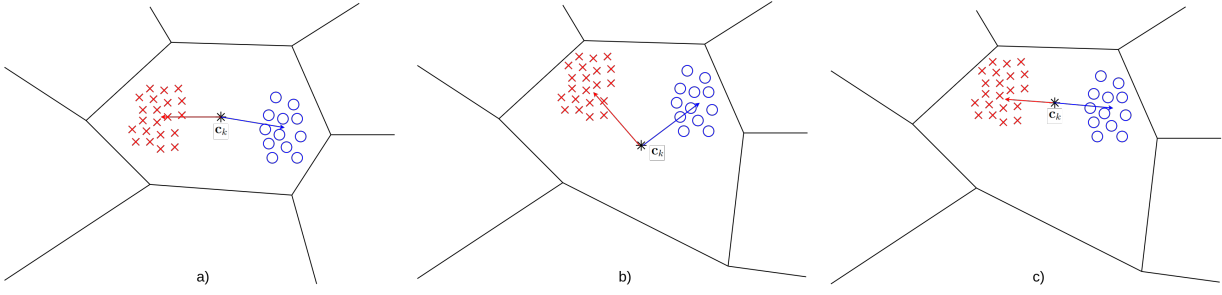


Figure 2.6: Voronoi cells demonstrating the cluster centre closed for a single cluster k . a) An ideal cluster that represents the two feature sets in blue and red. b) A cluster that does not represent the data in an optimal space. c) The adapted cluster centre fits the data in a geometry in which the difference between the residuals is maximum.

The VLAD cluster used to construct the VLAD descriptors can be represented by Voronoi cells. These illustrations allow explaining the feature clustering methodology of VLAD as shown in Fig. 2.6(a).

In later research, Arandjelovic *et al.* [8] propose cluster adaptation to improve retrieval performance using VLAD representation. As noted in their research, cluster centres might not be consistent with the entire dataset, as new data might be added to the BoVW, or only a subsample of the entire population was used to construct the BoVW. The proposed methodology adapts the cluster centres based on the average location of the cluster features, while VLAD fixes the cluster centre accordingly to the sampled data features. The best solution would be to recluster according to the complete dataset; however, this can be costly and not all the data will be known, causing a constant reclustering. The similarity between two image descriptors is shown in Eq. 2.4 demonstrating the importance of the cluster centre when calculating the similarity between VLAD descriptors. Note that the VLAD descriptors have unit norm by normalising the residuals with a constant C^i

$$\frac{1}{C^{(1)}} \sum_{i=1}^N (\mathbf{x}_{k,i}^{(1)} - \mathbf{c}_k)^T \frac{1}{C^{(2)}} \sum_{i=1}^N (\mathbf{x}_{k,i}^{(2)} - \mathbf{c}_k) \quad (2.4)$$

The alternative solution is to ensure the cluster centres maximise the distance between image descriptors. The proposed algorithm is divided into two steps. One, compute the adapted cluster centre as the mean of all local descriptors in the dataset assigned to the same cluster k . Two, recompute the VLAD descriptors by aggregating the new residual difference. Fig. 2.6 illustrates features from two different images matched to an arbitrary

cluster k . The figure demonstrates the advantages of recomputing the cluster centre. Fig. 2.6(a) shows a cluster centroid that classifies the data as it is geometrically placed between the mean of both feature clusters. However, it can be noted from the Fig. 2.6(b) that the cluster does not fit the data optimally, affecting the angle difference in between the residual vectors (blue and red arrows) of both features. Fig. 2.6(c) demonstrates how adapting the cluster centre to the data can increase the difference between the two features, resulting in better classification.

2.3 Image-Based Localisation (IBL)

With the ongoing advancement of image feature extraction techniques [48] [85] [13] [22], combined with improvements in feature-matching algorithms, these emerging technologies have increasingly been adopted for visual localisation tasks. Image-Based Localisation (IBL), also referred to in the thesis as Visual Localisation (VL) or place recognition (PR), is defined as **the task of predicting the location of a query image by matching it with a geotagged collection of images**. Normally, the VL challenge is formulated as an image retrieval task.

The generalised framework for VL is presented in Figure 2.7. First, a collection of geotagged images is pre-encoded to its feature representation by a feature extraction methodology of choice. These feature representations are stored in a database that is used for the retrieval step. The query image or image to locate is also encoded into its feature representation by the same feature extractor. Applying a matching algorithm, the query image feature is compared against the database of encoded features with a similarity measurement. Once the comparison is finished, the stored features are sorted based on the similarity metric. The top best matches are returned as the estimated location. As they have been tagged with a geolocation, the position of the query image can be estimated.

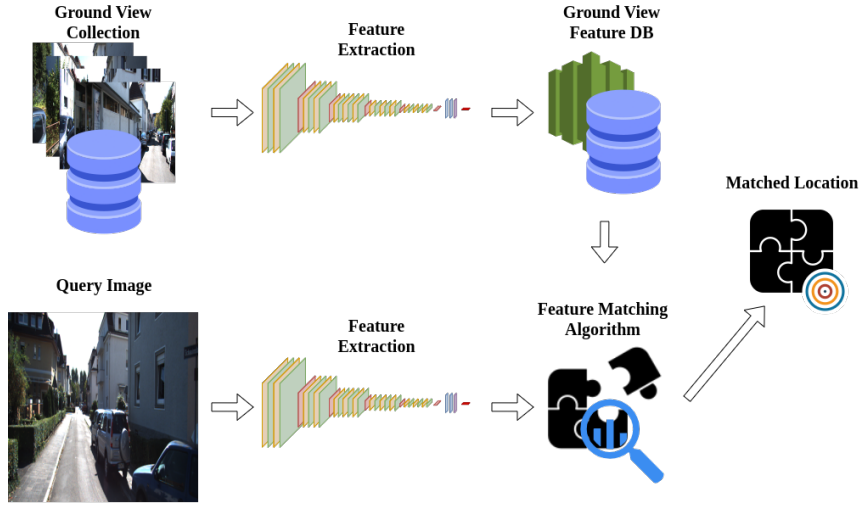


Figure 2.7: Framework for ground-level VL showing the general components needed to implement an IBL system.

First approaches for image based localisation employed hand crafted features (HCF) as the bases to match the query images to the geo-tagged image collection [27] [139] [114] [156]. Most of these methodologies are tested under controlled conditions, meaning the data sets used for development are captured mainly on urban areas and do not take into consideration seasonal or illumination changes, as was previously mentioned when analysing the existing datasets.

Although these approaches were successful for images sharing viewpoints, handcrafted features fail when matching images with drastic changes in viewpoint. The success of the methodologies generated interest for localising different types of mobile platforms using only the optical sensor. Platforms such as UAVs could benefit from an alternative localisation solution.

UAVs often perceive the environment from a birds-eye view, and ground vehicles are horizontally oriented to the Earth’s surface, also referred to as ground level images. Datasets can be collected from different platforms at different viewpoints, which opens the possibility of querying images against datasets captured from different perspectives. The difference in viewpoints makes VL a challenging problem to be generalised for all mobile platforms, hence it needs to be classified based on the viewpoint of the query image.

Comparing images captured at different viewpoints can lead to visual feature sparsity, preventing handcrafted features from finding common features, leading to failure. This

case is common with extreme viewpoints, such as comparing a ground level image against a satellite image.

In this section, VL is classified based on the platform and viewpoint at which the query image is captured. Ground view IBL defines the VL problem as the task of finding the geo-location of a query image taken from a platform operating on the surface of the Earth against a database of geo-tagged images from the same or similar viewpoint.

On the other hand, aerial VL refers to the IBL problem as the task of retrieving the geolocation of a query image taken from an aerial platform. The aerial localisation problem can be formulated in two different ways. One, with the query image taken from a camera pointing downwards, capturing an overhead view (nadir). Two, with the query image taken from a camera positioned at an arbitrary pitch angle, also called oblique image pointing downwards, as shown previously in Fig. 2.2. The VL problem requires to be formulated differently for the two scenarios, as the change in view point presents different challenges in the localisation workflow.

Another way of classifying the localisation problem is by cross-view localisation. In cross-view IBL, the problem is stated as the task of retrieving the geolocation of a ground level image on a georectified satellite image. The ground level query image is matched against satellite images to retrieve the correct location. The cross-view localisation problem presents particular challenges as the drastic change in viewpoint causes shared features to be occluded, making the problem difficult. In this case, matching images employing handcrafted features from both the query and database images is unachievable.

2.4 Ground View IBL

IBL is defined as the task of determining the geolocation of a query image by matching it against a database of images captured from similar viewpoints. The motivation for first examining ground-view localisation stems from the historical development of visual localisation. Early approaches to the IBL problem were formulated as image-retrieval tasks, and only later evolved into more complex multi-viewpoint retrieval methods. The techniques

that will be reviewed in this section are listed in Tab. 2.2, where each methodology has been classified by capture viewpoint and type of features, hand-crafted features (HCF) or learned features (LF).

GV Localisation Methodologies				
Researcher	Method	Viewpoint	HCF	LF
Cheng [27]	SIFT	GV	Yes	No
Torii [139]	SURF	GV	Yes	No
Sattler [114]	SfM	GV	Yes	No
Zamir [156]	GMCP	GV	Yes	No
Sünderhauf [131]	CNN	GV	No	Yes
Sattler [113]	GB	GV	Yes	No
Saurer [115]	DTM	GV	Yes	No
Arandjelović [7]	NET-VLAD	GV	No	Yes
Seymour [120]	SAANE	GV	No	Yes
Sarlin [111]	HF-Net	GV	No	Yes
Sarlin [112]	Superpoint	GV	No	Yes
[58]	SfM	GV	No	Yes
Brown [20]	Smooth-AP	GV	No	Yes
Cao [24]	GeM	GV	No	Yes
Hausler [49]	P-NetVLAD	GV	No	Yes
Vallina [76]	CNN GCL	GV	No	Yes
Berton [14]	CosPlace	GV	No	Yes
Ali-Bey [5]	MixVPR	GV	No	Yes

Table 2.2: Classification of methodologies for ground-view IBL based on the captured viewpoint; also identifies the feature-extraction technique as handcrafted features (HCF) or learned features (LF).

Cheng *et al.* [27] constructed a set of two databases (DB) of visual descriptors from collected panorama images around San Francisco. One DB corresponds to perspective central images (PCI) extracted from the panoramic image and the second DB corresponds to perspective frontal images (PFI). The PFIs are calculated using a vanish point algorithm that rectifies the PCI images. Each dataset consists of visual descriptors tagged over a large extent of the city. Utilising algorithms such as SIFT and RANSAC they run two separate processes to extract features and match the query image. The first methodology uses a vocabulary database to extract top match candidates. The second methodology, first rectifies the image utilising vanish points and then matches the rectified image towards the second database. This way, extracting the top matches of geographic locations takes into

consideration facades and perspective view. The methodology requires the query image to be rectified before the extracted features are matched against the databases. This makes the proposed method computationally inefficient. Additionally, the recall metric presents how noisy the algorithms are, resulting in a R@1 of approximately 10%.

Working with image maps structured as a graph, Torii *et al.* [139] formulate the image-localisation task as a regression problem. A graph methodology is proposed to predict the location of the query image. The graph consists of N vertices, D representing image descriptors and their corresponding locations. The Edges of the graph link the views that share visual characteristics. The features are described employing SURF [13] descriptors, which are used to find the matches and build the database. The query image descriptor is compared against the database descriptors to find a matching location. Once the top match has been identified, the algorithm interpolates the geo-location to estimate the query image location. The system is capable of achieving a mean average precision (mAP) of 75%. Despite the improved precision results, the complexity of the system is high, as it requires the building of a visual vocabulary and then using the descriptors to generate a graph model that works as a mapping function.

A different approach, presented in 2011 by Sattler *et al.* [114], localises an image towards a dense 3D reconstruction of an area instead of using a database of images. Employing the Structure From Motion (SfM) advances, a pre-generated 3D model is employed to geolocate a query image to the georeferenced point cloud. The 3D points are represented by the mean of the SIFT [86] descriptor. The SIFT image descriptors are extracted and assigned points from the 3D model, which are assigned to the words, generating a database of direct correspondence between visual words and 3D points. Employing a direct 2D to 3D descriptor matching correspondence is obtained, then a RANSAC algorithm is used to estimate the camera pose in the 3D model. This research shows the capabilities of image based localisation towards 3D models employing classical feature extraction methods are unknown and need to be further explored. Despite the unlocked capabilities of this technique multiple challenges are still latent for this technique to perform optimally. First, the database is complex to build as points need to be assigned to visual points to generate a database to locate against. Second, the method requires a RANSAC algorithm to estimate

the camera pose, which is computationally expensive.

With the current performance of Convolutional Neural Networks (CNN) at the time and the ability to extract features from scenery [45], Sünderhauf *et al.* implement features extracted from CNNs to the place recognition problem [131]. Therefore, introducing the ability of CNN features to maintain spatial characteristics in the feature space makes them suitable for place recognition. Proposing the image localisation problem once again as a matching problem, Sünderhauf *et al.* analyse specific CNN layers and their performance on matching scenery variations for the place recognition task. An ImageNet architecture is used as the backbone for the tests. ImageNet architecture consists of five convolution layers followed by three fully connected layers and one softmax layer. The output of each layer can be used as a holistic image feature extracted by the network. Sünderhauf *et al.* test the performance of the individual convolution layers on three different ground localisation datasets, Nordland, St Lucia and Garden Points. The results reveal that the third convolution features outperform the other layers, showing the relevance of network depth and the potential of convolution features for place recognition. Despite the deep study of the architecture for ground view localisation, a partial comparison with other state-of-the-art methodologies is presented, leaving a gap for further investigation on performance metrics such as recall.

Despite the breakthrough presented in 2015 in Australia by Sünderhauf *et al.* [131], classical methods continued to be researched for a period of time while machine learning methods were broadly accepted by the image localisation research community. A group in ETH Zurich present that geometric bursts (GB) can impact the performance of localisation recall, by inducing aliasing. Sattler *et al.* [113] state that for large scale datasets the assumption stating that “inliers found through spatial verification allow detecting image matches with high precision” does not apply [113]. The reason for the assumption to be false, is the perceptual aliasing induced by geometric bursts. As an example, buildings with multiple windows induce geometric bursts by introducing visual features of similar characteristic to other buildings that generates incorrect correspondences. To account for geometric bursts, Sattler *et al.* [113] propose an inter-image-weighted function that allows to down weight the noise induced by geometric bursts. Despite the improvement on

localisation performance, the proposed methodology employs handcrafted features. The tests are performed over a San Francisco dataset and the precision recall metric is used to compare against other state of the art methodologies that also employ hand crafted features.

Radenovic *et al.* [104] presents an approach that allows training CNN as unsupervised networks. The novel method does not require one to annotate the dataset, and achieves retrieval performances of up to 91% mean average precision. Radenovic et al. present a pooling layer that allows one to generate a joint multi scale representation of the local feature. Additionally, thanks to the use of features extracted via CNN's, the new approach outperforms NetVLAD and BoW among other methodologies both in efficiency and recall precision.

Following on the findings of Sünderhauf *et al.* related to the use of CNN features for place recognition, Arandjelović *et al.* [7] also encodes the ground view image features with a novel CNN architecture. After the CNN network, Arandjelović *et al.* explores adding a network of Vector of Locally Aggregated Descriptors (NetVLAD) layer, using the output as the localisation descriptor. The NetVLAD layer is one of the contributions of the paper, as it allows aggregating the first order statistics of the residuals in different parts of the descriptor space. As explained in the previous section, VLAD is employed for classifying handcrafted image features $\mathbf{x}_i$ based on a k-nearest neighbour algorithm that defines a set of clusters with fixed centers $\mathbf{c}_k$. Where the feature $\mathbf{x}_i \in \mathbb{R}^{(1 \times D)}$ has a dimension one by D . The VLAD algorithm sets the membership $a_k(\mathbf{x}_i)$ of a descriptor $\mathbf{x}_i$ to the k^{th} visual word as a fixed value of either 1 or 0. The value of 1 is assigned when the cluster $\mathbf{c}_k$ closest to the feature descriptor $\mathbf{x}_i$ and 0 otherwise. Instead, NetVLAD introduces a learnable parameter to the VLAD algorithm by assigning a weight $\mathbf{w}_k$ and bias b_k to each feature $\mathbf{x}_i$ depending on the proximity of the feature $\mathbf{x}_i$ to the cluster center $\mathbf{c}_k$. Making the membership $a_k(\mathbf{x}_i)$ differentiable $\bar{a}_k(\mathbf{x}_i)$, allows the parameters to be learnable, as shown in Eq. 2.5.

$$\bar{a}_k(\mathbf{x}_i) = \frac{e^{\mathbf{w}_k^T \mathbf{x}_i + b_k}}{\sum_{k'} e^{\mathbf{w}_{k'}^T \mathbf{x}_i + b_{k'}}} \quad (2.5)$$

The VLAD image representation $V(j, k)$ is shown in Eq. 2.6 as introduced by Jegou *et al.*

[64].

$$V(j, k) = \sum_{i=1}^N a_k(\mathbf{x}_i) (x_i(j) - c_k(j)) \quad (2.6)$$

The final NetVLAD layer presented by [7] is given by Eq. 2.7.

$$V(j, k) = \sum_{i=1}^N \frac{e^{\mathbf{w}_k^T \mathbf{x}_i + b_k}}{\sum_{k'} e^{\mathbf{w}_{k'}^T \mathbf{x}_i + b_{k'}}} (x_i(j) - c_k(j)) \quad (2.7)$$

Where the NetVLAD layer weights are described by Eq. 2.8.

$$\mathbf{w}_k = 2\alpha \mathbf{c}_k \quad (2.8)$$

And the NetVLAD layer biases are described by Eq. 2.9.

$$b_k = -\alpha \|\mathbf{c}_k\|^2 \quad (2.9)$$

Note that b_k is a scalar, and $\mathbf{w}_k, b_k, \mathbf{c}_k$ are learnable parameters. The three independent parameters $\mathbf{w}_k, b_k, \mathbf{c}_k$ adds flexibility to the original VLAD that only has one parameter $\mathbf{c}_k$.

The soft assignment is mathematically the same form of a softmax function presented in Eq. 2.10

$$\sigma_k(\mathbf{z}) = \frac{e^{\mathbf{z}_k}}{\sum_{k'} e^{\mathbf{z}_{k'}}} \quad (2.10)$$

Hence, the soft assignment of the array of descriptors $\mathbf{x}_i$ to the K clusters can be calculated in two steps. First, a convolution with K filters $\mathbf{w}_k \in \mathbb{R}^{1 \times 1}$ also referred as kernels of size one by one with biases b_k . The output of the convolution is given by Eq. 2.11.

$$s_k(\mathbf{x}_i) = \mathbf{w}_k^T \mathbf{x}_i + b_k \quad (2.11)$$

The soft assignment of the NetVLAD layer can be represented by Eq. 2.12 by substituting $\mathbf{z} = s_k(\mathbf{x}_i)$ into Eq. 2.10 resulting in Eq. 2.5.

$$\bar{a}_k(\mathbf{x}_i) = \sigma_k(s_k(\mathbf{x}_i)) \quad (2.12)$$

The NetVLAD feature representation output is of dimensions $(K \times D) \times 1$ after the $L2$ normalisations as per the original VLAD representation.

Before training the network, an initialization process is required to start the cluster centres. Features are extracted from the backbone chosen at the last convolution layer. The extracted features are used to generate cluster centres with a K-Means algorithm. Then the features are assigned to the corresponding cluster centres with a K-NN algorithm. With the centroids and the classification finished, initialization weights and bias can be calculated.

The network is trained with a contrastive loss employing the euclidean distance as the similarity metric. Where the Euclidean distance between NetVLAD features of the query image and positive and negative match candidates calculates a similarity metric between a query image and a positive and negative match pair. The network improves the recall metrics against simple CNN architectures such as AlexNet and VGG16 and outperforms the classical feature extraction methods.

Sarlin *et al.* [111] target the real time challenge for VL. A new neural network model composed of a MobileNet backbone and three heads is proposed. The system is capable of outperforming the inference speed of all the state-of-the-art algorithms. At the same time, it is capable of maintaining a state-of-the-art localisation metrics. The network architecture uses Superpoint decoder to extract local invariant key points and descriptors. Additionally, a NetVLAD layer is attached to the last layer of the MobileNet model to extract global features. The overall system generates a database of global indices from a collection of images by extracting the NetVLAD feature from each image through the proposed HF-Net. In parallel, a structure from motion model is built from the local features generated by the Superpoint decoder using the same collection of images. The query image is encoded using the pre-trained HF-Net and the extracted global descriptor is matched against the pre generated database of global descriptors with a NN search. Proposed poses with 6

degrees of freedom are utilised to match towards the query image local features. The HF-Net is trained in a teacher student network approach, allowing the system to be more compact and efficient. Cumulative distributions of position errors are presented as results for their localisation, where the novel methodology outperforms most of the state-of-the-art methodologies with fast computation allowing near-real time performance for image based localization. However, the training of such a model is complex and requires fine-tuning during experimentation.

Yu *et al.* [2] introduces a modification of the NetVLAD layer by exploiting the Spatial Pyramid structure of images. The novel methodology is called Spatial Pyramid Enhanced (SPE- NetVLAD). The method outperforms the NetVLAD method by 0.32 on recall@2 metric. Despite the improvement in performance, the complexity of the computation increases inference time, so it is worth evaluating the performance cost over the performance benefits that the methodology offers.

It has been proven that global descriptors are suitable for large scale localisation problems and that local features allow to match and compare individual scenes. Hausler *et al.* [49] introduce Patch-NetVLAD, a network that combines both local and global descriptors. The novel network produces a similarity score between two images. A hierarchical approach is applied, where first the original NetVLAD is applied and the top k descriptors representing the most likely matches are retrieved. After the first step, a new type of patch descriptor is computed and performs local matching of the patch level descriptors to re-order the original retrieved results. The combination of these two tasks makes the process more efficient. However, the research shows that Deep local and Global descriptor Network presented by Cao *et al.* [24] still outperforms Patch-NetVLAD in a VL task. The Recall@1 of NetVLAD is 88.7 while DELG [24] 90.0 and NetVLAD still performs as one of the highest state of the art methodologies with 85.1. A novel validation methodology is introduced by testing the training data extracted from Mapillary, as this data is collected by any civilian, showing an interesting comparison for generalization of the methodology. Additionally, NetVLAD is found to be one of the fastest methodologies, maintaining good retrieval performance while DELG is shown to take about 60s per query.

Berton *et al.* [14] approach the VL problem from a different angle. Instead of iterating over network architecture, a new training process is proposed. Where the data is subdivided into groups by fixing a minimum spatial separation in between classes. The specific training procedure allows one to train over multiple groups sparsely along the geographic area. The architecture choice is composed of a CNN backbone such as VGG16 or ResNet, followed by GeM pooling and a fully connected layer that outputs a feature vector of size 1×512 . The loss function chosen for experimentation is the Large Margin Cosine Loss (LCML) [144] also shown in the literature as cosFace. The methodology not only reduces the complexity of the architectures, but outperforms most of the state of the art methodologies with a recall@1 metric of 88.5 on the Pitts30K data set.

Ali-Bey *et al.* [5] attempt to implement an attention mechanism with Multi Layer Perceptrons called feature mixers. The methodology consists of extracting features with the CNN backbone and flattening the features before any normalisation. The features are fed to a cascade of fully connected layers that generate an automatic holistic aggregation. Feature-Mixer appears to replace pyramidal aggregation with the perceptive field provided by the number of connections. The Network is composed of a ResNet backbone cropped at the last second layer. The feature map is flattened and fed to the novel Feature-Mixer. The Feature mixer is composed of four layers: a normalisation layer, a fully connected layer, followed by a ReLU activation function, and a final fully connected layer. The input to the Feature Mixer is added to the output as a skip connection. The block is repeated four times and the final output is L2 normalised. The methodology is compared against other state-of-the-art frameworks and shows to outperform NetVLAD, CosPlace, and GeM among others, with a Recall@1 metric of 94.6, trained on the Pitts250k data set.

2.5 Aerial View IBL

As presented in ground view localisation, the VL challenge is often formulated as an image retrieval problem. For aerial VL (AVL) the problem can be formulated in the same manner. AVL has the caveat that can be separated into two types of VL based on viewpoint. First, where the UAV operates with the camera pointing nadirward. The second can be classified

as a type of cross-view localisation. This type of cross-view localisation is taken from an aerial perspective, where there is a pitch angle in the camera orientation. In this configuration, the query image is induced with objects in perspective, but at the same time has some aerial view of the area. In other words, the roofs of structures can still be visualised in the image as a perspective view. The techniques described in this section are listed in Table 2.3, where each methodology has been classified by capture viewpoint and type of features, hand-crafted features (HCF) or learned features (LF).

AV Localisation Methodologies				
Researcher	Method	Viewpoint	HCF	LF
Hou [51]	VAE	AV	No	Yes
Vanegas [140]	QR Tag	OAV	Yes	No
Jasrotia [61]	RFID	NA	No	No
Ding [36]	CTONN	OAV	No	Yes
Patel [102]	VT&R	AV	No	Yes
Bianchi [15]	VAE	AV	No	Yes
Dai [32]	Transf. Net	OAV	No	Yes
Wang [146]	Square Ring P.	OAV	No	Yes
Tian [134]	CGAN Synth.	OAV	No	Yes
Borgese [16]	Tether	NA	No	No
Lin [79]	USAM	OAV	No	Yes
George [44]	VIO	AV	No	Yes

Table 2.3: Methodologies implemented for aerial VL, classified by viewpoint, with the feature-extraction technique identified as handcrafted features (HCF) or learned features (LF).

Although Hou *et al.* [51] present the use of image feature extraction in the field of Facial Recognition, the methodology applied is relevant for Aerial View Localisation. Hou *et al.* [51] demonstrate the ability of Variational Auto Encoders (VAE) of extracting local features from images in an unsupervised manner. The VAE is composed of an encoder $E(x)$ that works as a function to compress an input x into a vector z of size $1 \times k$ features. Then, the decoder $D(z)$ decomposes the feature z into an estimate of the original input x .

The network receives an image I and computes the weights by minimising a loss between the input and the output. The purpose of the network is to reduce the dimensionality of

the input into a compressed feature space that is comparable in between features of the same space.

In search missions, UAVs are great tools due to their fast deployment and multiple viewpoints. Vanegas *et al.* [140] present a system to search a specific target in an obstacle cluttered GNSS denied environment using optical sensors. VL is utilised to estimate the current pose of the UAV in the environment in real time. The UAV estimates its position in the environment by detecting QR code tags with its onboard forward-looking camera. A flight controller is designed employing individual proportional, integral and derivative controllers to regulate the forward and lateral velocities, and to maintain constant altitude and azimuth. The forward and lateral velocities are estimated with optical flow using the downward looking camera. In addition, the azimuth is estimated from an onboard IMU and the altitude is estimated with the combination of an ultrasonic sensor, a barometer and the onboard IMU. To locate the target, the onboard camera looking in the direction of gravity detects the desired target. While the quad rotor searches the environment, the target moves stochastically in the environment with a target tag mounted on a remote control car. At the same time, the system compensates for position disturbances induced from wind through a Partially Observable Markov Decision Process (POMDP) algorithm that uses a reward system to represent sequential decisions. The specific algorithm utilised to solve POMDP is the adaptive belief tree (ABT), whose outputs are a sequence of pose references sent to the controller. Then the controller executes a sequence of state changes on the UAV until the target is found. After multiple iterations, the UAV is capable of finding the target.

The previous example demonstrates the need for image based localisation research. The experiments performed by Vanegas *et al.* [140] were carried out under controlled conditions, and the problem was simplified to develop control and search algorithms. However, in a real world application for autonomous search, GNSS technology is not enough, as many of the locations are GNSS denied.

Localisation technology is also required in wireless sensor networks (WSN), where data retrieval over large areas can be used for surveillance, disaster management and monitoring

of harsh environments. Jasrotia *et al.* [61] discuss the need of localisation technology over GNSS denied environments such as mining and remote locations. UAVs are attractive platforms for WSN systems thanks to their deployment flexibility and high mobility, aiding data sampling for environmental monitoring. Leveraging the field network deployed, Radio frequency identification technology is used as localisation support for adhoc stations during data retrieval. This supports the need to research IBL algorithms for multiple platforms, as RFID localisation technology can also suffer from signal loss and interference, causing inaccuracy in position estimation.

Tian *et al.* [135] addresses two important challenges in aerial IBL, cross-time image matching and orientation invariant image matching. A series of frameworks to deal with the different challenges are presented. The first framework focuses on addressing the cross-time image matching challenge. A Siamese network with a VGG16 backbone is proposed and trained on positive and negative pairs of satellite images from different time stamps. A contrastive loss function is implemented to teach the network a feature representation that discriminates between images captured from different locations, assuring they are far apart in the feature space, while images captured from the same locations at different times are close to each other in the latent space. In addition, to address the random orientation challenge on aerial images for IBL a different Siamese network is proposed, the Orientation Normalisation Network (ONN). Each branch of the Siamese network learns to predict the rotation angle for each image so that they are aligned.

Mutual information techniques and visual odometry are proposed by Patel *et al.* [102] to locate a UAV at a 26m altitude on a sunny day. Resulting in an accurate estimate of localisation and heading. Following on Patel *et al.* results and the findings of the findings of Hou *et al.*, Bianchi *et al.* [15] propose a real time system for UAV autonomous home return based on image localisation. A Variational Auto Encoder (VAE) is pre-trained with Google Earth cropped Satellite images in an unsupervised technique. The network is trained with ~ 48000 satellite images taken from a constant altitude. The VAE learns to encode the morphology of satellite images into a 1000 element vector $\mathbb{R}^{1000 \times 1}$. Once the training was complete, samples were taken every 0.5 m along the trajectory planned to survey. Additional samples were taken perpendicular to the path direction every 0.5 for 5m

on each side. Approximately 42 images per square meter are captured and extracted. Each sample is encoded, applying only the encoder block of the VAE. The database of encoded images is uploaded to the UAV as a local map of the area. The pre-trained encode block of the VAE is also uploaded to the UAV. The platform encodes its aerial live feed utilising the encoder and compares the 100 vectors against the database. After normalising the database matching the vectors with a weighted score approach, an estimate of the current location is calculated. The UAV is capable to return home following the path explored previously. Whilst, the experiment allowed the UAV to achieve the task, the system is very susceptible to lighting conditions, as shadows and illumination can cause the UAV to lose track of its location.

The operation of UAVs is not only under constant altitudes, sometimes due to terrain altitude changes and obstacles present in the path of UAVs altitude changes are required. These altitude changes need to be taken into consideration in the image-based localisation algorithms. Dai *et al.* [32] tackle this challenge starting with enhancing the UAV localisation datasets, introducing a new dataset named DenseUAV. The dataset is built by extracting data from Google satellite images at different zoom levels. In addition, Dai *et al.* [32] propose a metric to measure image retrieval that takes into consideration the distance between the query image and the compared images. Their motivation for this new metric was to improve the retrieval@K metric, which is calculated from binary labels. Besides the aforementioned contributions, the image features are extracted using a transformer network that applies the attention mechanism, enabling understanding of the global semantic context.

Taking into consideration the multi view challenge present in drone image based localisation, Wang *et al.* [146] approach the feature extraction process by applying a square ring partition approach. In the same manner as all the VL task, the system starts with a database of ground level images, satellite images and drone level images taken with an unknown pitch angle (not looking down). To extract features from the images, a three branch network with ResNet-50 as a backbone is proposed. The branches that process satellite images and drone images are implemented with shared weights. For every branch output, the feature vector is sliced into four feature parts, corresponding to four different square sections of

the image. Once sliced, an average pooling layer is applied to reshape all the features in the same dimensions. With the features obtained from different sources, a classification layer composed of a fully connected (FC) layer, a batch normalisation (BN) layer, followed by a dropout layer and a final classification fully connected (FC) layer. After the classification module, the final feature embeddings are extracted. The square ring slice in addition to the classification module work together as a transformation module that aligns the different view points in feature space for comparison. For training, a cross entropy loss function is formulated based on a probability measurement, stating the probability of match between query image and geotagged image. The results revealed that the proposed methodology outperformed the state of the art systems such as CVM-Net and the GAN approach proposed by Regmi *et al.* [106].

As previously mentioned, VL is highly dependent on the image matching process, making discriminative feature extraction the key aim to find the right image match. If the features extracted can not be compared between different viewpoints, the retrieval task becomes very challenging. Lin *et al.* [79] focus their research on improving the feature extraction methodology for images taken from aerial platforms. A new framework called RK-Net which learns representation and detects key points with one network. An attention mechanism named Unit Subtraction Attention Module (USAM) is introduced as the main module. The main operation performed in USAM is named Unit Subtraction Convolution (USC). USC detects key points, allowing the comparison between viewpoints. The complete framework contains a ResNet-50 backbone with major modifications. The first modification is made after the first and second layer of the ResNet-50. After layer one and two a USAM module is added. The third modification changes the classification module by adding a fully connected layer, a batch normalisation and a classification layer.

As mentioned before, the USAM block is used twice in the ResNet architecture. One time after the first convolutional layer and the second after the second convolution layer. For training of the network, an instance loss function is implemented. A stochastic gradient descent algorithm is used to optimise the network weights. The final results reveal that the new methodology outperforms most of the state-of-the-art techniques, except for the Polar transformation proposed by Shi *et al.* [124] and the transformer methodology proposed by

Yang *et al.* [155]. Methodologies that are discussed in detail later.

Different to all the research presented with mono localisation, George *et al.* [44] explores the idea of using stereo cameras for VL. A DJI Matrix 600 is mounted with a set of stereo cameras pointing downwards and the system flies on an 800 m long trajectory. Three methodologies are tested, stereo Visual Odometry (VO), mono visual inertial Odometry (VIO) and stereo (VIO). The experiments are performed under three different weather conditions (gentle breeze, moderate breeze cloudy, strong breeze cloudy), four different speeds (2, 3 and 4 m/s) and four different altitudes (40, 60, 80 and 100 m). An RTK system was utilised to measure the ground truth position of the vehicle during the data collection and was used to present the results of the VL system as absolute trajectory error (ATE) for position and relative error (RE) for orientation. Results show the best results were obtained at altitudes of 40 m and 60 m. However, scale error persisted for both stereo VO and mono VIO. Stereo VIO methodology reduces the scale error to a 2% scale drift.

2.6 Cross-View IBL

Cross-view geolocalisation is defined as the task of estimating the geolocation of a query image by matching the corresponding pair to a georeferenced data collection of images [152, 142, 67, 155, 39]. The most common case of cross-view IBL collects the database images from georeferenced satellite images, while the query image is normally a ground-level image. However, cross-view localisation has also been defined as the task of geolocating an aerial image not captured in the nadir direction [134, 146]. Instead, the query aerial image captures the perspective of the scene, making by nature a different problem. The difficulty of the retrieval task changes due to the shared viewpoints between aerial photographs and satellite images [134, 146]. On the contrary, horizontal and satellite images do not capture the same visual features due to occlusion and sparsity, which increases the difficulty for matching algorithms. Table 2.4 details the different methodologies covered in this section is presented, where techniques are classified by recall metrics, the datasets used for training and testing, and the overall methodology’s complexity.

CV IBL Methodologies					
Method	Pub.	R@1%	R@1	DS	Complexity
Vis. Vocab. [87]	ACM'08	-	-	-	High
Vis. Vocab. [12]	ACM'11	-	-	-	High
CNN [25]	ICCV'15	-	-	-	High
Where-CNN [81]	CVPR'15	-	-	-	High
MCVplaces [152]	ICCV'15	34.3	-	CVUSA	Moderate
DBL-Net [142]	ECCV'16	63.7	-	CVUSA	Moderate
ModVGG16 [67]	ICRA'17	20	-	KITTI	Moderate
CVST [158]	CVPR'17	43.2	-	CVUSA	High
CVM-Net I [54]	CVPR'18	91.4	22.53	CVUSA	Moderate
UV-Map [82]	CVPR'19	93.19	31.7	CVUSA	Moderate
Feature Fusion [106]	ICCV'19	95.98	48.75	CVUSA	High
SAFA [122]	NeurIPS'19	99.49	81.15	CVUSA	High
Polar+SAFA [122]	NeurIPS'19	99.64	89.84	CVUSA	High
VGG16 [122]	NeurIPS'19	96.38	39.72	CVUSA	Moderate
Polar+VGG16 [122]	NeurIPS'19	98.3	65.74	CVUSA	Moderate
Polar+CVM-Net [122]	NeurIPS'19	97.82	50	CVUSA	Moderate
CVFT [123]	CVPR'20	99.02	61.43	CVUSA	High
Polar+VGG16+DSM [124]	AAAI'20	99.67	91.96	CVUSA	Moderate
CVM-Net I [55]	IJCV'20	96.3	22.53	CVUSA	Moderate
Siam. ImageNet [55]	IJCV'20	4.7	-	CVUSA	Moderate
Siam. VGG16 [55]	IJCV'20	9.9	-	CVUSA	Moderate
EgoTR [155]	NeurIPS'21	99.75	91.99	CVUSA	High
Polar+EgoTR [155]	NeurIPS'21	99.67	94.05	CVUSA	High
ResNet-50 + SRP [146]	TCSVT'22	99.41	85.79	CVUSA	High
SAFA+SRP [146]	TCSVT'22	99.78	92.83	CVUSA	Very High
SAFA+USAM [79]	TIP'22	90.16	99.67	CVUSA	High
LPN+USAM [79]	TIP'22	99.67	91.22	CVUSA	Very High
GAMa-Net [143]	ECCV'22	91.9	15.2	GAMa	
WAG [38]	IROS'22	-	-	WAG	Very High
ReWAG [39]	ICRA'23	-	-	ReWAG	Very High

Table 2.4: A representative selection of methodologies for cross-view localisation and their performance on the CVUSA dataset.

Early works on cross-view IBL are based on similar techniques used for ground view VL. Originally, the features to describe images were extracted employing handcrafted features such as the previously discussed SIFT, SURF, FAST, among others. Luo *et al.* [87] utilised SIFT features obtained from a collection of satellite and ground level images to generate colour visual vocabularies to characterise both viewpoint images. Their aim is to classify events and environments based on a combination of satellite and ground level images. AdaBoost engine is used to train the satellite images classifier and Support Vector Machines

(SVM) for the ground level images classification. On their research, Luo *et al.* identify one of the most essential challenges in VL which is caused by seasonal and weather changes. These changes impact the environmental scenery and can cause the trained machines to fail when recognizing locations. As being one of the pioneering approaches for cross view localisation, Luo *et al.* [87] attempt to solve the problem by classifying environments into eleven categories. It is shown that by performing score level fusion, which uses the output score of each classifier that is fed to a meta classifier to obtain a better event prediction, they are capable of classifying scenes correctly with an overall recall of 0.724, a precision of 0.713 and an F-score of 0.7. Although the classification approach can help narrow down possible matches and identify the event, it is not possible to obtain an exact location based on these methodologies.

Bansal *et al.* [12] leverages the availability of oblique aerial data to build a visual vocabulary of structure facades. An edge detector is utilised to extract the facades, which are automatically geotagged by aligning roof edges with the corresponding satellite image. Due to large viewpoint and appearance differences between aerial images and street view images, handcrafted features such as SIFT or MSER are not feasible for image matching. To account for this, features from each facade are extracted by sampling a uniform grid of points over the facade. At each point, a self similarity descriptor is computed, and the visual descriptors are organised in an Adaptive Vocabulary Tree (ADT). Once the ADT has been built, it is employed to perform ground view geo-localisation at run-time. To extract the features from the ground level image, the same procedure is applied; first, a facade is extracted, and similarity features from the facades are computed. These self similarity features are queried against the ADT. This discovery makes this research one of the pioneers in applying cross-view geo-localisation as an alternative methodology to GNNS. Although the methodology shows the possibilities for cross-view geo-localisation, it highly relies on aerial oblique images to extract the visual vocabulary. Leaving the challenge of localising a horizontal image using only satellite images unsolved. In addition, no retrieval metrics were utilised to test the performance of the methodology.

Although, feature extraction algorithms relying on hand crafted features have been tested successfully, their computational demand does not allow the systems to perform in real

time conditions. The use of CNNs shown in 2012 [69], revealed the abilities of artificial intelligence to recognise patterns in images. To improve speed and accuracy, researches have adopted Convolutional Neural Networks (CNN)s as the main method for image feature extraction.

In an attempt to localise a ground level image, Castaldo *et al.* [25] applies semantic segmentation to extract scenery features as roads, vegetation and buildings. The segmented image is projected to a birds-eye view to be compared against a GIS map. Now that both images are in a similar view point, a Semantic Segment Layout (SSL) descriptor is extracted from both images to be compared. The SSL descriptor encodes into the rough geometric layout the semantic concepts present on each image. Applying the Bhattacharyya distance between descriptors, each descriptor is treated as a Gaussian Mixture Model to obtain a similarity metric. The novel framework generates a heatmap of likely locations where the ground level image was captured. The system is capable of analysing large areas, and it is tested on the whole District of Columbia which cover ($159km^2$). The methodology allows the reduction of the search area to less than 5% of the original search space. Despite the reduction in search area, the methodology is not very precise and generates aliasing when the environments have similar characteristics.

Lin *et al.* [81] uses panoramic images extracted from Google Street views with their corresponding satellite view. For the first time, they propose a cross-view dataset that is used simultaneously to learn features from both viewpoints at the same time. A Siamese neural network is employed to learn the feature representation in a shared feature space for both satellite and ground view images. Using a modified version of ImageNet and a contrastive loss, both satellite and ground level images are fed to the network to learn the shared feature representation. The architecture and methodology is inspired from the advances made in facial recognition. In other words, Lin *et al.* formulate the cross view localisation problem as an identity verification task. The novel methodology demonstrates that a common feature space can be learned from the distinct view points. To match the ground level images against the satellite images, a k-nearest neighbour methodology is utilised to compare features between both. The methodology shows advantages over the state-of-the-art techniques in precision, as the images can now be compared in the

same feature space. It also demonstrates the advantages of employing learned common features instead of using handcrafted features. Although the methodology is successful when facades are present in the satellite image, aliasing is still present when the area of search contains many similar visual characteristics or fails when facades are not present due to visual occlusion.

As aforementioned, the work of Workman *et al.* [152] introduces one of the most important datasets for cross view localisation CVUSA. In this work two novel architectures CVPlaces and MCVPlaces are proposed. The CVPlaces networks utilise ImageNet as the backbone and is trained with the CVUSA dataset. The loss function chosen to train the network is a mean square error metric. To account for multiscale resolution present in the satellite images, MCVPlaces architecture is proposed. MCVPlaces is composed of three single scale CVPlaces networks, each branch takes as an input a different scale satellite image and concatenates the output features from each branch. The concatenated feature is then fed to a classification network with 205 dimensions. The novel framework allowed Workman *et al.* [152] to generate localisation estimates around the whole USA. Despite the size of the networks and large scale of the dataset introduced, their methodology allows to estimate cities and generalised locations where the query image might be captured. However, the system does not allow to estimate a location within 100 meter error.

Leveraging the advantages of CNNs, Vo *et al.* [142] experiment with six different network architectures to perform corss view localisation. In addition to the neural networks a novel database is introduce to train the new techniques. ImageNet is utilised as the backbone for feature extraction, and it is arranged in six different ways for the cross-view localisation task. The first approach concatenates the satellite and ground view images in the channel dimension and feeds them to a modified ImageNet backbone. A binary label is used to classify the input as a “match” or a “no-match”. To train the network a SoftMax logarithmic loss function is implemented as shown in Eq. 2.13.

$$L(A, B, l) = \text{LogLossSoftMax}(f(I), l) \quad (2.13)$$

Where l is the label indicating a binary match $l \in 0, 1$. A and B are the input images, I is

the concatenation of both A and B , $I = \text{concatenation}(A, B)$. $f(.)$ represents the output of the ImageNet network.

The second architecture tested by Vo *et al.* is a Siamese network with ImageNet as backbone, that does not share weights between branches. The rationale for not sharing weights is given by the drastic difference in viewpoints of each image. This allows each branch to learn different weights to represent the features in a shared feature space. The features from each branch are compared between each other to determine a match between images. During training, a contrastive loss shown in Eq. 2.14 is used to optimize the network weights.

$$L(A, B, l) = l * D + (1 - l) * \max(0, m - D) \quad (2.14)$$

Where D represents the square distance between $f(A)$ and $f(B)$, m is a margin avoiding penalization when the distance of a non-matched pair is not large enough.

The third architecture presented is a Siamese classification hybrid network. This network is composed of two parts, a feature extraction stage f_{conv} and a classification stage f_{fc} . Features are first extracted individually for each image pair by an ImageNet branch. Then the features are concatenated and fed to a classification network made of fully connected layers. The loss function occupied to train the network is shown in Eq. 2.15.

$$L(A, B, l) = \text{LogLossSoftMax}(f_{fc}(I_{conv}), l) \quad (2.15)$$

Where $I_{conv} = \text{concatenation}(f_{conv}(A), f_{conv}(B))$.

The fourth architecture proposed is a triplet network which consists of three branches of ImageNet backbone. Two of the branches encodes the positive and negative satellite view pairs and the other one encodes the ground level image. In other words, the network takes three images, ground level view A and satellite images B, C . (A, B) are considered corresponding images, meaning that they are a match and (A, C) are negative pairs. To train the network a triplet margin loss shown in Eq. 3.1 is utilised which is also found in the literature as hinge loss.

$$L(A, B, C) = \max(0, m + D(f(A), f(B)) - D(f(A), f(C))) \quad (2.16)$$

Where D represents the square distance in between features and m represents the margin that avoids penalization if the difference between distances is large enough. The loss function aims to place the matching images closer in feature space and the negative pairs further away in the feature space.

The fifth architecture just replaces the contrastive loss of the Siamese network with a distance based logistic (DBL) loss formulation as shown in Eq. 2.17 and maintains the Siamese network.

$$L(A, B, l) = \text{LogLoss}(p(A, B), l) \quad (2.17)$$

Where $p(A, B)$ is defined in Eq. 2.18 as the probability of match given the squared distance.

$$p(A, B) = \frac{1 + e^{-m}}{1 + e^{D-m}} \quad (2.18)$$

The final architecture proposed maintains the triplet branch architecture and introduces a DBL loss function for the triplet network, as shown in Eq. 2.19. The DBL-log loss can be interpreted as a smoothed version of the hinge loss.

$$L(A, B, C) = \log(1 + e^{D(f(A), f(B)) - D(f(A), f(C))}) \quad (2.19)$$

Where $p(A, B, C)$ represents the probability of a valid triple as shown in Eq. 2.20.

$$p(A, B, C) = \frac{1}{1 + e^{D(f(A), f(B)) - D(f(A), f(C))}} \quad (2.20)$$

The networks are trained over batches of 128 images. After training the different architectures, the results show that the triplet network with DBL out-performs the Siamese network with higher accuracy in image matching. A recall at top 1% for the Triplet DBL-Net is reported at 49.3%. Additionally, to the networks tested, rotational invariance is also

explored during their research. Vo *et al.* [142] present one of the most important milestones for cross view matching, as the tests performed showed the importance of Siamese and triplet networks architectures for the cross view matching task. In addition, the networks presented are complex but intuitive for implementation. Despite the novel dataset and multiple architectures presented, the recall metrics presented need to be improved to achieve robust cross-view localisation.

Kim *et al.* [67] revisits the cross-view localisation task by adding a particle filter to the framework. A modified VGG16 is used as the backbone for a Siamese network that extracts local features from both ground and satellite images. The network is trained by applying the contrastive loss shown in Eq. 2.14, which penalises the network when the distance between the negative pair images is greater than a margin m . The similarity metric between image features is calculated as the Euclidean distance between the encoded features. Similar to the work previously presented by Vo *et al.* [142]. $\sim 540\text{K}$ image pairs are used for training, where 53k are positive pairs and 438K non matching pairs. A detailed explanation of the particle filter is presented in the following section, as this concept is then used in the following research methodologies.

2.6.1 Particle Filter

A particle filter is a Markov Chain Monte Carlo (MCMC) method used for leveraging stochastic sampling to estimate nonlinear probability distributions [133, 40]. The particle filter is an iterative process that attempts to estimate the pose of a moving vehicle and is a very common technique used in robotic systems. Given a set of measurements $u^t = \{u_0, u_1, \dots, u_t\}$ and a set of observations $z^t = \{z_0, z_1, \dots, z_t\}$ over a defined time period t , the filter estimates the probability of the vehicles pose x_t as shown in Eq. 2.21

$$p(x_t | u^t, z^t) \tag{2.21}$$

u^t measurements are collected from proprioceptive sensors that measure angular rate and forward velocity. The observations z^t are calculated as the Euclidean distance between the

current ground level feature $f(I_g)$ and the set of random satellite features $f(\mathcal{I}_s)$. Where $\mathcal{I}_s$ is the set of stochastic satellite images and $f(I)$ denotes the extracted image features through the VGG16 backbone. The posterior probability distribution is usually multimodal and nonlinear. Leveraging random sampling, a set of n particles $\mathcal{P}_t$ is uniformly distributed over a satellite area where the vehicle operates.

$$\mathcal{P}_t = \{P_t^{(1)}, P_t^{(2)}, \dots, P_t^{(n)}\} \quad (2.22)$$

$$\mathcal{I}_t = \{I_s^{(1)}, I_s^{(2)}, \dots, I_s^{(n)}\} \quad (2.23)$$

Each particle is composed of a pose $x_t^{(i)}$ and importance weight $w_t^{(i)}$, and each satellite image $I_s^{(i)}$ is sampled based on each particle's pose. The posterior distribution shown in Eq. 2.24 is updated as new odometry measurements and ground-level images are captured.

$$p(x_{t-1}|u^{t-1}, z^{t-1}) \quad (2.24)$$

Knowing a previous distribution at the vehicles pose x_{t-1} at time $t - 1$ represented by Eq. 2.24, the prior distribution at pose x_t is estimated by sampling from the probability of the current pose x_t given the pose at time $t - 1$, the set of proprioceptive measurements u_t and the set observations z_{t-1} , $p(x_t|x_{t-1}, u_t, z_{t-1})$.

As the next step, a new set of particles is proposed by propagating each particle pose forward accordingly to the motion model $p(x_t|x_{t-1}, u_t, z_{t-1})$ modelled as a Gaussian. The new importance weights of the particles are calculated with Eq. 2.25.

$$\tilde{w}_t^{(i)} = p(z_t|x_t, u^t, z^{t-1}) \cdot w_{t-1}^{(i)} \quad (2.25)$$

The probability of the observation z_t given the current vehicle's pose, the current set of proprioceptive measurements u_t and the set of observations z^{t-1} , is modelled as an exponential distribution shown in Eq. ???. This is also known as the likelihood of the current observations. α is a parameter finely tuned to a validation dataset.

$$p(z_t|x_t, u^t, z^{t-1}) = \alpha e^{-\alpha d(f(I_g), f(I_s))} \quad (2.26)$$

Once the importance weights are updated, a systematic resampling process is then followed to avoid particle depletion.

The particle filter uses the siamese network as the measurement mechanism to identify possible locations. A driving vehicle is simulated with the KITTI Data Set sequences to extract both the odometry measurements and consecutive ground level images. Utilising 5000 randomly initialised particles on a defined area with equal importance weights. The filter calculates the Euclidean distance between the current ground level image and the 5000 proposed satellite images. Through the iterative process, the novel system is able to estimate the location of the ground vehicle during movement by filtering aliasing with the particle filter. Although the localisation of the vehicle can be accurate, the particle filter is computationally inefficient and requires a large amount of particles to map the unknown nonlinear distribution. Additionally, the particle filter requires a very precise tuning process to operate and some times can fall in a local minima during the estimation. The Siamese network recall metric also affects the localisation performance as it is very noisy and generates aliasing, hence the particle filter is required.

Zhai *et al.* [158] utilise semantic segmentation to extract semantically meaningful aerial features and compare them with a segmented ground level panoramic image. This methodology enables better orientation prediction for cross-view IBL.

Hu *et al.* [54] propose a CNN siamese network with a VGG16 backbone named Cross-View Matching Network (CVM-Net). The network applies a NetVLAD after the convolution blocks of each branch instead of the MaxPolling layers present in the VGG16 architecture. The Convolution blocks extract local features from the images, and the NetVLAD calculates an encoding of global descriptors. Adding the NetVLAD block allows the network to localise over large areas and increases the recall metric of the methodology. Training is performed with a weighted soft-margin ranking loss shown in Eq. 5.9 and a quadruplet loss. Both loss functions employ the Euclidean distance as the similarity metric between embeddings.

The quadruplet loss is calculated with a soft logarithmic margin loss, outperforming the triplet and contrastive loss.

$$\mathcal{L}_{weighted} = \ln(1 + e^{-\alpha d(f(I_g), f(I_s))}) \quad (2.27)$$

CVM-Net is a compact solution to the cross-view localisation problem. The network architecture requires fewer parameters for training, meaning it performs faster during inference. Additionally, the improvement in recall is evident with the introduction of global features into the solution. Although the method proposed by Arandjelovic *et al.* [7] requires a clustering initialisation process for the NetVLAD layer, CVM-Net initialises all the parameters of NetVLAD randomly. For training, α in the loss function is chosen to be 100 and the learning rate is chosen to be 10^{-5} . Despite the random initialisation, the methodology outperforms most of the state-of-the-art frameworks.

In the field of Image generation, Generative Adversarial Networks (GANs) have been proven to be effective in reconstructing high-definition human faces. Regmi *et al.* [105] generate synthetic satellite images from ground view images by implementing the pix2pix model [59]. This research shows how GANs can be used to predict aerial scenery of hidden views by object occlusion. A new architecture called X-Fork is proposed and trained to generate synthetic street view images from an aerial view or aerial views from ground-level images. Additionally, the novel architecture is capable of learning semantic segmentation, which allows not only to generate the synthetic view but also to generate a segmented representation of the estimated view. The methodology presents very realistic synthetic satellite views generated from a ground level image. Despite the results, the complexity of the GAN training is high and involves multiple modules for the training.

Following on their previous research, Regmi *et al.* [106] extract features from the synthetic aerial view and a panoramic ground image to obtain a robust representation of the query image. The robust feature is calculated by fusing both encodings with a fully connected network. The ground truth satellite image is matched against real satellite images, calculating the Euclidean distance between features and applying a threshold which determines a positive match. Two different cross-view localisation architectures are

proposed, Joint Feature Learning and Feature fusion. Both novel architectures outperform the state-of-the-art CVM-Net at the Recall@k metric, showing one of the best recall metrics as shown in Tab. 2.4. Despite the high R@k metric presented, the methodology involves a generator module that first needs to synthesise every ground level image to then extract features from the query ground level image, the synthetic aerial view and a proposed aerial view through a VGG network. The features are then classified by a fully connected network before being compared. This long process is computationally expensive, not to mention the complexity required for training, as multiple hyperparameters need to be tuned for the different loss functions proposed.

After noticing that most of the research in IBL adopted an approach to solve the problem of extracting features from images to then compare similarity between them, Liu *et al.* [82] incorporates orientation into the neural network training. Similar to teaching the network a sense of orientation. A CNN backbone architecture is also utilised to extract feature in a Siamese configuration. The Siamese branches do not share weights between the ground view branch and the satellite view branch. To incorporate the orientation feature into the training process, the Boltzmann Network [157] was used as inspiration for the novel orientation methodology. A simple approach is taken to inject individual pixel orientation into the network using orientation maps. The idea represents an orientation taken from the ground truth and is concatenated as a mask to the ground and satellite image. Two approaches for the concatenation are taken, one that concatenates the UV map after every layer output of the convolution. The second one just concatenates the UV map to the input image. To train the network, a soft margin triplet loss is applied to the optimisation process. The results show that exploiting the orientation of the images outperforms the state-of-the-art methodologies, such as CVM-Net proposed by Hu *et al.* [54]. The advantage of this methodology over the feature fusion technique presented by Regmi *et al.* [106] is the complexity reduction of the architecture. Liu *et al.* present a much simpler and more compact network for training. The architecture, being a compact Siamese network, reduces the computational time during inference at the expense of 2% recall.

Extending on their previous work, Hu *et al.* [55] leverage their findings of CVM-Net to

implement a near-real time localisation system that only uses images for geo-localisation. Similar to Kim *et al.* [67] approach, a Markov Localisation process is employed to enforce temporal consistency of the trajectory. The odometry measurements are performed with Visual Odometry (VO), meaning that the system only requires images to operate. The Markov model uses the distance measurement between the query image and the satellite image database to filter the proposed particles, while the VO calculations propagate the particles forward. Hu *et al.* test multiple backbone architectures for the CVM-NET and VGG16, which outperform ResNet-50, DenseNet-121 and Xception. After extensive experimentation, a near-real time IBL system has been achieved with an approximate 20m accuracy. The addition of the particle filter removes the aliasing induced by the noise measurement of the siamese network, as shown previously by Kim *et al.* [67].

Shi *et al.* [122] present in 2019 one of the most elegant solutions to the cross view IBL challenge. Formulating the cross view problem, taking into consideration the difference in geometric configuration between ground and aerial view, a polar transformation is applied to the aerial view in order to increase the spatial correspondences between the image pixels. Both the transformed aerial image and the ground level view are processed through a Siamese CNN network (VGG19) that extracts the image features. An attention mechanism is added in sequence to the VGG19 network, which is designed to encode the relative positions among object features. The attention mechanism utilised is called Spatial Aware Feature Aggregation (SAFA). SAFA is composed of a max-pooling operator along feature channels, selecting the most distinctive features. Then it performs a spatial aware importance generator to obtain a position embedding map. The spatial aware importance generator is composed of two fully connected layers. The novel SAFA module incorporates a type of attention mechanism that discriminates features based on importance and relevance. Finally, to train the network, a soft margin triplet loss was utilised as presented in Eq. 5.9. After training and evaluation, the results revealed that the new methodology outperformed all the state-of-the-art approaches, including the novel CVM-Net proposed by Hu *et al.* [54] and the methodology proposed by Liu *et al.* [82]. Additionally, the complexity of the method is much lower than the one presented by Regmi *et al.* [106], making the system feasible for near-real time applications.

Following on their previous work, Shi *et al.* [123] continue to improve over their polar transformation implementation and SAFA contribution. The backbone of the siamese network is reduced to a VGG16 architecture that is modified in the last three layers. Only the last three layers of the VGG16 network are learned, while the others use fixed pretrained weights from ImageNet. This time, a Dynamics Similarity Matching (DSM) approach is added after the modified VGG16. The DSM approach consists of extracting the features from the ground level image along with the features of the polar transformation of the aerial image. Assuming that the ground level image orientation is unknown, a correlation metric is computed between the features of the aerial image and the features of the ground level image. Utilising the ground-level image features in a sliding window approach makes it possible to retrieve the unknown orientation of the ground level image. Once the orientation has been set, the aerial features of the polar transformed image are cropped to match the ones of the ground level image features. The cropped features are then processed by a soft margin triplet loss during training. The improved framework increases the recall metric and outperforms all the state-of-the-art methodologies. The complexity of training is increased caused by alterations to the architectures are applied. However, it shows that reducing the viewpoint difference with a transformation process increases the performance of the localisation with a low computational cost. The network architecture is compact and would be interesting to test in conjunction with a particle filter.

Approaching the cross-view localisation problem from a different perspective, the group of Vijay Kumar and Miller *et al.* [95] implement a localisation system that leverages semantic information from a multi sensor platform. However, LiDAR data is utilised to solve the cross-view localisation problem this time. Although an optic sensor is employed for the VL task, the methodology developed is relevant to the image-based localisation field. First, semantic information is extracted from a priori collection of satellite images and LiDAR scans. Each semantic representation is transformed accordingly to their viewpoint before they are compared. The LiDAR semantic segmentation is projected to a ground view, and the satellite semantic representation is projected via a polar transformation to match the same viewpoint as the LiDAR data.

To extract the semantic segmentation from the satellite view, Miller *et al.* implement a Fully Convolutional Network (FCN), with a ResNet-34 as backbone to encode the images. The LiDAR semantic segmentation is performed with two different methodologies depending on the dataset. The first one applies a projection of an FCN as well. The LiDAR semantic segmentation is trained with two different methodologies, depending on the dataset that was being evaluated. For the KITTI dataset, the same FCN structure as the satellite segmentation is applied. The LiDAR data is represented as a 2D Polar grid map with x,y,z and depth channels for training. For the Morgantown dataset, RGB images calibrate to the LiDAR scans are segmented with HRNets. The segmentation from the RGB image is then projected to each point of the scan.

For the localisation module, a particle filter is applied due to their ability to handle multimodal distributions. The particle filter is coupled with a Simultaneous Localisation and Mapping (SLAM) system named UPSLAM. UPSLAM receives a pose prediction from the particle filter and returns odometry measurements to the particle filter. While the system localises the platform globally, UPSLAM relies on the Iterative Closest Point (ICP) algorithm to generate the map. The system integrated not only localises the platform, but also segments the environment in both LiDAR and satellite images while generating a map. Although the system provides localisation with less than 5m error, it relies heavily on LiDAR scans to run the SLAM algorithm, which corrects over time. Leaving the open research topics for IBL.

The successful results of Transformer networks in Natural Language processing caught the attention of researchers, leading to the exploration of transformers in computer vision [66] [83]. Yang *et al.* [155] propose to apply a transformer head after a ResNet architecture, which extracts CNN features. Evolving geo-localisation Transformer (ErgoTR) introduces multi head self cross attention as a feature encoder that outperforms the state-of-the-art approaches presented by [106] [142] [54] [123] among others as shown in Tab. 2.4. However, inference times are not measured, preventing the evaluation of whether the architecture is capable of performing near-real time tasks. Considering that Visual Transformers require a large number of training parameters to operate [78], adding them to a feature extraction architecture makes the framework have a high computational cost.

Vyas *et al.* [143] use visual localisation techniques to geo-localise a video. Introducing a dataset of Ground video to aerial image Matching (GAMa), a novel GAMa-Net is trained with a video and its corresponding satellite image. The network is composed of two branches, one that extracts features from the satellite image, and the parallel branch that extracts features from a video. The video clip branch receives 8 frames as input, which are processed by a 3D CNN ResNet18 encoder called Ground Video Encoder (GVE). The video features extracted by the GVE are then processed by an MLP projection head with one hidden layer. The aerial image branch extracts local features from the image with a 2D ResNet18 CNN. The features are then classified based on importance by a transformer network made of 4 blocks and 2 layers. The output of the transformer is then projected by the projection MLP network to output the aerial feature vector. The network is trained with a contrastive loss inspired by NT-Xtent [28]. The methodology presents a framework that allows the localisation of a video to aerial images with a novel bi-branched network employing 3D CNN. Despite the novelty of the idea, the ablation study does not compare to all the state-of-the-art architectures, leaving room to improve video to satellite image localisation. Additionally, GAMa-Net also incorporates a transformer network into the methodology, increasing the number of training parameters that can reduce the speed of the framework.

Downes *et al.* [38] leverage the high recall metrics attained by using panoramic images in the recall process of IBL to present a city wide localisation system. The complete solution is named Wide-Area Geolocalisation (WAG) and takes after the work presented by Kim *et al.* [67]. The contributions of this research introduce an improved likelihood probability density function and a new training process that integrates a tiling sampling for satellite images with a novel loss function formulation. To extract the embeddings Downes *et al.* uses the neural network developed by Zhu *et al.* [166]. Even though the system presents an improvement over the similarity computed between panoramic images and satellite images, this technique requires a significant amount of distance to converge on the correct location and does not operate in near-real time.

Next year Downes *et al.* [39] present an improved solution named Restricted FoV WideArea Geolocalization (ReWAG), introducing a near-real time system capable of localising a

limited FoV image on a city wide scale. To encode the images, Downes *et al.* implement the VGG16 paired with a Spatial-Aware Feature Aggregation (SAFA) module developed by Shi *et al.* [122] along a particle filter. Near-real time performance is achieved by implementing a Data Base of precomputed embeddings similar to the technique presented by Bianchi *et al.* [15] and Avila *et al.* [9], this time ordering the embeddings by coordinates in a 2D KD-tree. The system is capable of converging to the correct location after traversing a 15 km trajectory in the city of Boston.

Exploring robot collaboration, the group of Vijay Kumar and Miller *et al.* [94] integrate this time multiple real platforms exploring an environment at the same time. The impressive system integrates multiple localisation algorithms to perform a search and rescue task to find survivors in car clusters. The system uses two UGVs and one UAV to explore an environment at the same time. First, the UAV is deployed to generate a semantic ortho map, using visual odometry to track its position over time. The position drift is then corrected with GNSS measurements. The UGVs use the semantic map to plan trajectories and search areas with car clusters without the need of GNSS data. The three robots share their navigation information over a decentralized WiFi network to avoid redundant plans. This way the area is searched more efficiently and goals are searched by different UGVs. The robots opportunistically synchronize their understanding of the explored areas. At the same time, the UAV continues the environment exploration while updating the semantic ortho map with the UGV's position when connection is available. New goals are set for the UGVs to explore. The novel system provides mechanisms for localising heterogeneous mobile platforms on a common map in real-time. Four experiments were performed, 22 goals out of the 26 selected goals were visited successfully in an autonomous workflow. In addition to the experiments in real environments, 96 experiments were performed in a simulation environment, 10 UGVs collaborate to explore the map while a single UAV explores the area. The presented system sets an important milestone for heterogeneous platforms collaborating together.

Recent papers in the area have shown the potential of approaching the challenge of collaboration between heterogeneous systems [95]. However, in order to achieve collaboration in between heterogeneous robots, the use of LiDAR was required to build a local repre-

sensation of the environment. These novel systems represent the state of the art in robot collaboration and show that collaborative robotics is not achieved only with optical sensors.

2.7 Discussion

The state-of-the-art methodologies aforementioned demonstrate how platforms have the potential to localise themselves based on the optical sensor. Despite the state-of-the-art discoveries in IBL, multiple challenges still need to be addressed to develop real-time platforms that only rely on the optical sensor navigation.

The evolution of the algorithms and architectures shows the direction of IBL in robotics and also demonstrates the effort that the research community is investing to address this problem. The image presented in Fig. 2.8 shows the taxonomy of the most relevant architectures and algorithms in IBL. The tree allows tracking the origins of different algorithms, which helps reveal essential techniques.

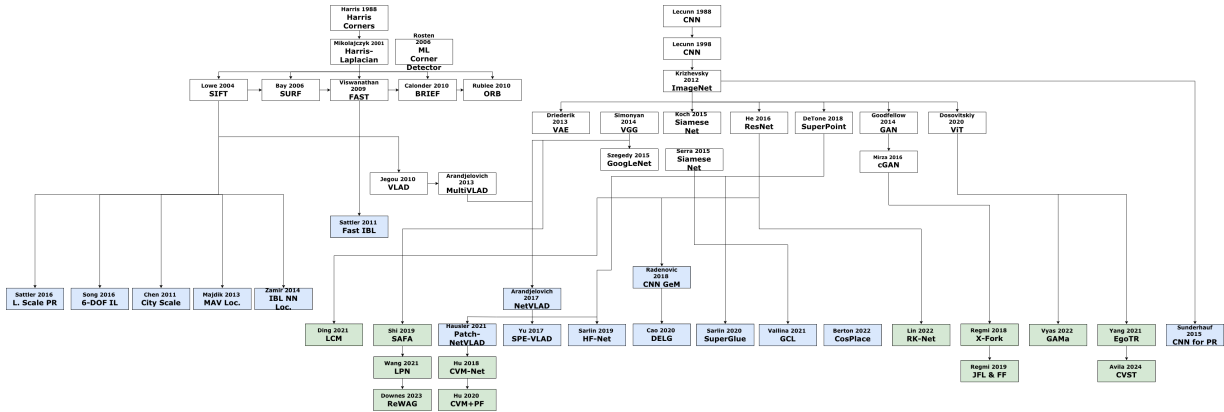


Figure 2.8: A tree that maps IBL research by methodology, showing the evolution of the different algorithms.

One of the main challenges in IBL is retrieval speed. Not all the methodologies presented can be implemented in real time applications due to the computational requirements of deep neural networks. Besides, the ones that can operate in real time, require a dense sampling approach that limits the localisation spatial extent such as [55, 15]. In addition, few real systems have been deployed relying on IBL technologies [15, 94]. Computational efficiency and precision are a big challenge to deploy platforms that only rely on IBL to operate.

In addition, the collaborative systems that have been developed rely on multi sensor systems that increase the cost, complexity and computational cost of mobile platforms [94]. Leaving a gap for platforms that rely on IBL to reduce weight, cost, and integration complexity.

To overcome these challenges, new network architectures that leverage the triplet margin loss [142] and cosine similarity [16] distance are proposed to generate near-real time localisation likelihood. However, the research on the impact of loss functions is still vague in the field of VL. It can be said that the three main loss functions that have shown improvement in training and performance are the triplet margin loss [67], the weighted soft margin loss [54] and the Large Margin Cosine Loss [16].

The experimentation on multiple Siamese and triplet network architectures in conjunction with different loss functions presented by Vo *et al.* [142] demonstrated the importance of comparing multiple architecture structures. Aiming to reduce the gap between viewpoints, the use of cGANs in cross-view localisation demonstrates the need for image transformations in cross-view VL to improve accuracy [105] [106]. This led to present the importance of polar transformation in the reduction of viewpoint differences [122] [124] [123]. However, it is shown that the deep learning methodologies are noisy and present aliasing problems when matching features, hence the implementation of particle filters allows removing aliasing, improving localisation accuracy [67] [55]. Recent research demonstrates that inducing a type of attention mechanism through square ring partitions [146] or transformer networks [143], [155] increases localisation performance. However, their performance under real time scenarios has not been tested.

There is still a discussion over the performance of NetVLAD and other types of pooling methodologies [5] [16], as they seem to perform better under different types of conditions. The performance of the methodologies shown by Alibey *et al.* [5] shows that recall@1 metrics for the Nordland data set are very poor compared to the tests performed for urban scenarios.

Additionally, an important challenge in the IBL field is the fair comparison of methodologies, as each method uses different backbones, loss functions, and data set iterations. The

comparison between methods is becoming more standardised for both Ground and Cross-View IBL [16, 5]. This thesis implements, trains, and tests a large number of state-of-the-art architectures using a fair methodology to assess the performance of each neural network. The results are presented in an extensive ablation study presented in Chapter 3 that targets this particular challenge identified.

Seasonal and light illumination changes in visual appearance have been explored in ground view localisation, as shown by [150]. However, this gap has not been addressed in the cross-view localisation and aerial literature; Luo *et al.* [87] has attempted to evaluate deep learning under seasonal variation similar to Seymour *et al.* [120], recently Deuser *et al.* introduced a methodology robust to temporal changes [34] and also present research exploring seasonal variations for aerial platforms using synthetic data [35]. One of the factors contributing to the lack of cross-view seasonal localisation is the absence of a dataset that matches seasonal data across both aerial and ground-level images. In this thesis, aerial IBL is tested in both urban and rural scenarios, with cross-time variations addressed in urban scenarios and lighting variations tested in rural environments. A collection of real-world data in rural scenes is used to evaluate the recall performance of neural networks for aerial IBL. Demonstrating the operation of IBL for a UAV in near-real time despite the lighting variations.

One of the central challenges identified in IBL is mitigating perceptual aliasing caused by sparse visual features and visual structural ambiguity. Perceptual aliasing can be caused by challenging environments composed of scenes that are difficult to differentiate due to similarities in visual structure. Another cause is feature sparsity and noise induced when IBL is applied to cross-view cases.

The most commonly used method in research to mitigate perceptual aliasing is through the implementation of neural networks capable of attaining high recall metrics [152, 142, 146, 122, 124, 123, 54, 55, 155, 164]. Recent research has implemented particle filters to address this particular issue [67, 55, 38, 39]. This is one of the specific challenges that we address in this thesis, specifically in Chapters 3 and 5.

2.8 Conclusion

This chapter has presented the bases of image-based localisation by providing an overview of the field. Data sets that have been developed in past years, focused on image retrieval tasks, have been analysed and presented. Feature extraction methodologies applied in computer vision for image explainability have been discussed and briefly explained. In addition, methodologies used to improve the efficiency and accuracy of image matching frameworks through clustering and classification algorithms have been introduced as the basis for the novel methodologies that employ deep learning as a means for feature extraction. Leveraging large amounts of data, image features can now be learned from image collections as has been shown by the research presented in the last decade.

IBL has been defined as the task of retrieving the most likely location given a query image. In the previous chapter, different types of IBL were classified according to the viewpoint from which the query and collection images are captured. IBL has grown significantly in recent years thanks to advances in deep learning.

The chapter presents a classification of the IBL task by splitting the challenge based on the viewpoint at which the query image is taken and the viewpoint of the pre-geotagged image collection. The IBL problem is divided into Ground View IBL, Aerial IBL, and cross-view IBL because the different viewpoints require different methodologies to solve them. Multiple methods have found that deep learning is the current state-of-the-art approach for extracting image features.

It can be seen how the research community has migrated from hand crafted features to learned methodologies that have been proven to be superior in efficiency, accuracy and more robust to viewpoint changes. Ground View Localisation has mostly pioneered the methodologies and deep learning architectures that were later adopted and modified by Aerial and Cross View VL. Ground View VL evolved to methodologies that only extract image features through architecture backbones such as VGG16 and ResNet50. The frameworks have settled over the years on a methodology that extracts a feature map with the aforementioned architectures and then performs some classification or attention

mechanism on the feature maps in order to improve the image representation. Methodologies such as NetVLAD, CosPlaces, Feature Mixing and HF-Net have demonstrated good performance in extracting image representations that can later be used by an image retrieval process.

Despite the many similarities in the algorithms presented for Cross View IBL, a more complex framework is required to address the drastic difference in viewpoints. A workflow composed of an image transformation, followed by a feature extraction architecture, has demonstrated to improve accuracy [124]. Once the feature maps have been extracted, an attention module or slicing tasks [146] have been proposed to match the ground level features with the overhead view features. The novel implementations have improved accuracy by mitigating viewpoint differences that cause feature mismatches. Many other deep learning approaches have been explored to reduce the challenges produced by viewpoint differences, such as the attempt of Regmi *et al.* [105] [106] to employ cGANs to generate synthetic aerial views that are compared against the satellite image collection. Although this methodology is capable of performing cross-view image localisation, the complexity of the architecture and training is high. Requiring to train multiple sub modules that operate independently with auxiliary loss functions requires the tuning of multiple hyper parameters. Additionally, increasing the number of modules increases inference time, going against the desire of real-time applications.

Aerial IBL adopts methodologies from both ground view and cross-view IBL frameworks. Due to the nature of aerial applications, two different viewpoints can be present on an aerial platform, as explained in the previous sections. When the approach localises overhead images against satellite images, the methodologies used by ground-view IBL are valid, as long as there is no scale variance between the query images and the satellite image collections due to altitude and resolution. When localising oblique-looking-down query images, cross-view methodologies are best suited.

Regardless of research advancements, multiple challenges remain to be addressed in each type of IBL framework. Ground view VL still has not settled on a methodology that is suitable for real time applications, with suitable memory costs which allow the algorithms

to run on a mobile device and with high recall performance above that which allows robust localisation while avoiding aliasing. Robustness in seasonal variations still needs to be further explored, as most of the research has focused on urban environments under a single season, and the ones that have explored seasonal variations have not performed as optimally as the new methodologies.

Cross-view geo-localisation methodologies do not achieve the same recall metrics as those proposed for ground-view VL due to the nature of the task, which struggles with the drastic viewpoint changes between the query image and the dataset. Novel methodologies for ground-view IBL have applied heuristic approaches that have improved performance [5], but they have not yet been tested in a cross-view scenario. To conclude, not all the methodologies that perform with high recal@1 have been tested for real time tasks, and the ones that have been tested still require improvements in robustness [67, 55, 39]. Future work in the area of IBL will improve the efficiency of feature extractors and implement new training methodologies that enable small, efficient networks to achieve the same accuracy as neural networks with a large number of parameters.

Chapter 3

Cross-View Image Based Localization with Transformer Network

3.1 Introduction

In this Chapter we present a novel deep learning architecture for cross-view localisation. Cross-view localisation (CVL) has received considerable attention in recent years due to its potential applications in areas such as automotive perception and field robotics. Despite the significant advances reported in the literature, a shortcoming of the research is the high reliance on panoramic images to improve recall performance. Capturing panoramic images impacts efficiency and platform cost, creating a motivation to explore this technology on platforms with standard pinhole cameras. In this chapter, we focus on the CVL task where ground level images are limited to a 90° field-of-view (FoV), motivated by the large amount of platforms that operate with standard pinhole cameras. We demonstrate the performance of our approach by presenting a comprehensive ablation study comparing various types of architectures. Extensive testing on multiple network architectures is performed to quantify the impact of the loss function and the effect of weight sharing in Siamese Networks for CVL tasks. Additionally, we introduce the Cross-View Shared Transformer (CVST) network, a novel Siamese Network architecture employing shared weights capable of extracting features from different viewpoints without the aid of polar transformation or other projection [122]. CVST demonstrates state of the art performance

on KITTI [43] a real-world image retrieval benchmark using limited field of view imagery, while reducing the complexity of the Siamese Network. Not only that, but the network shows a competitive score in the CVUSA dataset benchmark.

Cross-view localisation (CVL), a branch of Image based localisation (IBL), *a.k.a.* Visual Place Recognition (VPR), has received significant attention in recent years due to its potential applications in areas such as robotics and autonomous vehicles. As is the trend across computer vision, the past decade has seen the leading paradigms to VPR migrating from classic feature extraction methodologies [87, 12, 27, 139, 114, 156] to deep learning based approaches [69, 142, 67, 54, 55]. Cross-view localisation (CVL) refers to the task of estimating the location of a ground level query image by calculating its similarity relative to a georeferenced database of aerial images, where the coordinates of the aerial image with the highest similarity is assumed to be the capture location of the query image. It is common to treat CVL as a standard image retrieval task, where the challenge is to train a deep neural network to map the scene’s visual appearance from two extreme viewpoints to a common embedding space where they can be compared. Fig. 3.1 depicts an example of the viewpoint used to capture a ground level image.



Figure 3.1: Viewpoint of a ground-level image captured in Toronto.

Recent CVL research has tested multiple feature extraction techniques such as CNN based methods [12, 25, 152, 81, 142, 67, 158, 82, 124, 123, 79], feature pooling methodologies [54,

55, 122, 146], feature aggregation techniques [82, 106], and different attention mechanisms [155, 146, 164, 168]. Although these approaches have demonstrated compelling performance in scenarios computed with panoramic images, only a small number of studies present results on limited (i.e. $\leq 90^\circ$) FoV scenes [142, 67, 168], where those that have showed substantially degraded performance. Techniques reliant on panoramic images add extra complexity and cost to mobile platforms [55], thereby making them unsuitable for more general scenarios and use-cases. In this chapter, we specifically address the problem of cross-view localisation under such limited FoV conditions, providing a comprehensive set of experiments measuring and comparing the performance of a number of state-of-the-art approaches. In line with the previous work of Kim et al. [67], we use the KITTI [43] autonomous driving dataset in conjunction with corresponding satellite imagery as a challenging real-world benchmark dataset. Additionally, we also include results on the panoramic CVUSA [158] dataset to quantify the variation in performance between panoramic and limited field of view setups.

In our analysis, we measure the impact of sharing vs. non-sharing of weights in Siamese branches across a number of different network architectures for ground level imagery limited to a 90° FoV from the KITTI data set and panoramic images found in CVUSA. Based on our analysis, we propose a Cross-View Shared Transformer (CVST) network consisting of a shared branch network composed of a Vision Transformer (ViT) [37] network followed by an MLP-Mixer Head [136]. The MLP-Mixer Head encodes the output of the transformer block and maps the features to a common embedding space [136]. CVST demonstrates competitive performance against state-of-the-art approaches while reducing the complexity of the Siamese Network. In particular, sharing the weights between branches in the Siamese Network substantially reduces the number of network parameters. This in turn reduces the computational complexity when calculating the networks gradients, and as a consequence, the associated training time.

In summary, the contributions of Chapter 3 are: (i) A comprehensive comparative evaluation of state-of-the-art Cross-View Localisation architectures on both a challenging limited (i.e. $\leq 90^\circ$) FoV cross-view dataset, using the original KITTI dataset, and on the widely used panoramic CVUSA dataset; (ii) An ablation study measuring the impact of different

feature extraction architectures implemented with shared vs. non-shared weights across the evaluated Siamese configurations, showing that in the majority of cases sharing of weights and the use of a weighted soft-margin loss improves performance; (iii) A novel Cross-View Shared Transformer (CVST) network that shares weights and improves the traditional MLP with an MLP-Mixer Head, achieving state of the art performance on limited field of view ground level imagery.

The work presented in this chapter is based on the peer-reviewed publication [10]. Eduardo Andres Avila Herrera, the author of this thesis, is the lead author of this work and was responsible for the conception and design of the method, its software implementation, the design and execution of the experiments, the analysis of the results, and the writing of the manuscript. The co-authors, Tim McCarthy and John McDonald, contributed in a supervisory capacity, providing guidance on the research and feedback on the manuscript.

The remainder of the Chapter is organised as follows. In Section 3.2, we provide a summary of the state of the art in CVL. In Section 3.3, we describe the methodologies applied in detail and the proposed CVST architecture. Section 3.4, presents our experimental results and compares our approach with state-of-the-art methods. Finally, in Section 3.5, we conclude the chapter and discuss future research directions.

3.2 State of the Art

To geolocate a query ground level image, it is common to formulate the task as an image retrieval problem. Visual place recognition started geolocating ground level images by comparing the query image to a collection of ground level images [50, 87, 12, 80, 156]. Workman *et al.* [152] migrated from classic feature extraction methodologies to deep learning techniques employing convolutional neural network (CNN) architectures.

CNN Methodologies: An extensive series of approaches have been evaluated using different CNN architectures. Lin *et al.* [81] demonstrate that deep learning techniques allow learning a common latent space where images from different view points can be compared using Siamese Networks. Castaldo *et al.* [25] attempt to solve the cross-view

localisation task by incorporating semantic segmentation. The segmented query image is projected to an aerial view that is compared to an overhead map with the same semantic representations. Workman *et al.* [152] test CVL over a large scale, introducing the CVUSA dataset to the area, which collects panoramic images from across the USA. Vo *et al.* [142] present a comprehensive study of the impact of different loss functions comparing Siamese and triplet network configurations implemented with an AlexNet backbone. The tested networks do not use shared weights between branches of different views. A novel dataset of limited FoV ground level images with their corresponding satellite image matches is used for testing. Additionally, they present DBL-Net, a triplet network architecture using the novel Distance Based Logistic (DBL) Loss which they compare against six different variants.

Kim *et al.* [67] demonstrate the use of a particle filter to compensate the aliasing caused by noisy estimates output by a Siamese network. Experimental results demonstrate localisation of a ground level vehicle through a modified VGG16 (ModVGG16) architecture implemented in a Siamese configuration, where ground and satellite images are encoded using a 4096 dimensional embedding space. The similarity between images is calculated using the Euclidean distance between the embeddings and is later used by the particle filter to compute importance weights for each sample. Zhai *et al.* [158] utilised semantic segmentation for both ground and aerial views to extract the important features. To reduce the variations due to changes in viewpoint, a learned transformation matrix is used to transform the aerial semantic maps into the ground view plane. Zhu *et al.* [167] present GeoNet a novel Siamese architecture, where each branch integrates a ResNetX backbone followed by capsule layers which they refer to as GeoCaps layers. The GeoCaps encodes spatial hierarchy information and learn viewpoint invariant representations. However, the model is limited in its ability to generalise to novel scenes that are not seen during training.

Feature Pooling: A number of papers have been presented that investigate pooling methodologies for generating global feature representations. Hu *et al.* [54, 55] experiment with CVM-Net, a VGG16 backbone architecture followed by a NetVLAD pooling layer to generate global feature representations and reduce viewpoint variations. The networks are trained with a Weighted Soft Margin loss that uses a dataset of triplets composed of an

anchor ground panoramic image paired to a positive and negative satellite match.

Synthetic Feature Aggregation: Regmi *et al.* [105] develop a GAN architecture capable of performing one shot segmentation and generating a synthetic aerial image from the ground view image. The novel generator is the used by Regmi *et al.* [106] for cross-view localisation, where the generator is used to measure a similarity metric between a synthetic and a proposed aerial view. Liu *et al.* [82] introduce the concept of colour coded orientation maps to CVL. Augmenting the RGB channels of both ground and aerial views with the orientation maps within the training dataset improves the network’s robustness to orientation variations. Additionally, the CNN network is drastically reduced to only 7 convolution layers.

Attention Mechanism: Shi *et al.* [122] introduced a Self Attention Feature Aggregation (SAFA) block that emulates self attention to rank feature importance. The module results in a weighted feature map that is computationally efficient and can be trained in an end-to-end process. Applying the attention mechanism to the local features extracted from the CNN backbone improves performance over the state-of-the-art methodologies. Shi *et al.* [124] investigate how to align feature maps from different viewpoints with a novel Feature Transport Module. Designing a cost generation subnetwork, the feature transport problem is formulated and optimised using a Sinkhorn solver to find a feature transport matrix. Shi *et al.* [123] introduce Dynamic Similarity Matching (DSM), a methodology that determines the orientation of a query image by finding a correlation between polar-transformed aerial images and the query image.

Zhu *et al.* [166] introduce a novel CVL dataset, VIGOR, where panoramic ground level query images are no longer aligned in the centre with the corresponding overhead view. Using global negative mining [165] further experiments on SAFA [122] are presented. Wang *et al.* [146] investigate a novel attention methodology called Square Ring Partition (SRP) that takes advantage of contextual information, attempting to reduce the view point difference between ground and overhead images. Lin *et al.* [79] introduce a novel attention module named USAM which is capable of extracting local key points used as a residual attention mechanism.

Transformer Networks: Yang *et al.* [155] present work applying transformer architectures to the CVL task. L2LTR [155] is a hybrid architecture composed of a ResNet50 backbone and a modified Vision Transformer Network, where a self-cross attention mechanism allows interactions between features in adjacent layers of the network.

Zhu *et al.* [164] implement TransGeo, a two stage transformer architecture that reduces inference time and improves image retrieval performance. The first stage compares a ground level panoramic image against a satellite image using a ViT architecture. The second stage zooms into the satellite image performing a non uniform cropping based on the attention maps calculated from the first stage. The attention maps are used as a binary mask to crop the overhead view. In the second stage, the resolution of the aerial image is increased and the attention calculated from the first stage is reused to generate a new embedding. Zhu *et al.* [168] present SAIG which leverages the feature extraction of a six layer convolutional stem neural network to compute the image patches in a vision transformer network. The idea is adopted from Xiao *et al.* [153]. Once the patches are obtained, learned positional embeddings are added, followed by a modified transformer network. The ViT transformer is further modified by removing the feed forward network (FFN) to improve computational efficiency at the cost of performance. To encode the output of the transformer modified network, a final feature mixing block is used to generate the feature embeddings. Deuser *et al.* [33] explore the recent ConvNext [84] architecture accompanied by an InfoNCE [101] loss function, achieving state-of-the-art results on Cross-View Localisation employing panoramic images.

A notable feature of the majority of CVL research to date is the use of panoramic ground level images in the design and evaluation of the proposed frameworks, with only a small number of approaches employing limited Field of View (FoV) ground level images [81, 142, 67, 168]. As demonstrated by [168], current state of the art approaches suffer significant deterioration in performance when applied in limited field of view conditions. We argue that the requirement for panoramic sensor input limits the applicability of these techniques. On the contrary, extending their applicability to the more predominant limited FoV sensors represents an important and yet under-addressed issue in the field of CVL. In this work, we specifically address this issue both through a comparative analysis of existing

techniques on ground level images limited to a 90° FoV, and, the development of a novel network demonstrating state of the art performance in this setting.

3.3 Methodology

In this section, we first present the methodology employed to collect the dataset utilised within our experimental evaluation. Next a summary of each of the existing architectures evaluated in our experiments is provided. This is followed by a detailed description of our proposed CVST network. Finally, we review each of the loss functions used for training the networks used within our evaluation.

The cross-view localisation problem is addressed as an image retrieval task, where a query image is searched in a collection of georeferenced images and the closest match determines the location of the query image. The query image in the cross-view localisation task is a horizontal view captured from a moving vehicle, and it is compared against a catalogue of satellite images. To compare the images we employ a deep learning approach where we propose a Cross-View Shared Transformer Network as the feature extraction method. The features extracted from the neural network allow for comparison between the different viewpoints in a learned latent space. The latent space is learned through training the network on image triplets consisting of an anchor image, a positive and a negative match, where the positive and negative matches are satellite images and the anchor image is the query image or horizontal view. A metric of acceptance or rejection is learned via the soft margin triplet loss. Further details on the neural network training are presented in Section 3.3.4, where we approach the network training as a metric learning problem.

The Neural Network can be seen as a function $f(I_n)$ that receives an image as an input and returns a feature vector in a latent space that can compare an aerial viewpoint with a horizontal viewpoint. We use the network as a comparative function that estimates how similar the current scene is to the compared satellite image. To implement the localisation system, two main components are required. i) A cross-view network and ii) a collection of satellite images captured from the area where the vehicle will be operating. The entire

collection of images is processed by the neural network and allows to create a database of georeferenced feature vectors. This database is the map used to localise the query image.

When a vehicle with unknown position wants to find its location, the current horizontal view of the vehicle also referred to as the query image, is processed by the neural network returning a feature vector. This vector representation of the current scene is compared against the database, where the position of the vehicle is assumed to be the position of the matched image. To evaluate the performance of the neural network a retrieval metric is employed.

3.3.1 Dataset



Figure 3.2: Satellite image of Europe showing the location of Karlsruhe, Germany. Location where the dataset is collected.

The KITTI dataset presented by Geiger *et al.* [43] is a multi sensor vehicle dataset collected in the city of Karlsruhe, Germany. Following the methodology of Kim *et al.* [67], satellite images corresponding to the ground level images are collected to generate a cross-view dataset. Karlsruhe is a medium size city located in mainland Germany as shown in 3.2.

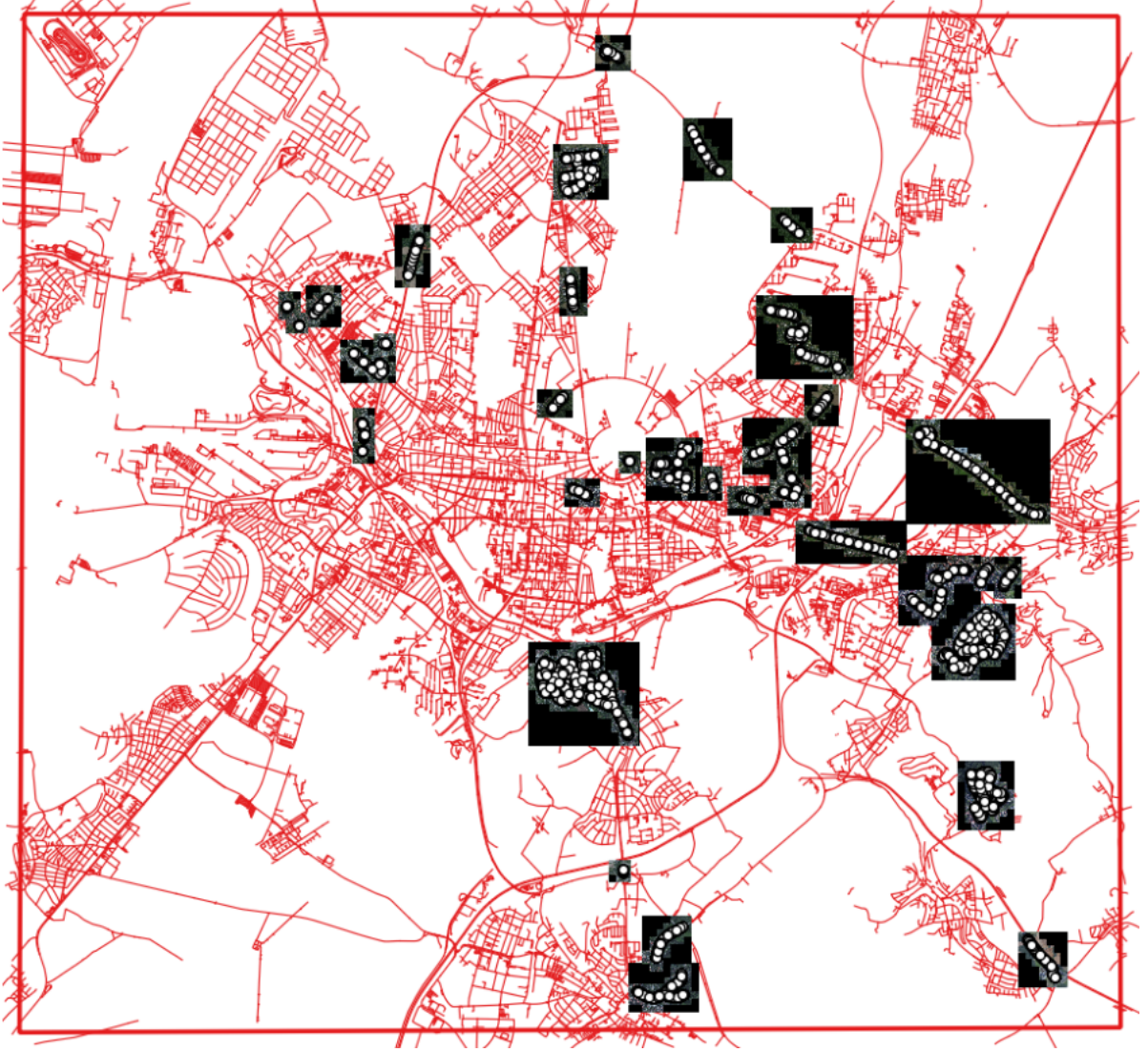


Figure 3.3: Satellite images corresponding to the ground-level images collected in the KITTI dataset.

To extract the satellite images corresponding to the locations of the collected scenes in the KITTI dataset we use the GPS coordinates collected, as shown in Figure 3.3. Each ground level image is paired with its corresponding Google satellite image captured 5 metres in front of the vehicle geolocation. In line with Kim [67], the satellite image covers an area of 71 metres in front of the vehicle and a width of 50 metres. Each ground level image covers a FoV of 90° according to the KITTI dataset documentation. Both satellite and

ground level images are resized to $224\text{px} \times 224\text{px}$. For training, we randomly generate 94,121 triples.



Figure 3.4: Road network of Karlsruhe, Germany.

To increase the number of negative pairs, we generate 12,162 random satellite views located in road network of the Karlsruhe region shown in Figure 3.4. The Karlsruhe road network is used to stochastically sample the additional satellite negative pair images as shown in Figure 3.5. The random samples are collected by generating random coordinates normally distributed over the road network as shown in Figure 3.6. For testing, we isolate 5,177 images collected by KITTI [43] on 2011_09_30 drive 0028 from all training work flows. An example of the generation of image triplets is shown in Figure 3.8.

For comparison of performance with panoramic images we utilise the CVUSA dataset

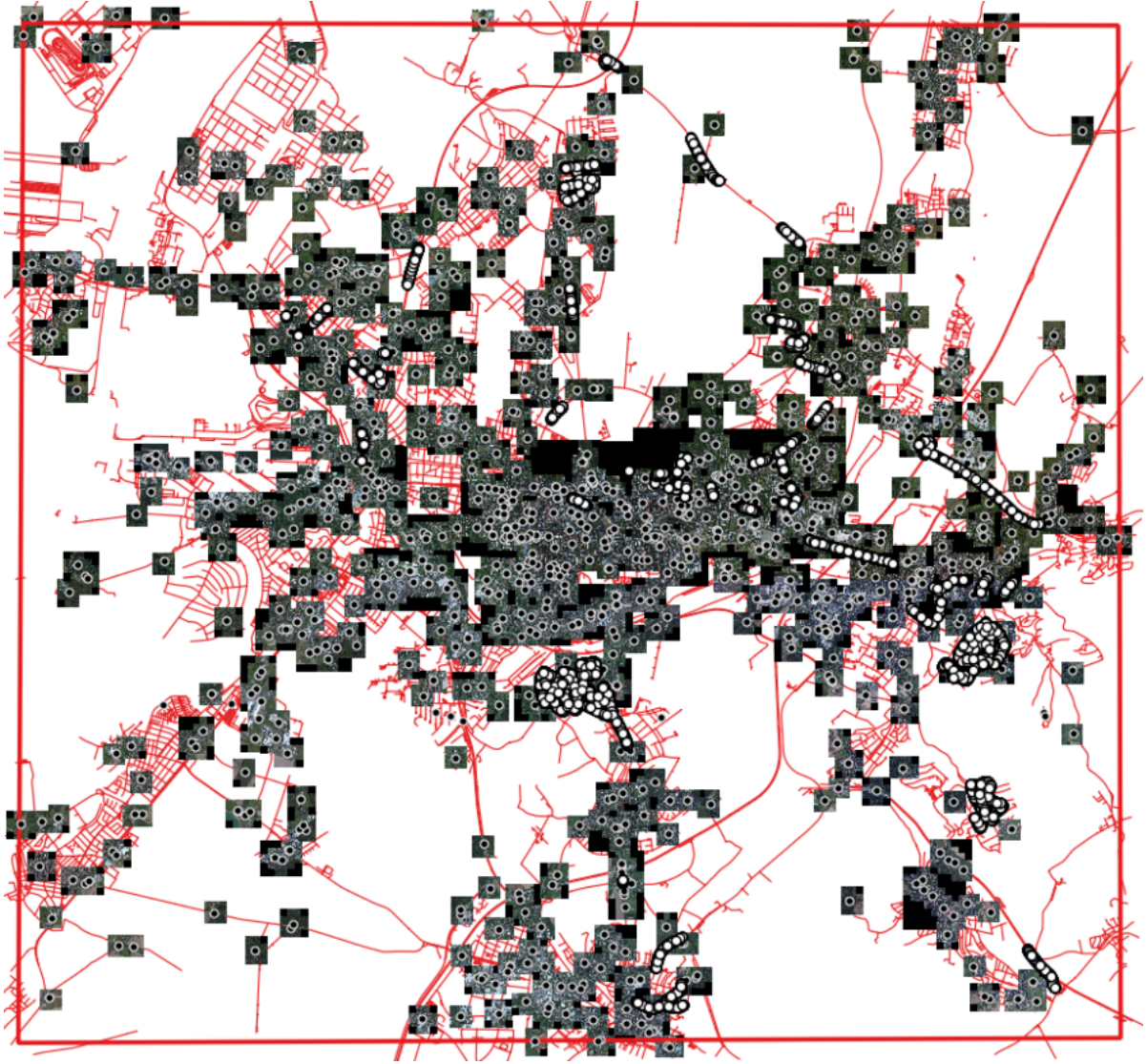


Figure 3.5: Normally distributed samples collected over the Karlsruhe road network to generate negative satellite image pairs. In white satellite image samples from the KITTI Data Set locations. In black stochastic samples of satellite images.

consisting of a collection of 35,533 training panoramic images each of which is paired with its corresponding satellite image. The dataset includes a collection of 8,884 testing images that are only used during testing of the architectures. The sampling density of CVUSA compared to the KITTI dataset is lower given that CVUSA covers multiple cities. We note that to facilitate comparison of shared and non-shared weight configurations, both satellite and panoramic images are resized to $224\text{px} \times 224\text{px}$ for training and testing of the architectures.

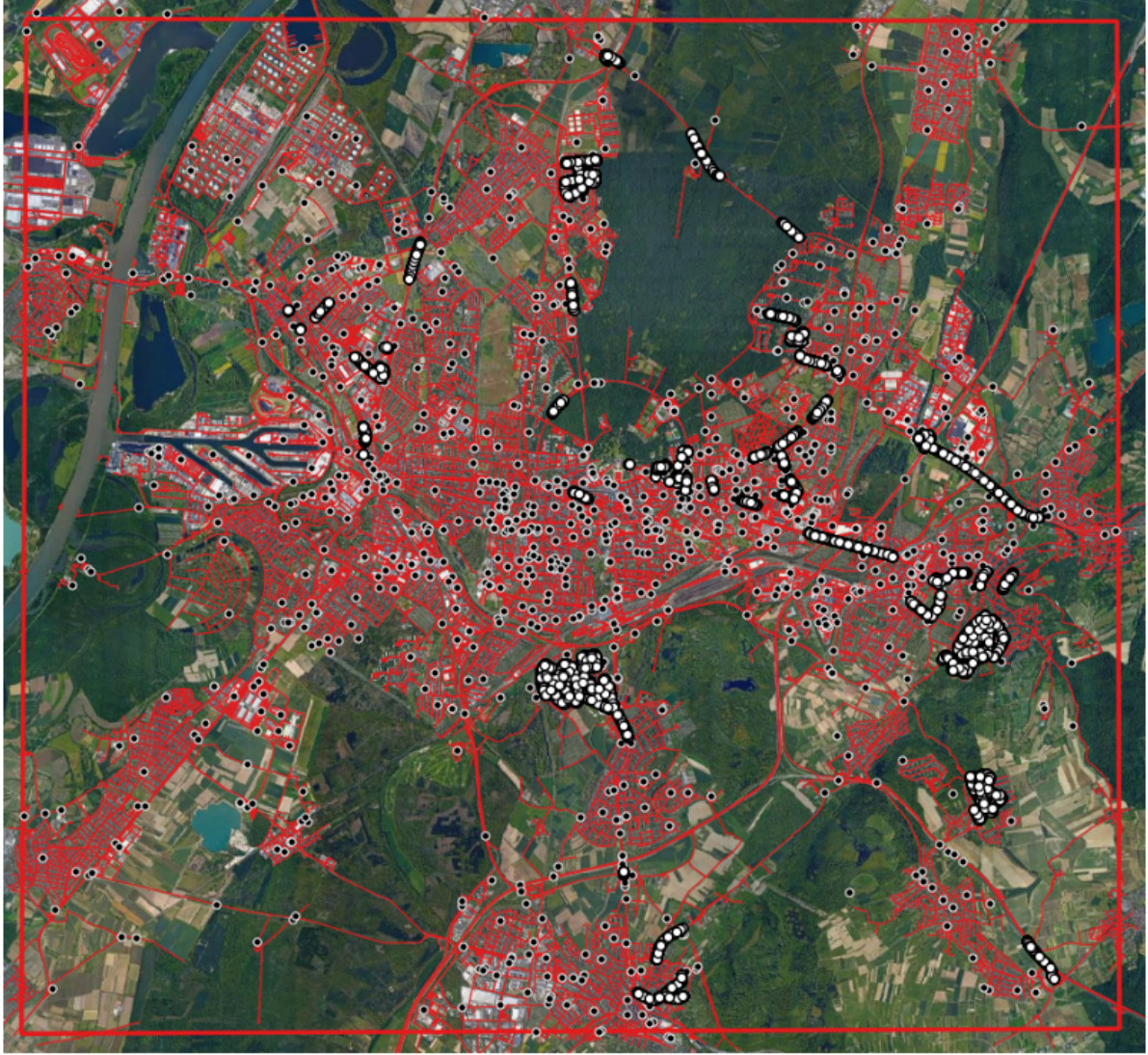


Figure 3.6: The white markers show the collection locations of the KITTI data set. The black markers show normally distributed samples collected over the Karlsruhe road network to generate negative satellite image pairs.

3.3.2 Network Architectures

This section of the chapter explains the implementation details of each architectures used for experimentation and comparison.

Multiple architectures are tested to identify the characteristics that improve performance for CVL. For fair comparison between methodologies, the same backbone is maintained across each technique. VGG16 is chosen to be the feature extraction backbone for the CNN architectures, following Hu *et al.* [55]. Each network is trained from scratch unless pre-trained weights are specified.

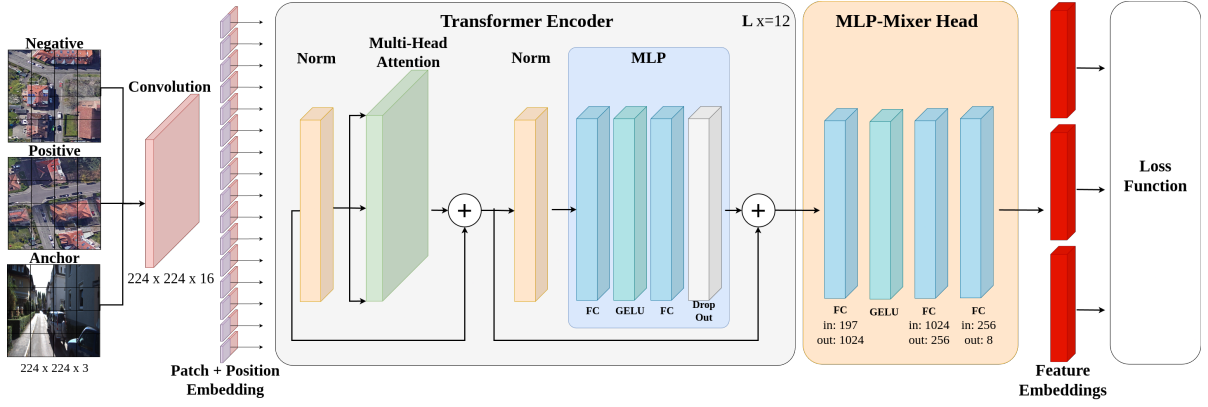


Figure 3.7: The CVST Architecture employs a Siamese Network with identical branches represented in the image by a single branch network. Sec. 3.3.3 provides a detailed explanation of the architecture.

ModVGG16: Proposed by Kim *et al.* [67] this network uses a VGG16 backbone modified with a skip connection from the first convolution layer of the fourth convolution block. The skip connection is computed through an average pooling layer, and added to the last max pooling layer of the network. A kernel size of 4 and stride of 4 are used for the average pooling layer in the skip connection. In contrast to the Siamese Network presented by Kim *et al.* [67], the weights are shared between branches. The modified VGG16 network, referred to as ModVGG16, is used to build a triplet network trained employing both a triplet margin loss and a weighted soft margin loss in (Eq. 5.9). We evaluate the network trained both from scratch and with pre-initialised ImageNet-1K weights.

NetVLAD: The NetVLAD pooling layer [7] is tested using a non modified VGG16 backbone that is only trained on the last convolution block. The rest of the layers use ImageNet-1K pretrained weights that are frozen before training starts. The NetVLAD network is trained with 64 cluster centres and initialised on a subset of the training set by applying a KNN clustering. In the case where weights are not shared, each branch uses its corresponding viewpoint data for initialisation consisting of a subset of ground level images for the ground branch and a subset of aerial images for the satellite encoder.

CVM-Net I: As shown by Hu *et al.* [54, 55] CVM-Net I uses a VGG16 backbone pretrained on ImageNet-1K. Dropout layers are added to the VGG16 network in the last convolution block. NetVLAD is used as the pooling layer at the end to reduce differences in between viewpoints. The NetVLAD block is initialised randomly for training. The last

block of convolutions is modified by adding dropout layers with $p = 0.8$.

SAFA: Tests using Spatially Aware Feature Aggregation (SAFA) [122] are performed without polar transformation to compare the methodologies under the same training scenario. Additionally, the fully connected block present in VGG16 [67] is replaced with the SAFA block to test its performance. Both networks are trained from scratch and with ImageNet-1K initialised weights, where the SAFA module is initialised randomly. It is noted that the use of the SAFA block results in a significant reduction in the number of training parameters.

L2LT: Layer to Layer Transformer (L2LT) is a hybrid architecture composed of a ResNet50 block followed by a ViT network [155] achieving state of the art results in image based localisation. The L2LT network is tested on both the CVUSA and KITTI datasets. The architecture is tested as a hybrid network including the ResNet-50 network (L2LT Hybrid) and as a stand alone modified ViT without the ResNet-50 backbone (L2LT) to measure its impact on performance.

ConvNext: ConvNext is a novel convolutional neural network implemented with a series of improvements that show competitive results on image classification tasks [84]. A ConvNext-Base is used for our tests where shared weights and non-shared weights networks are tested. The ConvNext architecture is implemented in our tests with pre-trained ImageNet-1K and computes a 1000 dimensional embedding from the input images.

3.3.3 Cross-View Shared Transformer (CVST)

Following on from the results of [37, 155, 168], we propose a network architecture that employs a combination of the Vision Transformer (ViT) [37] network with a MLP-Mixer Head [136], maintaining the original skip connections and MLP of the ViT. The ViT Network is implemented with 12 heads and 12 blocks. We set up the network in a shared weight Siamese configuration, which we show both reduces the overall complexity whilst also improving the network’s performance. The original classification block of the ViT is replaced by the Mixer Head MLP block following Tolstikhin’s findings [136]. Each image is processed independently by the transformer encoder, then the Mixer Head maps the

extracted features to the embedding space. Each image is mapped to a corresponding feature embedding separately. A diagram of the CVST architecture is shown in Fig. 3.7.

The initial network block consists of a patch extraction module, where patches are extracted using a single convolution layer with kernel size of 16 and stride shift of 16, resulting in 16 patches, followed by a linear transformation applied to each patch. A positional encoding is added to each linear projection. This is followed by 12 transformer encoder blocks that compute attention and extract features. The final stage is the MLP-Mixer Head block that compresses the features and maps them to a common latent space.

The MLP-Mixer is composed of four layers. The first one is a fully connected layer that takes a 197 dimensional linear vector and expands it to a 1024 dimensional linear feature followed by a GeLU activation layer. The third layer is a fully connected layer that reduces the vector to a 256 dimensional linear feature. The final fully connected layer projects each path to an 8 dimensional encoding. Given 768 patches this results in the final feature vector flattened to a 6144 dimensional encoding. CVST is trained as a metric learning problem and not a classification task using the Weighted Soft Margin Loss in Eq. 5.9.

The CVST network consists of two identical branches during inference, where each branch takes an individual image and computes a single 6144 dimensional embedding. The embeddings are mapped to the same latent space where they can be compared for localisation using the cosine similarity metric. When training CVST the network encodes the anchor image, the positive image and the negative image. The network loss is then computed using the three embeddings as explained by weighted soft margin loss equation in Eq. 5.9. Finally, the AdamW optimiser is chosen for training.

Tests are also carried out on a Siamese network variant that does not share weights between branches as a comparison. CVST has a number of important differences when compared to similar transformer based architectures [155, 164, 168]. In particular, the CNN stem presented by [155] requires additional training parameters and adds an extra computational block to the process. In contrast CVST does not employ a CNN stem, thereby reducing further training parameters and inference time. This is supported by [37] where the authors demonstrate similar performance of the ViT transformer without the ResNet50 CNN

backbone. Different from [164] CVST is a single stage transformer architecture with an MLP-Mixer Head. As such, the CVST network learns to map features from ground level and overhead views using a single embedding branch.

3.3.4 Training

Each network undergoes 40 training iterations on the full training-set and is trained, validated, and tested on KITTI to measure performance under limited FoV. To evaluate its performance on panoramic imagery, the best performing networks from the limited FoV tests are trained and tested separately on CVUSA. The datasets are not mixed during training nor testing. We set a learning rate of the optimizer to $lr = 10^{-5}$, as suggested by Hu *et al.* [54]. A mini batch size of 128 triplets is used for training, as proposed by Vo *et al.* [142]. Two different loss functions are used to train VGG16 and ModVGG16 networks to measure their impact on the CVL task. The soft margin loss function presented by Hu *et al.* [55] and the triplet margin loss also known as the hinge loss used by [142] are chosen for testing as they are commonly used in CVL and image recall tasks. All transformer based architectures (L2LT, L2LT hybrid, ConvNext and CVST) are trained using the AdamW optimizer with a weight decay of 0.3, whereas the remainder use the Adam optimizer.

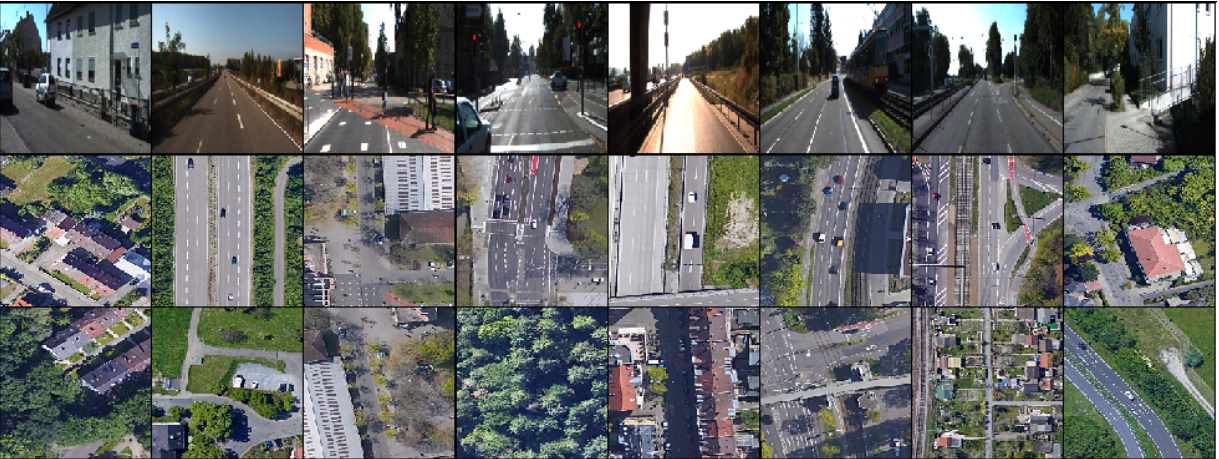


Figure 3.8: Example of mini batch of 8 triplets. The top row shows the ground level image also referred to as the anchor. The middle row of images represent the positive overhead matches. The bottom row of images show negative pairs. Data Attribution: Imagery 2023 CNES / Airbus, GeoBasis-DE/BKG, GeoContent, MaxarTechnologies, Maps Data: 2023 Google

Additionally, the best performing networks on the KITTI dataset are trained over CVUSA.

In order to test the performance of the networks between shared and non-shared weights the ground level panoramic CVUSA images are resized to $224\text{px} \times 224\text{px}$. This ensures a fair comparison between the chosen networks where we find that sharing weights in Siamese networks tends to improve recall metrics. Results are shown in Table 3.1 and Table 3.2.

Triplet Margin Loss: The triplet margin loss, also known as hinge loss, is defined as,

$$\mathcal{L}_{hinge}(f(I_a), f(I_p), f(I_n)) = \max(0, m + \Delta d(f(I_a), f(I_p), f(I_n))) \quad (3.1)$$

where, the loss uses a fixed margin, m , increasing the distance between negative pairs whilst decreasing the distance between positive pairs. $f(\cdot)$ represent the features extracted from one network branch. I_a , I_p and I_n refer to the anchor image, positive image and negative image of each triple, respectively (see Figure 3.8). $d(f(I_i), f(I_j))$ denotes the Euclidean distance between the feature embeddings for images I_i and I_j . Finally, Δd is the difference between positive and negative pair distances.

$$\Delta d(f(I_a), f(I_p), f(I_n)) = d(f(I_a), f(I_p)) - d(f(I_a), f(I_n)) \quad (3.2)$$

In our experiments we set $m = 80$ following [67].

Weighted Soft Margin Loss: The weighted soft margin loss has been extensively tested in the past [54, 55, 155, 124, 166, 164], hence we choose it as a standard loss function for the comparative study between architectures.

$$\mathcal{L}_{weighted} = \ln(1 + e^{-\alpha(\Delta d)}) \quad (3.3)$$

This loss function has the advantage of not requiring a hard margin, allowing the network to discriminate between negative and positive pairs without choosing a hard margin m . The value of alpha $\alpha = 10$ is found to be the best value after comparing recall and training loss results with other values.

3.4 Experimental Setup and Results

In this section we evaluate the performance of each architecture. The results are shown in Table 3.1, 3.2 and 3.3.

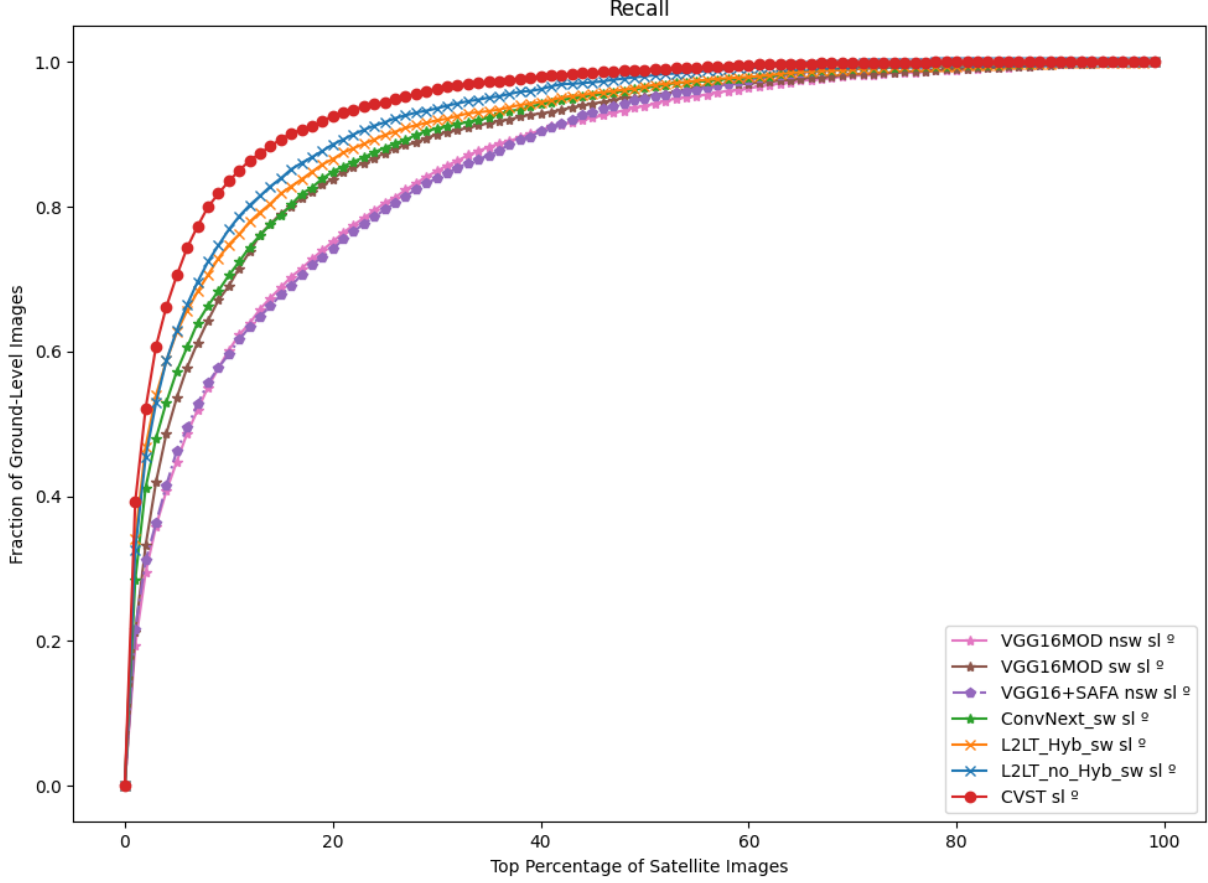


Figure 3.9: Fraction of query ground level images for which the corresponding overhead view is in the top-K%. shared weights (sw), non-shared weights (nsw), Weighted Soft Margin Loss (sl), pretrained ImageNet-1K weights (^o)

Evaluation Metric: To measure the performance of each architecture, we employ the recall at top K%, in line with the metric used in multiple studies on CVL [54, 122, 155, 168]. For a given ground level query image, it is considered successfully localised if its corresponding ground truth aerial image is in the top K% of retrieved images. A measure of the discriminative ability of the networks is shown in Figure 3.9 where we visualise the variation of recall as a function of increasing K .

Geo Localisation Recall: Recall is a measurement used to evaluate how effective a model can identify the positive location match. Recall is calculated as the ratio between

true positive matches and the addition of true positive matches and false negative as shown in Equation 3.4.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \quad (3.4)$$

KITTI Ablation Study			
Network	R@1% (%)	R@5% (%)	R@10% (%)
VGG16 * † ^o	10.95	33.05	49.14
VGG16 * ^o	15.61	40.66	56.69
VGG16 †† ^o	13.83	40.08	57.00
VGG16 † ^o	19.10	49.51	65.44
ModVGG16 * ‡	12.69	35.5	51.11
ModVGG16 *	15.05	38.65	54.10
ModVGG16 †‡	15.34	37.26	53.99
ModVGG16 †	17.77	47.00	66.31
ModVGG16 †† ^o	19.41	44.79	60.15
ModVGG16 † ^o	21.19	53.62	69.04
VGG16 NetVLAD †‡	12.69	32.26	44.56
VGG16 NetVLAD †	13.39	31.87	46.28
CVM-Net I †† ^o	8.85	31.74	48.18
VGG16 + SAFA †‡	8.60	24.76	37.30
VGG16 + SAFA †	13.41	36.95	54.49
VGG16 + SAFA †† ^o	21.67	46.28	59.63
VGG16 + SAFA † ^o	20.32	48.77	64.94
L2LT †† ^o	27.78	55.12	69.25
L2LT † ^o	32.58	62.93	76.83
L2LT Hybrid †† ^o	31.81	63.70	78.54
L2LT Hybrid † ^o	34.17	62.66	74.75
ConvNext † ^o	28.41	57.27	70.50
ConvNext †† ^o	16.86	50.07	67.53
ViT + Mixer Head †† ^o	22.48	49.68	63.86
CVST † ^o	39.2	70.5	83.52

Table 3.1: Results on the 5K test set. † denotes that the model was trained using weighted soft margin loss. ‡ represent Siamese networks that do not share weights between branches. ^o represents networks with weights pretrained on ImageNet-1K. * represents networks trained from scratch.

Ablation Study: We present an ablation study to measure the impact of the loss functions, and the effect of sharing or non-sharing of the weights between Siamese branches. Tables 3.1 and 3.2 present recall performance across the two datasets. Here it can be seen that the skip connection used in [67] in ModVGG16 improves performance relative to the VGG16

CVUSA Ablation Study			
Network	R@1% (%)	R@5% (%)	R@10% (%)
ModVGG16 † ^o	57.35	89.58	96.30
VGG16 + SAFA †‡ ^o	69.75	94.77	98.34
L2LT †‡ ^o	91.36	99.36	99.76
L2LT † ^o	91.75	99.38	99.82
L2LT Hybrid †‡ ^o	93.28	99.36	99.78
L2LT Hybrid † ^o	89.75	99.00	99.64
ConvNext † ^o	84.30	98.28	99.36
ConvNext †‡ ^o	81.38	98.50	99.56
ViT + Mixer Head †‡ ^o	92.99	99.23	99.66
CVST † ^o	95.62	99.68	99.87

Table 3.2: Results on the CVUSA test set. † denotes that the model was trained using soft margin loss. ‡ represent Siamese networks that do not share weights in between branches. ^o represents networks that have been pretrained with ImageNet-1K weights. * represents networks trained from scratch.

network when employing pretrained networks. It is demonstrated that applying a weighted soft margin loss results in a 4% improvement in the VGG16 model and $\sim 6\%$ improvement for the ModVGG16 model. Table 3.1 shows that the best performing network is CVST with a 39.2% Recall @ 1%, a 83.52% Recall @ 10% and a $\sim 5\%$ improvement over L2LT Hybrid architecture. Table 3.2 shows the results on CVUSA. Additionally, it is noted that sharing weights in a Siamese network configuration improves performance by $\sim 5\%$ in the case of VGG16 networks and sharing weights results in a $\sim 16\%$ improvement by CVST over the ViT + Mixer Head based Siamese network. Furthermore, we note that weight sharing can significantly reduce training times.

Inference Time: Table 3.3 compares each network in size and inference time. It is demonstrated that CVST is faster than any of the top performing networks. All networks are tested under the same data and a single NVidia RTX 3090 GPU containing 24 GB on board memory for fair comparison.

Attention Map Visualisation: The attention maps for each block are reshaped and interpolated to match the shape of the original image for visualisation. Figure 3.11 illustrates the generated attention maps of the ninth head, showing the focus on the road regions across the two different viewpoints.

Figure 3.10 portrays several recall examples when querying a ground level image limited

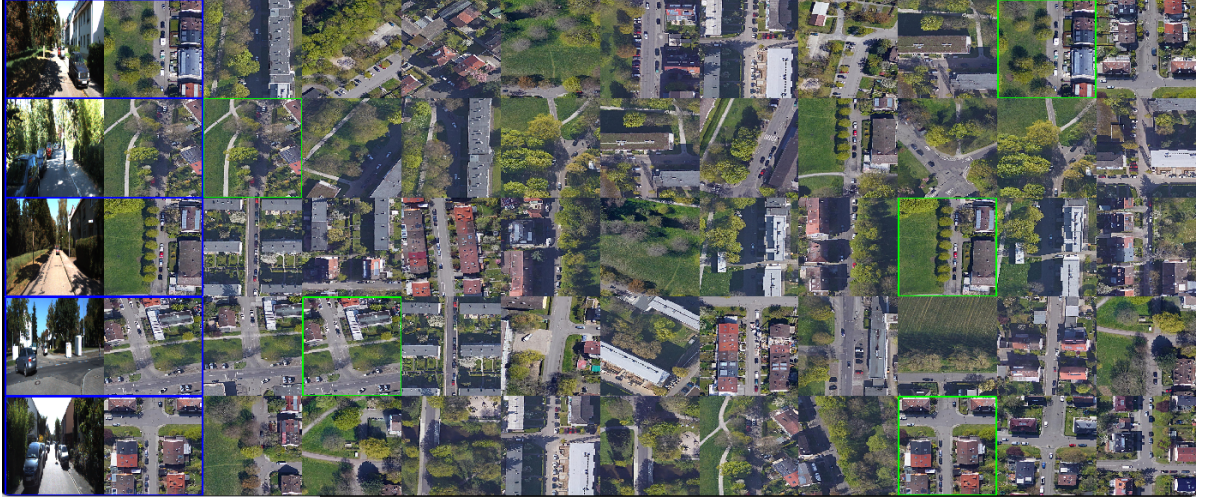


Figure 3.10: Example of query ground level images. Highlighted in blue the ground truth pair and in green the correct match. Data Attribution: Imagery 2023 CNES / Airbus, GeoBasis-DE/BKG, GeoContent, MaxarTechnologies, Maps Data: 2023 Google

to 90° FoV. These samples highlight the challenging nature of the CVL task due to the extreme variation in viewpoints. In particular, it can be seen how the view point differences induce inherent ambiguity in the data resulting in aliasing during the image retrieval process.

Network	Networks Size		
	Parameters (M)	I.T. (ms)	Relative I.T. (%)
VGG16 ‡	268.52	7.33	43.95%
VGG16	134.26	7.33	43.95%
ModVGG16 ‡	268.52	7.35	44.07%
ModVGG16	134.26	7.35	44.07%
VGG16 NetVLAD †‡	29.5	4.78	28.65%
VGG16 NetVLAD †	14.75	4.78	28.65%
CVM-Net I †‡°	29.5	4.60	27.56%
VGG16 + SAFA ‡	29.5	4.58	27.45%
VGG16 + SAFA	14.75	4.58	27.45%
L2LT	85.82	20.54	123.18%
L2LT Hybrid	97.91	45.10	270.54%
ConvNext	95.86	29.36	176.00%
ViT + Mixer Head †‡°	172.22	16.67	100.00%
CVST †°	86.11	16.67	100.00%

Table 3.3: Network comparison of training parameters and inference time (I.T.) on a single NVidia RTX 3090 GPU containing 24 GB on board memory for fair comparison, percentages relative to CVST.



Figure 3.11: Visualisation of the ninth head attention map extracted from CVST. The attention map of each view is projected to each computed image.

3.5 Discussion and Conclusion

In this chapter we addressed the problem of cross-view localisation on ground level images limited to a 90° FoV. An extensive ablation study on multiple state-of-the-art architectures is presented, where weight sharing and attention methods (SAFA, L2LT) are evaluated to provide a comparative analysis of the impact of each module. It is noted that the Soft Margin Loss has a positive impact on performance as shown with the VGG16 based networks. Additionally, a novel Cross-View Shared Transformer (CVST) is presented, where the network shows significant performance improvement in recall, and compared to the top three networks, CVST has less training parameters with faster inference time. Finally, we demonstrate that sharing weights in the Siamese Network consistently benefits performance for CNN and transformer architectures.

These results allow us to revisit the research questions addressed in this chapter. Regarding

the first research question, on the neural network architectures that can improve recall for ground-level cross-view IBL, the ablation study across the KITTI and CVUSA datasets (Tables 3.1 and 3.2) shows that the proposed CVST attains the highest recall of all the evaluated architectures, reaching 39.2% Recall @ 1% and 83.52% Recall @ 10% on the challenging 90° FoV KITTI test set, an improvement of approximately 5% over the next best L2LT Hybrid architecture, and 95.62% Recall @ 1% on CVUSA. These improvements are obtained while CVST retains fewer training parameters and a lower inference time than the other top performing networks (Table 3.3). The ablation also indicates that the weighted soft margin loss and the sharing of weights between the Siamese branches are the two design choices that contribute most to this improvement, with weight sharing additionally reducing the training time.

The third research question, on the techniques that can mitigate perceptual aliasing, is addressed at the feature-learning level. Perceptual aliasing manifests as incorrect retrievals in which visually similar locations are ranked above the true match, so a network that produces more discriminative embeddings is less affected by it, and this is reflected directly in the recall metrics. The recall improvement obtained by CVST, in particular the approximately 5% gain in Recall @ 1% over the next best architecture on the KITTI test set (Table 3.1), indicates that CVST better distinguishes the correct overhead view among visually similar candidates, thereby reducing the impact of perceptual aliasing during retrieval. Nevertheless, the absolute Recall @ 1% of 39.2% on the 90° FoV KITTI set shows that a substantial fraction of limited-FoV queries remain ambiguous and are still affected by aliasing. Improving the network therefore mitigates, but does not fully eliminate, perceptual aliasing, which motivates the temporal filtering approach developed later in the thesis.

This limitation, together with the strong recall obtained by CVST, motivates the remaining chapters. Chapter 4 evaluates whether the CVST architecture can be transferred to aerial platforms operating in urban and rural environments, while Chapter 5 integrates CVST within a particle filter and a road-network likelihood, using temporal information to resolve the residual aliasing identified here and to achieve robust near-real time localisation.

Chapter 4

Aerial Image Based Localization

4.1 Introduction

Visual Place Recognition (VPR) has drawn significant attention in the past decade thanks to the advantages it can offer for localisation and navigation. However, challenges persist when perceptual aliasing is present in the environment. UAV image-based localisation has been demonstrated to be effective in urban environments where geometric features such as buildings and road shapes act as distinct features to aid in localisation. However, the application of such techniques in rural environments presents additional challenges due to the absence of such distinct features. In this chapter, we apply a cross-view shared transformer (CVST) network developed in Chapter 3 to train a robust model capable of handling cross-time data in urban environments and challenging rural scenes for daylight navigation. The model is tested with sequential images, successfully localising a UAV operating over a challenging rural setting, regardless of light variations and dynamic scenes. A comparative study of the transformer network’s performance on both rural and urban cross-time datasets is presented against state-of-the-art architectures, demonstrating its superiority in the localisation task.

The use of unmanned aerial vehicles (UAV) in commercial applications has drastically increased in the last decade. Applications such as drug delivery systems, such as that developed by Zipline in Rwanda, have the potential to revolutionise access to medical treatments by making urgent health resources available to remote communities. A related

use of the technology is autonomous product delivery systems, which have attracted the interest of major corporations. Additionally, UAVs are used along with ground-based stations for recurrent inspection, imaging and surveillance purposes [73, 162]. All the previously mentioned applications require low-altitude manoeuvres and are heavily reliant on Global Navigation Satellite Systems (GNSS) for localisation. However, signal jamming, multi-path effects, and component malfunction pose significant risks to the flying platforms. Due to such potential risks, several methodologies are available for validation and fault tolerance for localisation in a situation where the use of GNSS data is not viable [26]. The use of onboard optical sensors for localisation is one solution that has the potential to improve localisation robustness in these systems.

As shown by [15, 135], visual localisation has been proven to be a feasible technique for aerial vehicles; however, a high sampling density of the area of operation is required to implement a working system. In particular, their system suffered from a limited localisation range and sensitivity to illumination. To localise an aerial vehicle, recent techniques [102, 135, 15] extract features from geo-tagged satellite imagery and build a pre-encoded database of locations, which is uploaded to the UAV. The encoded database is used as a map of the area of interest, allowing the embedding computed from the live aerial view to match the corresponding pre-computed embedding, estimating the platform position. The most common techniques used to extract image embeddings involve convolutional neural networks (CNN) and, more recently, transformer networks [8, 54, 67]. Although successful experiments to localise a quadrotor in real time have been performed, the systems are susceptible to variations in lighting conditions, dynamic scenes and visual aliasing [15].

In this chapter, the performance of image-based localisation for UAVs is investigated for both urban and rural environments. Two comparative studies of four state-of-the-art network architectures used for UAV localisation are presented, one for cross-time urban data and the second for challenging rural scenarios. To improve the robustness of the localisation systems, we propose the use of our previously introduced Cross View Shared Transformer (CVST) network as the feature extraction mechanism for aerial image-based localisation (IBL). CVST was originally developed to solve the more challenging task of cross-view localisation as seen in Chapter 3. However, we demonstrate that CVST is

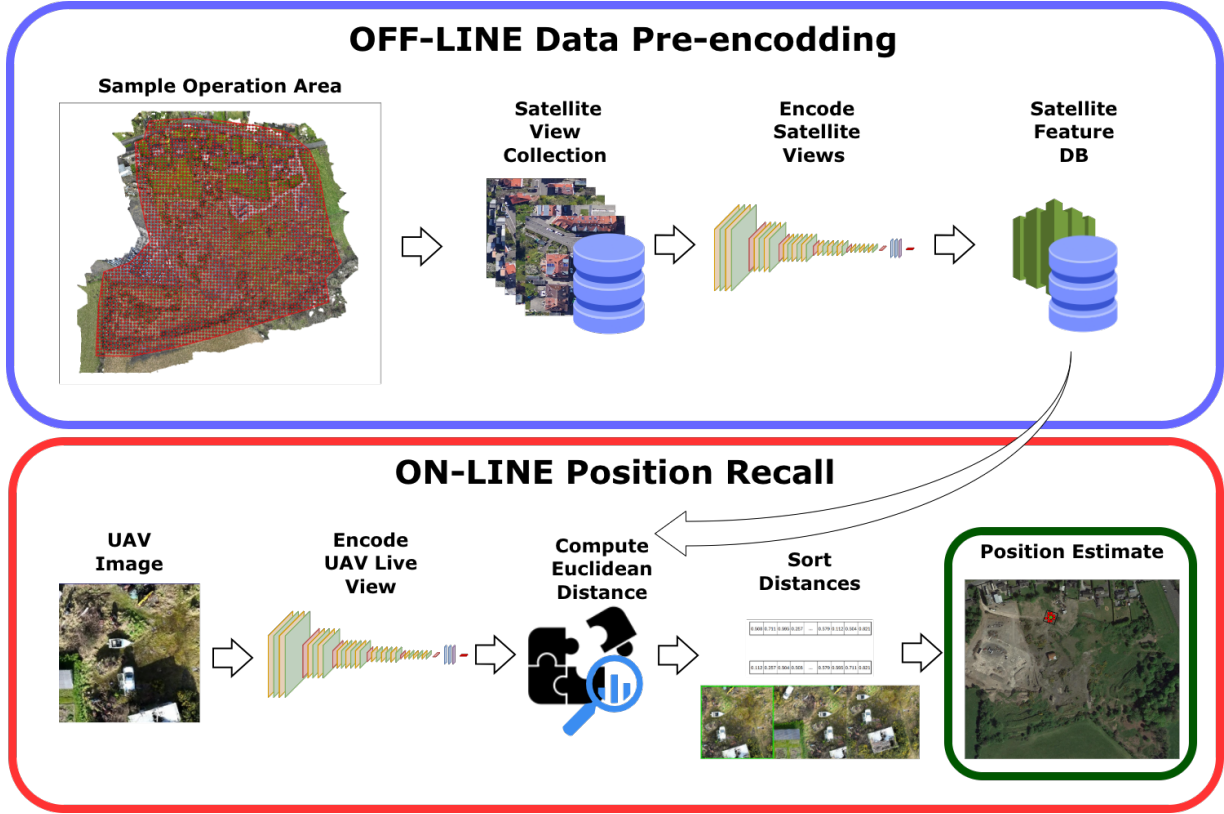


Figure 4.1: Localisation Work Flow. In Blue, the offline process of encoding the satellite image collection. In red, the iterative process of encoding the live drone feed and the localisation algorithm.

equally applicable to aerial localisation tasks and shows advantages when operating in rural conditions due to the use of attention maps. To demonstrate the superiority of CVST, we measure the recall metric for a cross-time urban dataset and a challenging rural environment. Similar to ground view localisation, we formulate UAV localisation as an image retrieval problem, where a UAV overhead view is used as a query image and compared against a pre-computed database of orthorectified images, as shown in the workflow depicted in Fig. 4.1.

In summary, the contributions of this chapter are: (i) the development of a transformer-based approach for aerial image-based localisation for cross-time data and challenging rural scenes. (ii) The improvement of the recall metric in challenging rural and cross-time urban environments. (iii) A comparative study measuring the recall metric of state-of-the-art aerial localisation architectures in a new challenging rural environment and cross-time urban data. Finally, (iv) we demonstrate the solution to the kidnapped robot problem with real UAV data captured in a challenging rural environment where the sequential UAV images are localised relative to an orthomosaic visual map employing CVST embeddings

as the position estimation methodology.

The work presented in this chapter is based on the peer-reviewed publication [9]. Eduardo Andres Avila Herrera, the author of this thesis, is the lead author of this work and was responsible for the conception and design of the method, its software implementation, the design and execution of the experiments including the collection of the rural UAV dataset in Ireland, the analysis of the results, and the writing of the manuscript. Aidan Magee assisted with the UAV data capture and the geo-referencing of the drone imagery. The co-authors, Tim McCarthy and John McDonald, contributed in a supervisory capacity, providing guidance on the research and feedback on the manuscript.

This chapter is structured in five sections. Section 1 introduces the methodology and the chapter’s motivation. Section 2 presents the State-of-the-Art research in the area of aerial image-based localisation. Section 3, details the method employed for aerial image-based localisation. Section 4 covers the details of the experimental setup and presents the experimental results. Finally, Section 5 concludes with a discussion of the findings and results demonstrated in the previous sections.

4.2 State of the Art

Current Aerial localisation systems formulate image localisation as an image retrieval task, similar to most visual localisation systems developed for ground-to-ground or cross-view localisation. Visual place recognition has migrated from classic computer vision techniques to deep learning techniques capable of computing one-shot local features, such as CNNs [67, 55, 135, 146, 15]. Most common deep learning methodologies apply CNNs to extract local image features [135, 15]. To include global representations into the features, pooling mechanisms such as NetVLAD [7] and attention [146] methods have been developed, thereby improving the performance of visual place recognition approaches. However, Tian *et al.* [135] demonstrates that local features outperform global features when searching a query image in a large satellite image. The aerial localisation solutions can be classified into approaches using CNNs [51, 104, 7], Autoencoder architectures [15], techniques using attention mechanisms [155, 10], or techniques that rely on classical feature extraction

methodologies such as Visual Odometry [116] methods. We briefly discuss each of these methodologies below.

CNN Approaches: Vanegas *et al.* [140] presented a system that searches a defined target in an obstacle-cluttered GNSS-denied environment using optical sensors. Not using a neural network that enables comparison of observations in latent space forced Vanegas *et al.* to develop complex localisation/tracking, revealing the challenges faced when attempting visual localisation with classic computer vision techniques. Jasrotia *et al.* [61] reaffirms the need for localisation technology over GNSS denied environments such as mining and remote locations, which is also required in wireless sensor networks (WSN), where data retrieval over large areas can be used for surveillance, disaster management and monitoring of harsh environments. CNNs enable the extraction of image features comparable in the latent space, as presented by Hou *et al.* [51], Kim and Walter [67] and Bianchi and Barfoot [15]. The parcimonious encoding of images into compact vectors is a key component when implementing IBL. Ding *et al.* [36] leverage the capabilities of CNNs to compare images in the latent space, performing tests with oblique aerial views captured from a UAV. Ding *et al.* propose the use of a localisation classification method (LCM) extracting features with a CNN. LCM treats the retrieval-matching problem as a classification problem. Tian *et al.* [135] address two important challenges in overhead visual localisation : cross-time image matching and orientation invariance. A Siamese network with a VGG16 backbone is proposed and trained with positive and negative pairs of satellite images from different timestamps.

VAE Approaches: Building on the results of Patel *et al.* [102] and the findings of Hou *et al.* [52], Bianchi *et al.* [15] propose a real-time system for UAV teach and repeat for autonomous *return to home* based on image localisation. A Variational Autoencoder (VAE) is a CNN encoder decoder architecture that permits the reconstruction of the input image from an encoded embedding. Bianchi *et al.* use a VAE network pre-trained with Google Earth satellite imagery using an unsupervised technique. The feature extraction methodology was based on the successful work of Hou *et al.* [51], where the researchers present the use of image feature extraction in the field of Facial Recognition. The methodology applied is relevant for Aerial View Localisation because satellite images are treated in the same

way as faces. Hou *et al.* [51] demonstrate the ability of VAE architectures to extract local features from images in an unsupervised manner and map them to a common latent space.

Although CNN or VAE architectures permit the comparison of images in latent space and have been successful, perceptual aliasing remains a significant challenge, requiring very high sampling across the search space. This causes the systems to be noisy in both rural and urban areas, as shown in our later results.

Visual Odometry: Mutual information techniques and visual odometry are proposed by [102] to locate a UAV at a 26m altitude on a sunny day, resulting in accurate localisation and heading estimation. George *et al.* [44] explores the idea of using stereo cameras for IBL. A DJI Matrice 600 is mounted with a set of stereo cameras pointing downwards, which is then flown for an 800 m trajectory. Three methodologies are tested: stereo visual odometry (VO), mono visual inertial odometry (VIO) and stereo VIO. Due to changes of the terrain topography and the presence of obstacles, altitude changes of the UAV’s trajectory are required. Such altitude changes need to be taken into consideration within IBL algorithms due to the induced variation in scale. Dai *et al.* [32] tackle this challenge by first enhancing the UAV localisation datasets and introducing a new dataset named DenseUAV. The dataset is built by extracting data from Google satellite imagery at different zoom levels. In addition, Dai *et al.* [32] propose a metric to measure image retrieval that takes into consideration the distance between the query image and the compared images.

Attention: Taking into consideration the multi-view challenge present in aerial IBL, [146] approach the feature extraction process by applying a square ring partition approach. Tian *et al.* [134] leverage a projection technique to increase the similarity between oblique aerial viewpoints and satellite images of common locations. Lin *et al.* [79] focus their research on improving the feature extraction methodology for images taken from aerial platforms. A new framework called RK-Net learns representations and detects key points using a single network. An attention mechanism named Unit Subtraction Attention Module (USAM) is introduced as the main module.

Even though there have been significant improvements in IBL methodologies for aerial platforms, little research has been done in implementing near-real time localisation in real-



Figure 4.2: Raster Image extracted

istic scenarios. Recent methods continue to identify perceptual aliasing and visual feature sparsity as a challenge for aerial IBL. This chapter applies the CVST network developed in Chapter 3 to demonstrate its recall accuracy in both rural and urban environments. We integrate the feature extraction model into a complex algorithm capable of performing near-real time localisation, as shown in Fig. 4.1. This work not only demonstrates the successful implementation of aerial IBL but also showcases the capabilities of our workflow in complex rural scenes with changing lighting conditions, thereby bridging the gap between theory and practice.

4.3 Methodology

In this section, we present a detailed explanation of the dataset collection methodology as well as the building blocks of the neural network used for testing.

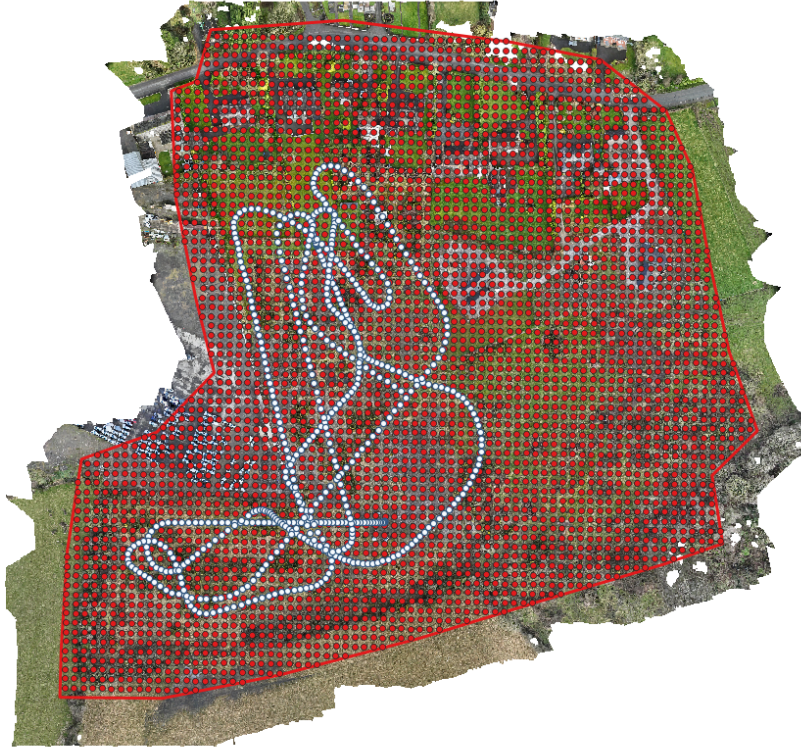


Figure 4.3: Sample grid locations shown in red and in white the random trajectory followed by the UAV

4.3.1 Dataset

In our experiments, two distinct datasets are used for training and evaluation. The first is an urban dataset captured from satellite images, where the UAV images are simulated by sampling its nadir view along a simulated flight path. The second is a rural dataset captured as part of this PhD, consisting of three trajectories flown by a quadrotor at different times of the day, incorporating lighting variations and shadowing. Further details of each dataset are shown below.

Urban Dataset: The dataset used to test the capabilities of the network for cross time images consists of $\sim 114\text{K}$ triplets extracted from random locations in an urban area over Karlsruhe. Each triplet consists of an anchor image, a positive match and a negative match. The data used for urban experiments is captured from two different online sources (Bing and Google satellite imagery) at different times of the year. During training, the time difference between data collections is used to account for lighting and scenery variations in the model. The positive and negative images are chosen from the same satellite image and the anchor from the second satellite image. Building the dataset in this manner allows

the network to learn the low frequency features such as buildings and roads, making it invariant to lighting conditions and noise. Testing imagery is extracted from a different geographical area than the one used for training. A collection of 4663 sequential locations is used to extract images from both satellite views. The images are captured from a sequence of locations following a predetermined path. The Google satellite views are used as the pre-collected database and the Bing satellite imagery is used as the query image to calculate the network performance.

Rural Dataset: To evaluate the performance of aerial IBL in real sequential data, we deployed a UAV in a challenging rural environment. The unique dataset was collected in Co. Meath, Ireland using a DJI Mavic 3 Enterprise employing a wide angle camera at a constant altitude of 40m. The camera used during collection had the following specifications: Focal Length of 12.29mm, 84 deg *FoV*, *imageresolutionof* 5280px $\times$ 3956px, video resolution of 1920px $\times$ 1080px at 30 fps. Further specifications are included in Table 4.1.

Focal Length	12.29
FoV	84°
Image Resolution	5280x3956
Video Resolution	1920 $\times$ 1080
Video FPS	30

Table 4.1: DJI Mavic 3 Enterprise Wide Camera Specifications

The test area is predominantly rural with some buildings to the north of the site as shown in Fig. 4.2. The dataset consists of an aerial survey conducted by the UAV in the operational area, paired with three random walk flights that capture images from directly overhead (nadir), all covering the same area. The orthomosaic was processed offline and used as the UAV’s reference frame during navigation.

The training data was collected in two stages. First, capturing and computing an orthomosaic and second, sampling the orthomosaic to generate image triplets. A pre-defined sequential flight pattern was used to capture a single grid of nadir imagery required to generate an orthomosaic. A total of 356 images were captured as part of the grid pattern flight to preprocess a high definition map of the area. Agisoft Methashape Pro Version 2.1.1 was used for the generation of an orthomosaic with a ground resolution of 1.42 cm/pix.

The second stage, samples the high resolution orthomosaic in a grid pattern every 5 metres as shown in Fig. 4.3. This grid is used as the anchor images to generate the triplets. The orthomosaic is randomly sampled to generate positive and negative pairs. A pair is considered positive when the distance between images is less than 5 metres apart, otherwise it is classified as negative. A total of 21637 triplets were extracted from the raster to train the models used to test the localisation of the real UAV data.

For testing, three different sequences are collected. Each flight trajectory consists of a random walk collected 2 hours apart from the imagery used to generate the orthomosaic map. The two-hour difference allowed capturing a change in lighting due to the Sun’s position, inducing shadows into the random walk imagery.

4.3.2 Network Architectures

To extract features from each overhead image, we evaluate several Siamese networks with shared weights but different backbone architectures. The recall performance is evaluated on both datasets using three different CNN architectures and the CVST network developed in Chapter 3. To assess performance in cross-time data, the urban dataset is used, and for rural scenarios, the captured UAV data is employed. The VGG16 architecture is chosen as the common CNN backbone for our study, as it has been the chosen backbone for cross-view localisation tasks, as shown by [51, 67, 55]. Due to the extreme viewpoint difference, cross-view IBL tasks face different challenges than aerial IBL. The CNN based architectures chosen for comparison are the ModVGG16 [67], VGG16+SAFA [122] and a VAE network [15]. All the networks chosen, except for the VAE, have been explored previously in Chapter 3 in the context of cross-view IBL, demonstrating state-of-the-art performance; hence, we compare them in the context of aerial IBL.

VAE: We choose the Variational Autoencoder (VAE) as one of the networks to compare due to its performance shown by Bianchi *et al.* [15] in aerial platforms. VAEs are compact architectures that have been successfully used for comparing images captured from the same viewpoint [51]. Bianchi uses this architecture to deploy a Quadrotor capable of repeating a trajectory previously explored. The UAV can navigate autonomously without

GPS by using IBL to determine its location.

ModVGG16: The modified VGG16 network (ModVGG16) implements a skip connection between the first convolution layer of the fourth convolution block and added to the last convolution layer as proposed by Kim *et al.* [67]. The skip connection computes an average pooling layer with a kernel size of 4 and a stride of 4 before adding it to the last convolution layer, allowing better information flow across the network. This network is chosen as a pure CNN encoder backbone to evaluate the baseline performance of CNN architectures in this context.

VGG16+SAFA: Spatial Aware Feature Aggregation (SAFA) [122] is implemented for our tests as it has been shown to be a compact network with 14M parameters and ranked as a state-of-the-art network for cross-view localisation tasks. The architecture consists of a VGG16 backbone and a SAFA module that computes a compact attention mechanism in the local features extracted by the CNN layers. VGG16+SAFA is chosen for its compact parameter count and demonstrates the best recall rates among CNN backbone architectures in Chapter 3.

CVST: Cross View Shared Transformer (CVST) [10] is a Siamese Network with both branches sharing weights. Each branch consists of a Vision Transformer modified with an MLP Mixer head. The Vision Transformer Network is set with 12 encoder blocks and 12 heads. The MLP Mixer head replaces the original classification head and consists of a fully connected layer followed by GELU [75] activation function and two consecutive fully connected layers to reduce the dimensionality of the feature vector and map the features to a common latent space. CVST has demonstrated the best performance in the tests carried out in Chapter 3, therefore we choose to test its capabilities in aerial IBL.

4.3.3 Training

Each network is trained for 40 iterations over the training dataset. Mini batches of 128 images are used to calculate each back propagation step. Adam optimizer is used to train the ModVGG16 and the VGG16-SAFA networks. The VAE is trained as presented in [15]. To train the CVST network the AdamW optimizer is set with a 0.3 weight decay rate. All

the networks are trained using a weighted soft margin loss function where the learning rate is set to 10E-5, following the methodology of [54, 55]. This loss function is chosen as it has been an effective feature discriminator for retrieval applications. Additionally, the weighted soft margin loss avoids the need for a fixed margin parameter. The loss function is formulated in Eq. 4.1.

$$L_{weighted} = \ln (1 + e^{-\alpha(\Delta d)}) \quad (4.1)$$

where, $\Delta d(f(I_a), f(I_p), f(I_n)) = d(f(I_a), f(I_p)) - d(f(I_a), f(I_n))$. The value of alpha is set to 10 for all the experiments as mentioned in [55]. I_a , I_p and I_n are defined as the anchor image, positive image and negative image of each triplet. $f(\cdot)$ indicate the resultant features extracted from one network branch. $d(f(I_i), f(I_j))$ represent the Euclidean distance between the feature embeddings for images I_i and I_j . Finally, Δd is the difference between positive and negative pairs.

4.4 Experimental Setup

The performance of each architecture is measured based on the recall@K metric, as is commonly used in evaluating the performance of IBL methods [55, 67]. The UAV localisation is considered successful when its corresponding ground truth image is found in the top K set of retrieved images. Fig. 4.4 and Fig. 4.5 visualises the variation of the recall metric as a function of K. As mentioned earlier, four architectures are evaluated in urban and rural environments. The first test evaluates the performance of image retrieval in urban environments captured in different years. Each network is applied to a cross-time dataset of satellite images captured in different years, under different lighting conditions and with different sensors. The different architectures are compared based on the recall metric calculated when applied to a sequence of 4663 images captured from the cross time data set.

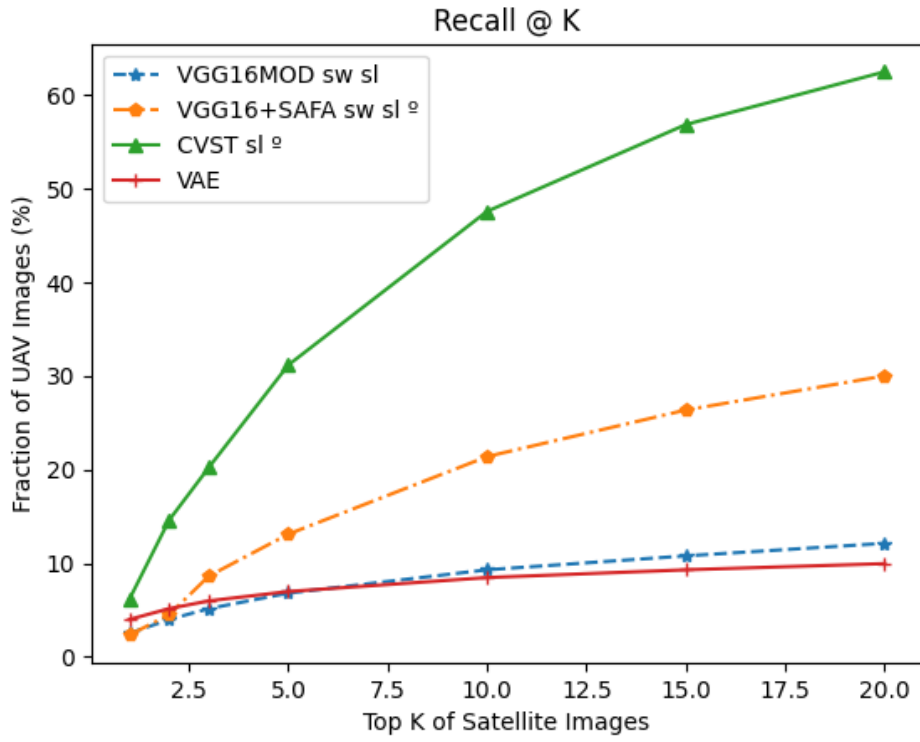


Figure 4.4: Recall results for different architectures extracted from the grid samples and test trajectories.

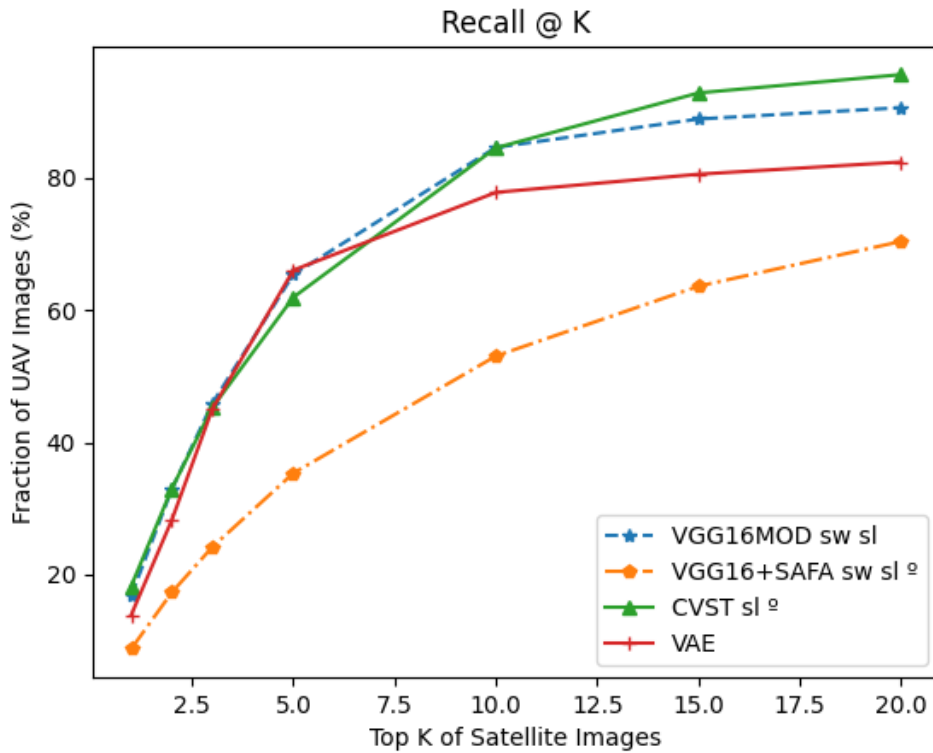


Figure 4.5: Recall metrics computed for the cross-time dataset.

The second study demonstrates the use of IBL in challenging rural conditions. This study

tests the ability of the CVST network to locate the UAV undertaking three flights over an area of 428×400 metres, as shown in Fig. 4.5. The test area is sampled every 5 metres to pre-encode embeddings using the CVST network as shown in Fig. 4.3, resulting in less sampling density than Bianchi *et al.* [15], who sample every 0.5m. The embeddings are concatenated with their corresponding coordinates into a database that serves as a visual map of the environment. The query image is compared against the entire database using the Euclidean distance, and then all the distances are sorted in ascending order, allowing for the retrieval of the top ten matches. The ten locations with the lowest distances are used to estimate the position of the UAV. Fig. 4.1 depicts the complete offline and online workflow used for the localisation system. The complete process is repeated at a rate of 2 frames per second using a 64GB, Ryzen 7 CPU and an NVIDIA 2080Ti GPU.



Figure 4.6: Attention Maps Visualisation of aerial views at different time and lighting conditions.

Attention Map Visualisation: The attention maps generated by the CVST network are visualised in Fig. 4.6 where it can be seen that the network learns to identify similar features in both images regardless of the lighting and scene variations. We note a tendency where features such as roofs and road shapes help the network to match between scenes

in urban environments. In rural scenes, the Transformer Network uses changes in grass tone, clutter, and trees to identify similarities across images. However, these are only observations taken from the attention maps computed from our tests.

4.5 Results and Discussion

Each architecture is compared in Table 4.2 where the recall at top K metrics are presented. It can be noted that CVST outperforms ModVGG16 by 2% at the R@1 metric and 5% at the R@10 metric. It can be noted that the CVST network outperforms all the networks in the R@1% metric and R@1. Results show that CVST is capable of locating a query image with more confidence than the compared architectures, as shown in Fig. 4.9. Fig 4.10 shows examples of how CVST classifies the correct location in the top ten matches.

Comparative Study							
Architecture		Cross Time Test			Rural UAV Test		
Name	Parameters (M)	R@1% (%)	R@5 (%)	R@10 (%)	R@1% (%)	R@5 (%)	R@10 (%)
VAE	194.02	88.1	66.01	77.76	13.26	6.97	8.46
MODVGG16 †	134.26	93.33	65.37	84.56	17.74	6.80	9.29
VGG16 + SAFA †	14.5	85.7	35.34	53.04	38.30	13.10	21.39
CVST † ^o	86.11	98.05	61.91	84.5	72.80	31.17	47.60

Table 4.2: Results on the 5K test set. The symbol † denotes that the model was trained using soft margin loss. ^o represents networks that have been pretrained with ImageNet weights.

Additionally, we test the robustness and generalisation ability of the CVST network by testing the model pre-trained for a cross view localisation task in the cross time aerial data. The tests revealed that CVST requires task specific training to perform at its best. A CVST network trained for cross view localisation tasks is tested for the aerial localisation and it can be noted the performance is affected. Success examples when applying the recall process are visualised in Fig. 4.7 and Fig. 4.8 for rural and urban environments respectively.



Figure 4.8: Example of successful aerial images queries in a cross-time urban environment. The green box depicts the correct match and in blue the ground truth pair. Data Attribution: Imagery 2023 CNES / Airbus, GeoBasis-DE/BKG, GeoContent, MaxarTechnologies, Maps Data: 2023 Google.

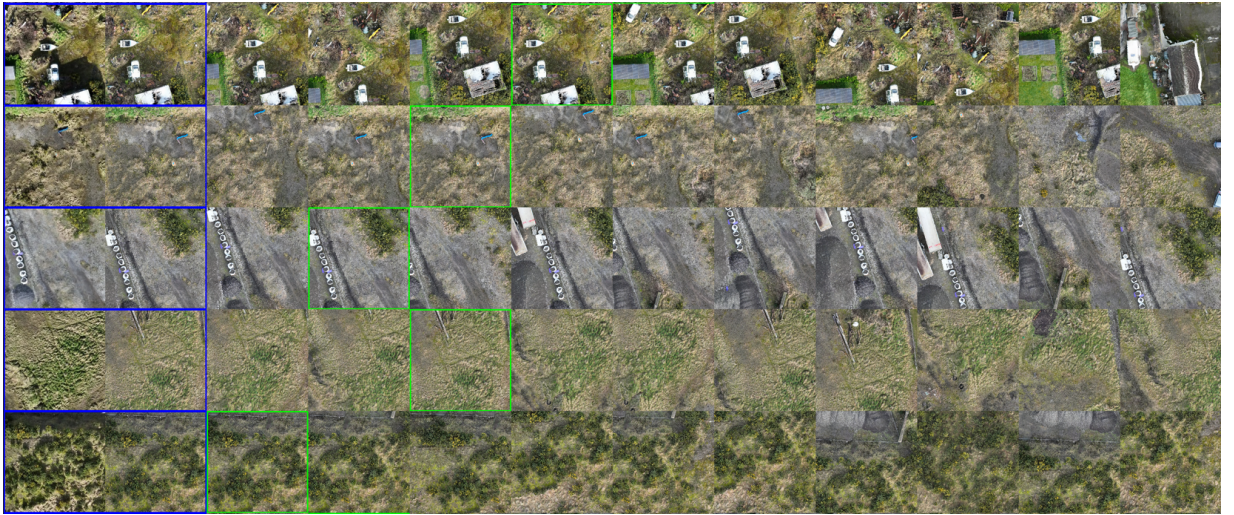


Figure 4.7: Example of successful aerial images queries in a rural environment exposed to lighting variations. The green box depicts the correct match and in blue the ground truth pair. Data Attribution: Imagery 2023 CNES / Airbus, GeoBasis-DE/BKG, GeoContent, MaxarTechnologies, Maps Data: 2023 Google.

The recall results computed from the rural dataset for are visualised in Fig. 4.4. CVST shows superiority across all architectures in the rural environment, maintaining high recall when $K > 15$.

The recall over the urban dataset presented in Fig. 4.5 shows a significant improvement compared to the Rural recall. It can be seen from Fig. 4.5 that CVST, VGG16MOD,

and the VAE architecture perform very similarly when $K < 10$. However, as K increases, CVST outperforms all the architectures.



Figure 4.9: Examples of the IBL system successfully estimating the location of the UAV. The green particle shows the ground truth of the UAV, and the red show the top 10 position estimates.

Based on the results for cross-time image-based localisation, we demonstrate the CVST network’s ability to estimate a drone’s position by measuring the similarity between the current live drone image and a database of pre-encoded embeddings derived from a geo-referenced orthomosaic. The recall results for the different architectures shown in Fig. 4.4 indicate that all models tested on the cross-time dataset achieve lower recall in the rural setting. This demonstrates that IBL in rural areas presents a more challenging localisation task due to the nature of the scenery. The VAE and ModVGG16 architecture fail to find the correct match for the query image, while the CVST network maintains a recall level that allows the system to identify the correct location of the query image during the majority of the sequence. To test the performance of CVST, we localise a real drone flying a random walk pattern. Fig. 4.9 shows an example of the UAV being correctly estimated during motion and Fig. 4.10 shows a sample of frames from a sequence showing the localisation system working on temporal data. The complete video can be found at <https://youtu.be/p8IRbvpWZF0>.



Figure 4.10: Sample of frames from localising a drone over an extended trajectory in a rural environment. See <https://youtu.be/p8IRbvpWZF0> for the complete sequence.

The temporal sequence tests reveal the loss of continuous localisation as a consequence of repeated visual patterns across different locations. Having sparse visual features and an ambiguous rural scene caused by similar vegetation results in perceptual aliasing. The system operates without a temporal filter, so no attempt is made to mitigate the noise generated by the neural network. Scenes with similar visual patterns will trigger a high similarity metric, which, in theory, is correct, as in fact the scenes are visually similar; hence, a temporal filter is recommended (see Chapter 5). Fig. 4.11 shows the cases where localisation is lost, and it can be noted that the images selected for possible locations are similar to the UAV live feed. Applying a temporal filter, such as a Kalman filter or particle filter, can mitigate and solve this sudden localisation loss.



Figure 4.11: Examples of failure in estimating position due to ambiguous vegetation present in a rural scene. The green particle shows the ground truth of the UAV, and the red show the top 10 position estimates.

4.6 Conclusion

In this chapter we present a study of aerial image base localisation (IBL) in urban and rural environments, demonstrating improvements in aerial IBL when implementing CVST as the feature extraction mechanism. Furthermore, we presents a recall metric comparison between the CVST network and state-of the-art architectures used for UAV localisation. Our results show that CVST outperforms the compared architectures in urban and rural environments. CVST extracts features that allow localising a UAV in a cross-time dataset and a challenging rural environment. In summary, we have identified a significant improvement in localisation accuracy when using a modified transformer based architecture for the task of UAV image-based localisation and demonstrate the use of embeddings to localise an aerial robot using real temporal data. However, the lack of temporal filtering does not permit continuous localisation, allowing the system to lose track of position at certain instances of the sequence. To overcome this challenge we integrate the methodology presented in Chapter 3 with a bayesian filter presented in Chapter 5, where the advantages of temporal filter are demonstrated.

Chapter 5

Temporal Filtering in Image-Based Localisation

5.1 Introduction

In this chapter, we present a novel method for cross-view image-based localisation (IBL), capable of operating with limited field of view (FoV) images while performing in near-real time. Cross-view IBL is the task of geolocating a ground level image into a georeferenced satellite image. State-of-the-art cross-view IBL systems have demonstrated the ability to position a ground vehicle in complex urban environments [67, 55, 38, 39]. The majority of the research has focused on panoramic images while ignoring the complexities faced when implementing IBL on a limited FoV image [142, 54, 55, 122, 106, 124, 123, 155, 166, 146, 38, 33, 160, 77, 34]. However, perceptual aliasing and sparse visual features remain the main challenges for visual localisation (VL) with limited FoV, showing low recall performance in retrieval tasks [81, 67, 145, 125, 39, 10]. Extreme viewpoints between query image and satellite image datasets makes matching visual features highly challenging, resulting in noisy position estimates [67, 55, 39]. Recent works have focused on addressing the geometric issues at the cost of using computationally expensive transformations [122, 123, 106]. To address the problem, research has developed from CNNs [152, 81, 142, 67, 54, 122, 130, 55, 124, 123, 79, 38, 146, 147, 33], GANs [106], Transformers [155, 165, 164, 168, 160, 10] to ConvNext [33] networks providing researchers with a wide range of approaches. Yet

problems such as perceptual aliasing, robustness, simplified training methodologies, and runtime efficiency remain unsolved, particularly for limited FoV cross-view IBL.

To mitigate perceptual aliasing, it is necessary to integrate a temporal filter into the IBL process when applied within a mobile robotic context [67, 55, 38, 39]. Particle filters are computationally expensive, limiting systems from operating in near-real time [67], however precomputed embeddings allow the use of data structures to improve runtime efficiency [15, 10, 39]. Downes *et al.* implement a 2D KD-tree that allows near-real time operation of cross-view IBL. This work aims to simplify the training methodology while maintaining near-real time operation, mitigating perceptual aliasing and improving robustness. We present a novel adaptive formulation for the proposal distribution used in the particle filter, allowing weighting between image-based and road network-based likelihoods.

The methodology presented in this chapter extends the work of Chapters 3 & 4 by integrating the CVST network within a temporal filtering framework. The resultant implementation provides for improved reliability in the localisation estimates, operating in near-real time performance. To reduce the training complexity presented by [39, 166], we propose training the neural networks as proposed by Hu *et al.* [54, 55] and pre-sampling 1,000 satellite images over random locations every 15 deg. Each satellite image is pre-encoded and ordered into a 3D KD-tree, which provides an efficient matching mechanism for the particle filter when computing particle weights.

The contributions of this chapter are: i) Development of a novel localisation pipeline comprising a CVST network, a particle filter and a 3D KD-tree capable of near-real time operation. ii) Demonstration of the reduction of the number of particles required as a consequence of using the CVST network. iii) Presentation of a novel comparison of different architectures working in conjunction with a temporal filter. iv) Proposal of a novel likelihood formulation for the particle filter that mitigates against perceptual aliasing.

5.2 State of The Art

Previous research has demonstrated that training neural networks for IBL on 360 deg panoramic images yields high recall metrics [54, 55, 10, 38]. However, implementations utilising limited field of view images have proven to be more challenging [142, 67, 10, 39]. Various works have implemented a sequential Bayesian filter, also known as a particle filter, to fuse pose estimates over time, thereby reducing noise induced by perceptual aliasing [67, 54, 55, 38, 39]. These methods have been applied accross a wide array of mobile robotic platforms, demonstrating the advantage of leveraging temporal information within the localisation process. Downes *et. al* and Hu *et. al* [54, 55, 38] demonstrate that perceptual aliasing is present even when using panoramic images, also requiring a particle filter despite the high recall achieved when using 360 FoV configurations. This issue becomes more pronounced when operating on limited FoV images, where dealing with perceptual aliasing makes the particle filter essential often requiring meticulous tuning [67, 39]. A known issue with particle filters is their potentially high computational requirements [67], making them unsuitable for real-time processing in high-dimensional systems [39]. To address this, the use of precomputed embeddings is required [15, 9]. Downes *et. al* [39] introduces a system capable of handling a large number of particles by pre-processing a pose-aware satellite image to compute embeddings in a similar technique to the one used by Bianchi *et. al* [15] and the method presented in Chapter 4 [9]. Downes uses the VGG16 backbone paired with a Spatial-Aware Feature Aggregation (SAFA) module developed by [122] to encode the images into embeddings. By ordering such embeddings in a KD-Tree, Downes *et. al* attains near-real time localisation. The system demonstrates the convergence of the particle filter over a large search area; however, it requires processing of several kilometres of a trajectory prior to achieving a stable pose estimate.

Other systems attempt to incorporate other sensors and data types to improve upon recall metrics. Hu. *et. al* [53] utilise the structure of roads extracted from LiDAR measurements to improve the recall metric of cross-view localisation by matching the pointcloud structure with the road shapes found in the satellite images. However, this method is not tested in a trajectory and only operates with a LiDAR unit. Hu *et. al* [57] transform a horizontal

image into a birds-eye view and segments classes (building, roads, parks, etc). The semantic segmentation prediction is then compared with a satellite image combined with an OpenStreet Maps satellite view. The novel cross-view matching with road network data improves recall and helps the neural network learn common features.

Recent research has demonstrated that using static satellite images allows the problem to be scaled into city-wide scenarios in near-real time [39]. In this chapter, we aim to reduce the number of particles required in particle filter based formulations by applying CVST to the limited FoV problem and also improving the convergence time by leveraging knowledge of the road network. We propose a novel likelihood formulation of the particle filter working in conjunction with a 3D KD-Tree that improves robustness by mitigating perceptual aliasing while maintaining near-real time performance. Our aim is to demonstrate the improvements over limited FoV cross-view IBL that can make such systems more efficient, accurate and robust to perceptual aliasing.

5.3 Methodology

In this section, we first present the detailed methodology used to formulate the particle filter. Next, the description to integrate the KD-tree with the IBL system. Followed by a detailed explanation on sampling the training dataset and training the network. Finally, we conclude by detailing the pre-encoded Dataset ordered by the KD-Tree.

The limited FoV cross-view localisation problem is implemented as an image retrieval task, where a ground level image is searched in a catalogue of georeferenced satellite images. The georeferenced satellite image catalogue is precomputed and ordered in a 3D KD-tree; further details will be provided later in the chapter. The match image is determined by computing the euclidean distance between the query image encoding and the precomputed embeddings stored in the KD-Tree. To filter for perceptual aliasing a particle filter is implemented in a tight connection with the KD-tree as shown in Fig. 5.1.

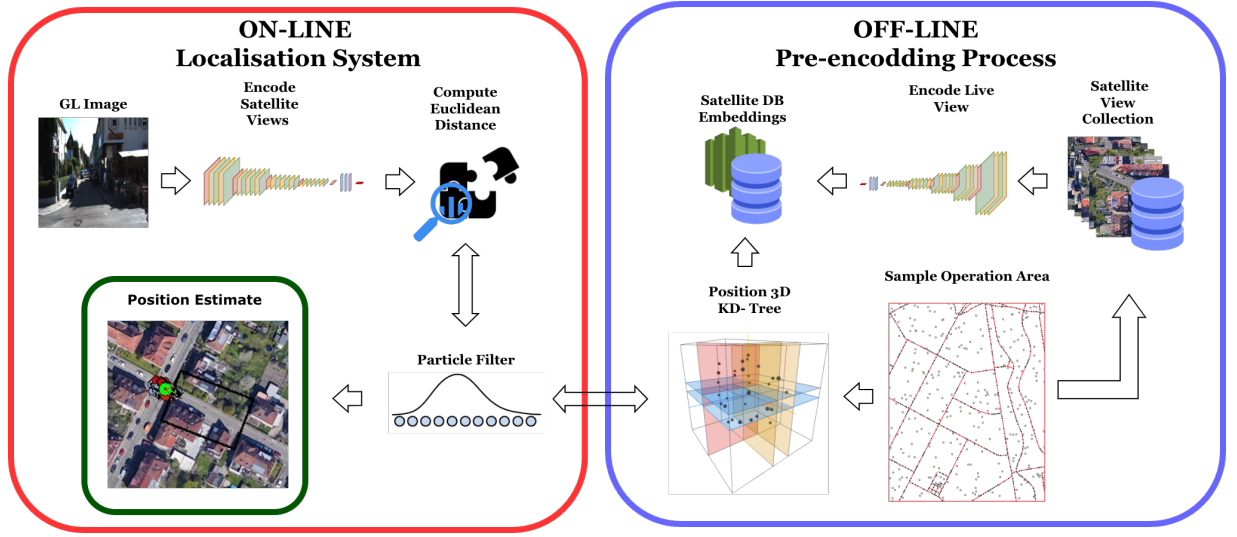


Figure 5.1: Workflow used in both the online and offline stages of the IBL System. In red, the online sequential algorithm encodes the ground level image, then the particle filter queries the KD-tree to retrieve the corresponding embeddings which are compared. Finally, the particle filter computes an estimate of the vehicles pose. In blue, the offline pre-encoding process employed to generate a 3D KD-tree structure.

The complete system consists of an offline stage and an online stage, similar to the one presented in Chapter 4. The offline stage computes the query database, which contains the embeddings for each sample, and implements the 3D KD-Tree. The online stage computes the embedding for the ground-level image captured by the mobile robot and feeds this embedding to the particle filter. The Particle Filter contains randomly initialised particles over the road network, which iteratively query the KD-Tree for satellite image embeddings corresponding to the closest pose of the particle.

With both the ground-level image embedding and a set of particle embeddings, each particle is assigned a weight. Each weight is calculated based on the likelihood, as explained in the Section 5.3.1. The particle filter iteratively filters out locations with low probability, resulting in an estimate of the mobile vehicle’s pose. Fig. 5.1 shows a workflow diagram that depicts the flow between the offline and online stages.

To test the system’s performance, we choose three architectures: VGG16MOD [67], VGG16+SAFA [122], and CVST [10], all developed in Chapter 3. We experiment with and without the road network formulation to evaluate the impact of our proposed methodology.

5.3.1 Particle Filter

The Particle Filter is a sequential Bayesian filter that approximates posterior distributions using importance sampling with weighted samples (particles) drawn from a proposal distribution. This sequential Monte Carlo approach is particularly effective for nonlinear and non-Gaussian estimation problems.

The particle filter has been widely used in robotics estimation problems, particularly in mapping and localisation [67, 55, 38, 39]. The particle filter uses an initialisation step that stochastically proposes a set of N particles $\mathcal{P} = \{P^{(1)}, P^{(2)}, \dots, P^{(N)}\}$, where each particle represents a possible vehicle pose. A particle $P^{(i)}$ consists of a pose $x_t^{(i)}$ at time t and a weight $w_t^{(i)}$, $P^{(i)} = \{x_t^{(i)}, w_t^{(i)}\}$. At initialisation, a uniform distribution is used to ensure equal probability across the environment. Using an iterative process of predictions and update steps, weights for each particle are computed on each iteration and then resampled based on the sensor measurements. After a series of iterations, the aim is for the filter to converge to a pose estimate. The particle filter is a process used previously for IBL tasks to mitigate perceptual aliasing [67, 55, 38, 39].

Once the particles have been initialised, the system propagates each particle using the odometry captured from the vehicle's onboard sensors in conjunction with a vehicle motion model. These sensors are very reliable and accurate, providing measurements for the vehicle motion with low noise. During the motion step, each particle is propagated stochastically by sampling from a motion distribution. The motion distribution is formulated based on the actuation method. The odometry parameters are used to propagate the vehicle, adding noise to the motion model. In particular, sampling from the motion model ensures that the noise level corresponds to the physical devices.

After propagation, each particle represents a pose hypothesis that incorporates the platform's motion. For each particle, a weight is computed by evaluating the likelihood of the sensor measurement given the particle's proposed pose and the known map. On a generalised mobile robotic platform, this measurement could be a single scalar from an ultrasonic sensor, a series of measurements from a 2D LiDAR, or a point cloud from a 3D LiDAR. Various formulations for computing the likelihood are available for image-based

localisation. In this implementation the likelihood is calculated using either the Euclidean distance [67] or cosine similarity [55, 39] between the current ground level image embedding $\mathcal{E}_g$ and the particle's current satellite image embedding $\mathcal{E}_s^{(i)}$.

Each measurement z_t extracted at time t will be used to compute a weight using the measurement likelihood $p(z_t|x_t, z_{t-1}, u_t)$. The likelihood can be described as the probability of the current measurement given the current pose, current proprioceptive sensor measurements, and the series of previous measurements.

The weights $w_t^{(i)}$ are updated on every iteration, and each time the particles are propagated by multiplying the previous weight by the new likelihood. At initialisation, each particle is assigned a weight based on the number of particles $w_{t=0}^{(i)} = \frac{1}{N}$

$$w_t^{(i)} = \frac{w_{t-1}^{(i)} p(z_t^{(i)}|x_t^{(i)}, z_{t-1}^{(i)}, u_t^{(i)})}{\sum_{j=1}^N w_j}, \quad \text{with} \quad \sum_{i=1}^N w_t^{(i)} = 1 \quad (5.1)$$

In the image-based localisation systems presented by Kim and Downes *et al.* [67, 39], the particle filter uses the Euclidean distance or the cosine similarity measured between the ground truth embedding and the embeddings computed from the satellite image samples. This distance is used to calculate the likelihood of each particle, which is used as its weight.

The likelihood of the particle filter is modelled as proposed in [67], shown in Eq. 5.2, due to the fact that the α allows the possibility to be tuned during experimentation.

$$p(z_t|x_t, z_{t-1}, u_t) = \alpha e^{-\alpha d(\mathcal{E}_g, \mathcal{E}_s)} \quad (5.2)$$

However, it can also be modelled as a normal distribution $\mathcal{N}(0, \sigma)$ with a standard deviation σ and a zero mean $\mu = 0$ estimated from a calibration process as proposed by Downes *et al.* [55, 39].

$$p(z_t|x_t, z_{t-1}, u_t) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-d(\mathcal{E}_g, \mathcal{E}_s^{(i)})^2}{2\sigma^2}} \quad (5.3)$$

We propose a combined likelihood measurement that uses both a measurement from the image while leveraging the road network to improve the accuracy. For the image part

of the likelihood, we employ the alpha parameter, as proposed by [67], to maintain the tuning option. However, we use the square of the Euclidean distance to simulate a normal distribution, as used by [55, 39]. The probability of the measurement given the current pose and satellite map is defined as $p(z_t|x_t, M)$ as shown in Eq. 5.4. Where $M \equiv$ Satellite map.

$$p(z_t|x_t, M) = \alpha e^{-\alpha d(\mathcal{E}_g, \mathcal{E}_s)^2} \quad (5.4)$$

The probability of the measurement given the current pose and the road network is defined as $p(z_t|x_t, R)$ as shown in Eq. 5.5. Where $R \equiv$ Road Network.

$$p(d_{r_t}|x_t, R) = \frac{1}{r_w \sqrt{2\pi}} e^{\frac{-d_r^2}{2r_w^2}} \quad (5.5)$$

Where $r_w \equiv$ road width and $d_r^{(i)} \equiv$ Perpendicular distance of the particle to the closest road, as shown in Fig. 5.2.

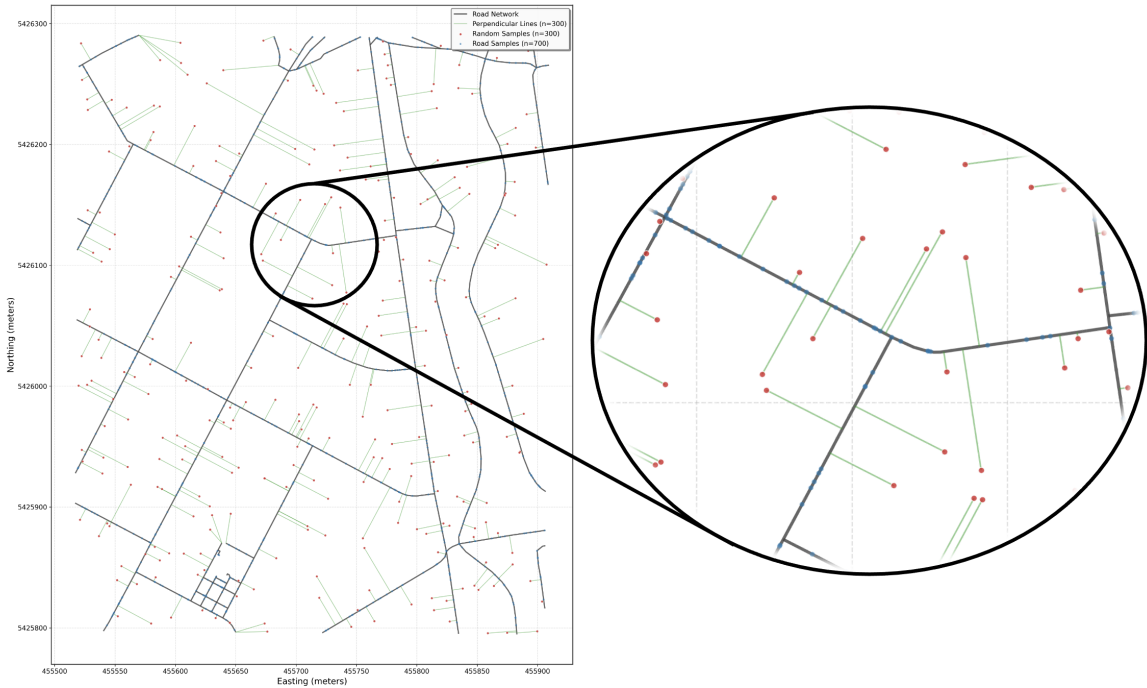


Figure 5.2: Closest perpendicular distance of a particle to the road.

Additionally, we introduce β and γ tuning parameters that allow relative weighting of the image-based likelihood and the road network-based likelihood.

$$p(z_t|x_t, z_{t-1}, u_t) = \beta\alpha e^{-\alpha d(\mathcal{E}_g, \mathcal{E}_s)^2} + \gamma \frac{1}{r_w \sqrt{2\pi}} e^{\frac{-d_r^{(i)2}}{2r_w^2}} \quad (5.6)$$

To avoid particle depletion low-variance resampling is triggered when the effective sample size (ESS) drops below 80% of the total particle count. The effective sample size is using the standard expression shown in Eq. 5.7

$$N_{eff} = \frac{1}{\sum_{i=0}^N w_t^{(i)2}} \quad (5.7)$$

5.3.2 KD-Tree

K-Dimensional Trees (KD-Trees) are data structures that organise data points through recursive spatial partitioning. They structure the data according to parameters such as the maximum number of divisions and the spatial distribution of the points. This organisation enables efficient nearest-neighbour searches and significantly accelerates N-point query problems. Previous research has leveraged 2D KD-trees in IBL [39], achieving successful results. This works when the satellite image is sampled as a non-overlapping grid and positional-aware embeddings are available. The satellite image sampling is similar to the one shown in Figure 4.3 and presented by Bianchi *et al.* [15]. This sampling method limits the resolution permitted unless tiles overlap, while tiles in cross-view IBL can cause visual ambiguity, leading to perceptual aliasing.

Downes *et al.* [39] pre-sample the satellite image as a non-overlapping grid and precompute position-aware embeddings of each cropped image. The encodings are stored in a 2D KD-Tree structure that enables fast position based queries. Applying this structure in IBL requires the training process to incorporate knowledge of ground-level image orientation. This specific implementation requires the training data set to consider visual-feature sparsity in cases where the ground level image field of view falls outside the matching satellite tile.

To avoid the complexity of training position-aware embeddings and sampling with non-overlapping tiles while maintaining the advantages of the KD-tree, we propose using a 3D KD-tree with eastings, northing and orientation as the partitioning parameters. Figure

5.3 provides a 3D Cartesian visualisation of the structure to represent the data ordering. This data structure works hand in hand with our sampling process when precomputing the satellite image embeddings. The visualisation purposefully biases the sampling to 50% Top Right, 30% Bottom Left, and 10% Top Left and Bottom Right to show the structure’s behaviour. The KD-Tree employed in the experimentation orders 24,000 samples.

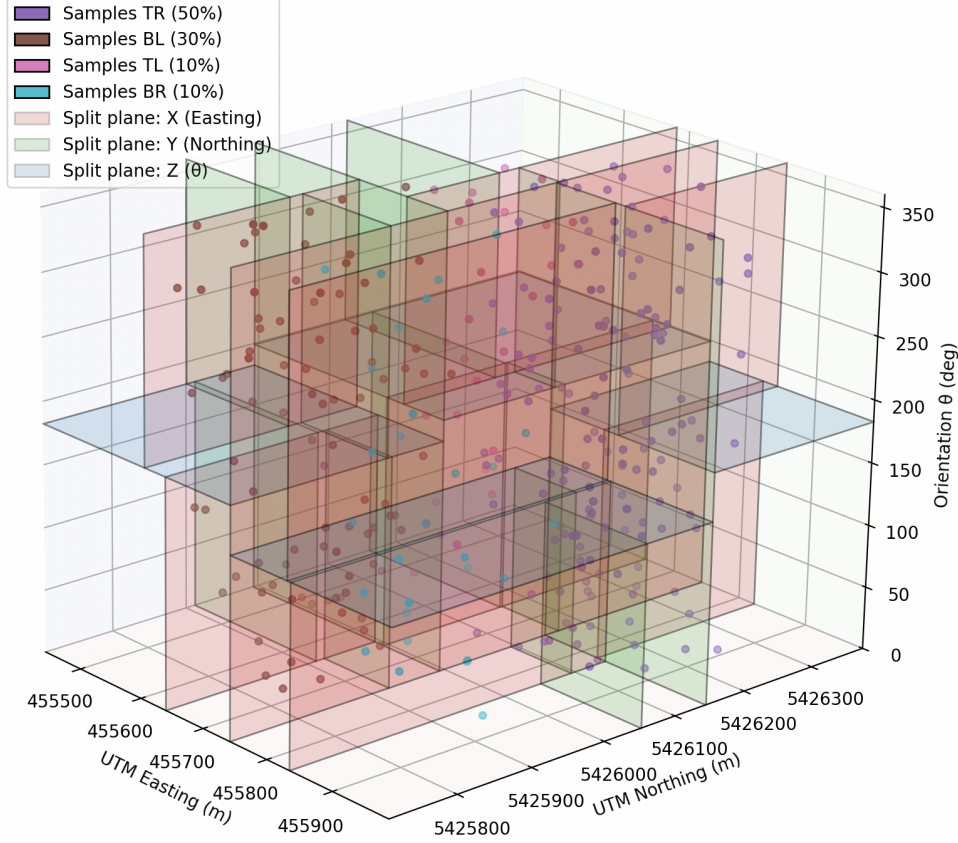


Figure 5.3: Visual representation of a 3D KD-tree. The x and y axis represent the coordinates of the satellite images, and the z axis represents the orientation of sampling.

The KD-Tree serves as a search function that returns the satellite image embeddings given the current particle pose, as shown in Eq. 5.8. Introducing this sampling methodology increases the sampling density at each sampled location.

$$\mathcal{E}_s^{(i)} = KD - Tree.query(x^{(i)}) \quad (5.8)$$

We employ the KD-Tree as an acceleration structure capable of returning the relevant embeddings efficiently, thereby avoiding the need to perform satellite image inference at every iteration of the particle filter.

5.3.3 Network Architectures

To test the performance of the IBL system with different architectures, we choose to compare the CVST network developed in Chapter 3 with the VGG16MOD network presented by Kim *et al.* and the VGG16+SAFA network presented by Shi *et al.* [122] used by Downes *et al.* [39] when implementing the wide area IBL system. These networks have been introduced in previous chapters and are chosen for their relevance to the limited FoV IBL.

5.3.4 Training Dataset

In this chapter, we use the KITTI dataset, introduced and used in chapters 3, for our experiments. In particular, we employ both the image data (for computation of the embeddings and likelihood), and also, the odometry information for computation of the motion update for each particle. Over the years, this dataset has become one of the standard benchmarks for multiple computer vision tasks, among those, the limited FoV localisation, as shown by [67, 39].

Corresponding satellite images are extracted from a georeferenced raster 5 meters in front of the sampling location. The satellite image covers 70 meters in the forward-facing direction of the orientation and 30 meters laterally. This approach was initially proposed by Kim *et al.* [67], as the features from the ground-level image will be present in the corresponding satellite image in front of the vehicle. This is in contrast to the sampling method in Downes *et al.* [39], where the satellite image is a square area with a fixed north direction, extracted in a reticle form from the search area.

We then generate image triplets composed of a ground-level image, its corresponding matching satellite image, and a random negative satellite image with an offset distance set to be greater than 100 meters. Approximately 70,000 triplets are generated and used for the training of all the networks.

5.3.5 Training

To test our system, we train a collection of networks on the same sequence of images. We chose sequence 0027 from the KITTI dataset for testing the complete localisation system. This sequence was not used during training or validation of the neural networks.

To train the networks, we employ the soft margin triplet loss presented by Hu *et. al* [54] and successfully used in other works [55, 155, 124, 166, 164, 9, 10, 39], which is shown in Eq. 5.9.

$$\mathcal{L}_{weighted} = \ln (1 + e^{-\alpha(\Delta d)}) \quad (5.9)$$

Eq. 5.9 has been successfully demonstrated as a loss function that allows the different network architectures to learn features from cross-view images.

Each network is trained for 60 complete iterations over the entire database, using a learning rate of 0.01 and the Adam optimiser. The networks are trained and implemented using the PyTorch library.

5.3.6 Query Database

After training the networks, the testing area is pre-sampled from a geo-referenced satellite image. This raster functions as the static map where particles obtain their measurements with respect to a fixed reference frame.

We sample the road network present in the area using a uniform distribution, as shown by the black locations in Fig. 5.4. Each computed coordinate is sampled every 15 degrees, using the same characteristics as those in the training dataset, generating 24 satellite images per coordinate. In addition to the road samples, a uniform distribution is used to sample locations over the complete area without any restriction, as shown by the white locations in Fig. 5.4. Coordinates are now spaced throughout the region, including zones between roads and other places. For each sample, the process is repeated every 15 degrees.

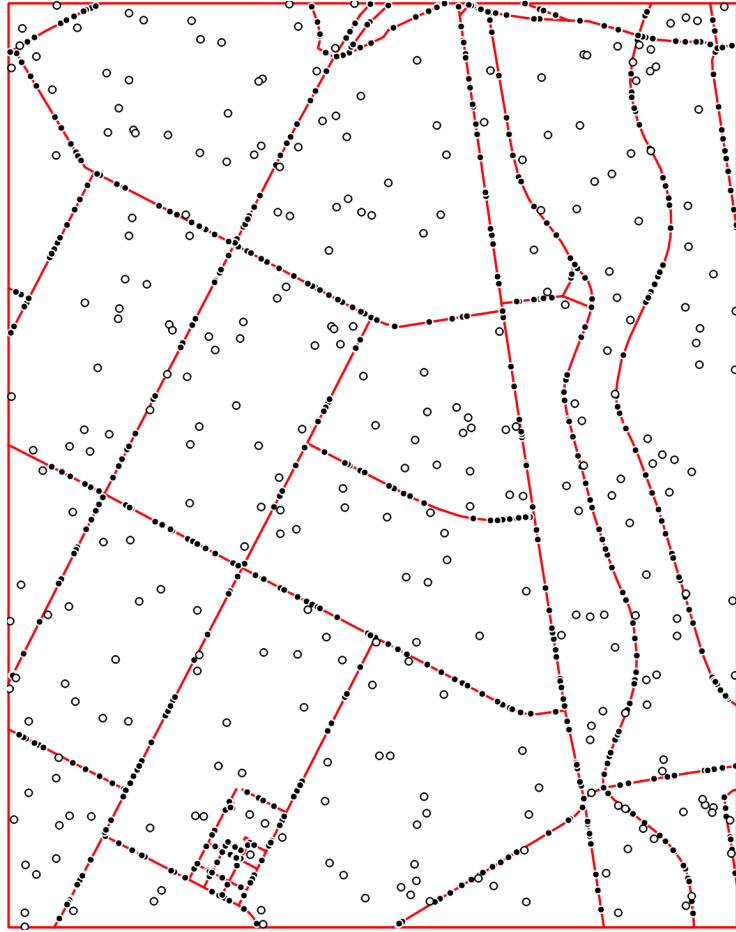


Figure 5.4: Sampling methodology used to pre-process the KD-Tree databases. Samples captured over the road network are shown in black, while non-road network samples are shown in white.

After sampling the satellite images, each sample is encoded by the trained neural network, yielding approximately 24,000 embeddings, each associated with a corresponding pose. To facilitate fast embedding matching, the database is ordered in the 3D KD-Tree described previously. The particle filter uses each particle’s location to retrieve the corresponding embedding based on position and orientation.

5.4 Results

We test the methodology on trajectory 0027 from the KITTI Dataset, which was not used in the neural network’s training, and does not overlap geographically with the training sequences. We perform the experiments with different neural network architectures to evaluate the impact on the estimated convergence of the particle filter and localisation

accuracy. Each test is evaluated both with image-based localisation alone and with the road-assisted distribution. The particle filter is initialised with particles uniformly distributed over the road network, assuming the probability of a vehicle being in between roads is very low.

To measure the accuracy of the methodology, the GPS data provided by the KITTI dataset is used to calculate the position error, which is computed as the delta between the average particle filter position and the ground truth GPS location. Correct localisation is assumed once the error falls below 10m from the ground truth position. To measure the convergence of the particle filter, an analysis of the standard deviation of the particles is presented. When the standard deviation falls below a desired threshold, it is considered that the system has converged.

5.4.1 IBL vs Road Assisted IBL

To evaluate the impact of IBL vs. Road-Assisted IBL, tests of the sequence with and without the probability of the measurement given the current pose and the road network $p(z_t|x_t, R)$, by defining $\gamma > 1$ or $\gamma = 0$ respectively, depending on the test. Position error results of each architecture evaluation are presented in Fig. 5.6, 5.11, 5.8, 5.14 and 5.17. These tests show that although perceptual aliasing is prominent, the CVST network paired with the particle filter is robust and converges to a position error of less than 10m. Results demonstrate that introducing likelihoods relevant to the road network reduces the localisation error of all architectures to below 10m.

When testing the performance of the VGG16 network with the Particle Filter, the simulations demonstrate that the neural network struggles to learn cross-view features, resulting in noisy measurements. This noise is fed to the Particle Filter, causing convergence instability. With noisy measurements, the Particle Filter is susceptible to particle depletion, trapping the estimate in a local minima, which can prevent the system from converging or cause it to converge in the wrong location.



(a) IBL

(b) Road Assisted

Figure 5.5: Particle Filter converging to the correct position when implemented with VGG16.

An example of this is shown in Figure 5.5(a) where perceptual aliasing caused by the noisy measurements, results in particles converging. The error and the standard deviation of the particles oscillate as the Particle Filter struggles to converge into the correct location, shown in Figure 5.7. When the road network assists the system, Figure 5.5(b) shows the same example, but with the addition of the road network likelihood term where it can be seen the clustering of particles around the ground truth, estimating the correct position.

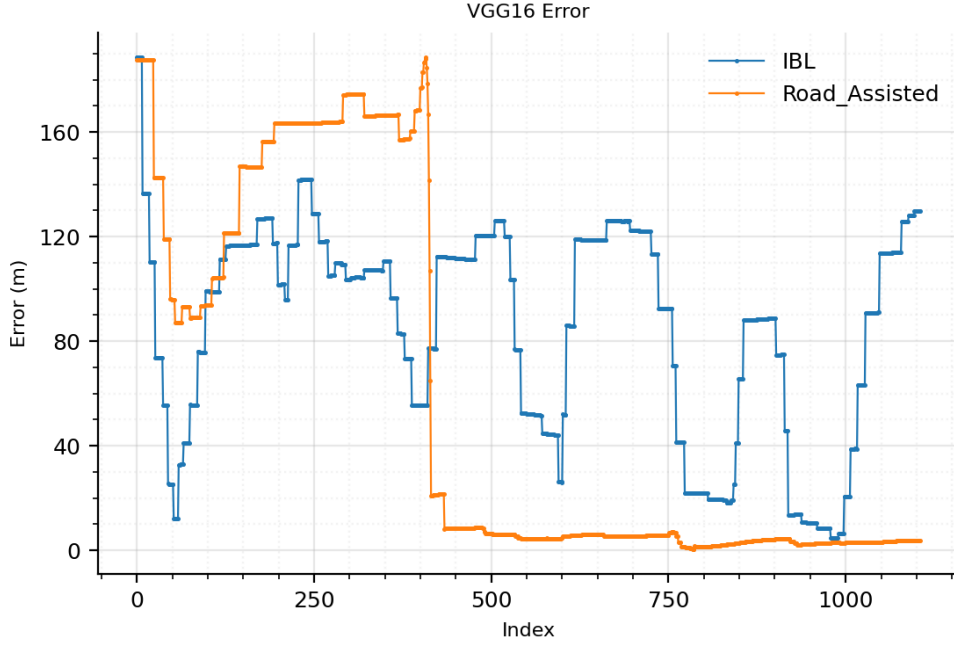


Figure 5.6: Comparison of position error between the pure IBL system and the road-assisted IBL system.

Applying $\gamma > 1$ to the VGG16 architecture enables the system not only to rely on the noisy visual measurements of the network but also to introduce an alternative measurement that allows the Particle Filter to estimate the position more robustly. This allows the system to now converge in the correct location, reducing the error below 10m. This is shown in Figure 5.6 where the error of the road-assisted system converges.

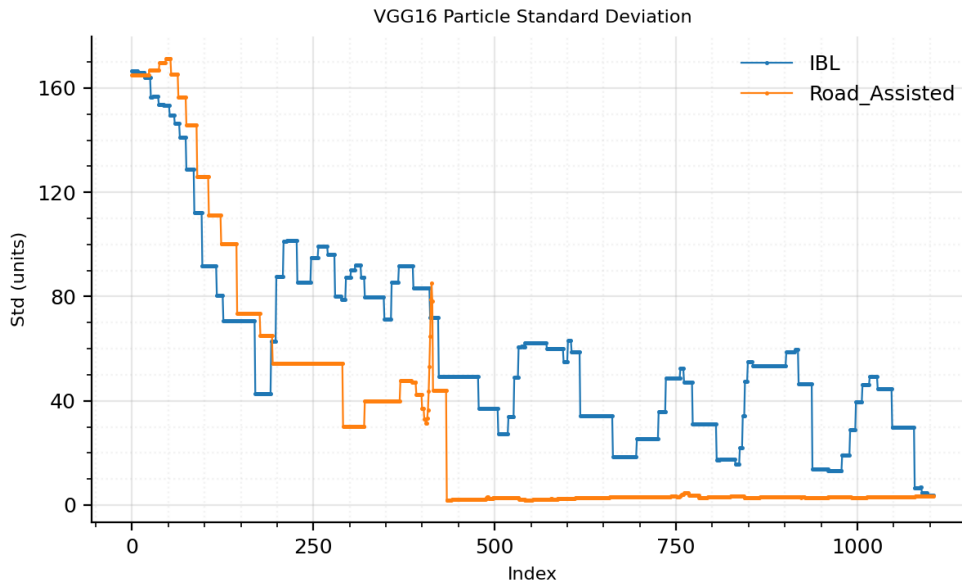


Figure 5.7: Comparison of the standard deviation between the pure IBL system and the road-assisted IBL position.

Figure 5.7 shows the standard deviation history of the particles' behaviour for both approaches tested with the VGG16 architecture. Here, it can be seen that the pure IBL approach struggles to converge due to the multimodal nature of the hidden (p.d.f), due to perceptual aliasing, causing the system to get trapped in a local minima or oscillate between two distinct modes. However, the system successfully converges to the correct location once the road network is integrated. The addition of the road network as a prior allowed the system to disambiguate between the two modes. Results show that incorporating the road network yields strong convergence, with position error reduced to $< 5m$, thereby achieving accurate localisation.

VGG16 paired with the Spatial-Aware Feature Aggregation (SAFA) module [122, 39] and the CVST networks demonstrate higher robustness to perceptual aliasing and compute higher-quality cross-view embeddings, resulting in less noisy measurements. These results align with those presented in Chapter 3, where different neural networks were compared against their recall metrics.

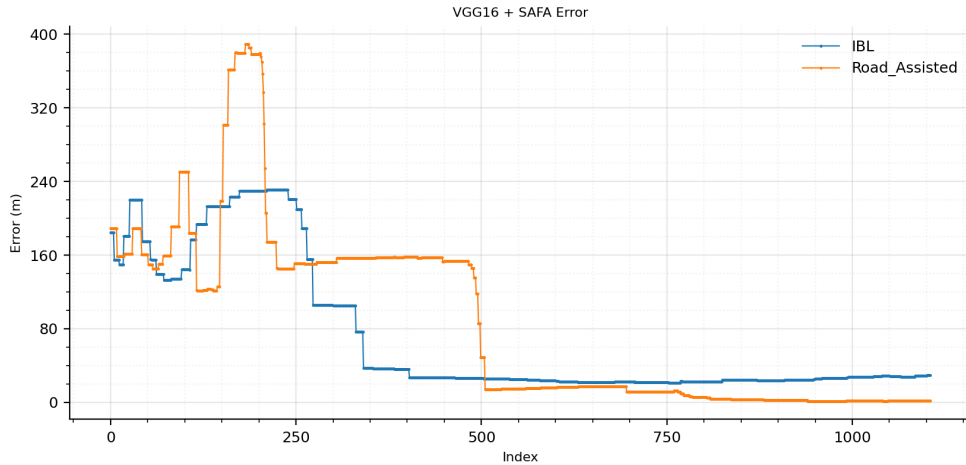


Figure 5.8: Position-error comparison between the pure IBL system and the road-assisted IBL system, for the VGG16+SAFA network.

It can be noted from Fig. 5.8 and Fig. 5.9 that implementing the SAFA module allows the system to converge to a close estimate of the location. However, the estimated error is measured $\sim 30m$ from the correct position. The SAFA module clearly improves the quality of the embeddings, allowing for a converged position. However, when integrating the road network assistance, the error is significantly reduced to less than $5m$ as the vehicle propagates forward.

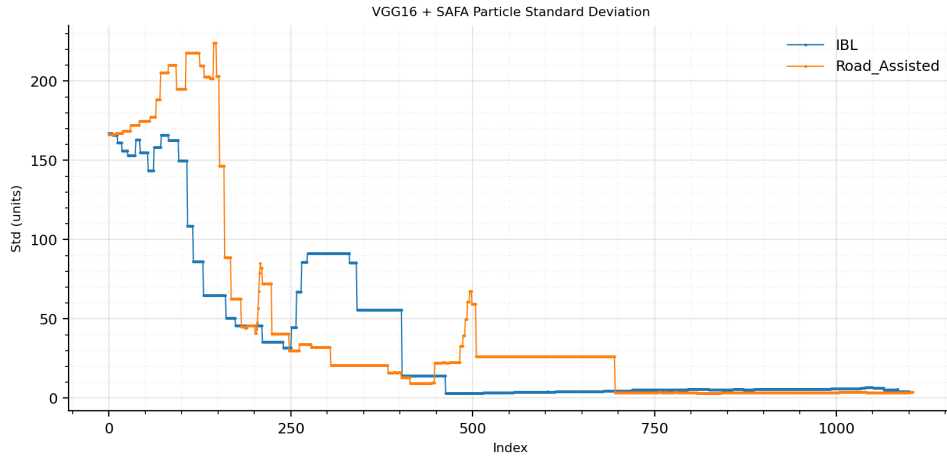


Figure 5.9: Standard-deviation comparison between the pure IBL system and the road-assisted IBL system, for the VGG16+SAFA network.



(a) IBL



(b) Road Assisted

Figure 5.10: Particle Filter converging to the correct position when implemented with VGG16 + SAFA.

Figure 5.10 shows an example of particles clustering around the correct localisation for both cases. Figure 5.11 shows the performance of the CVST neural network as a pure image-based localisation system (shown in blue), and incorporating the road network assistance (shown in orange). It can be noted that including the road network allows the system to converge to the correct location faster. However, this also demonstrates that the pure image-based localisation system with CVST is robust enough to converge to the

correct location.

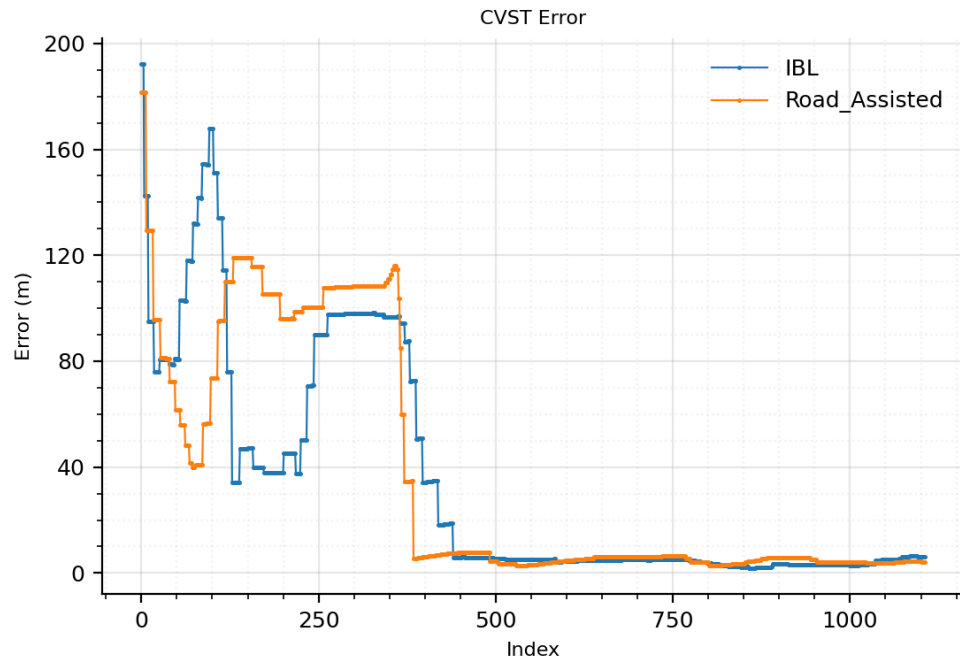


Figure 5.11: Position-error comparison between the pure IBL system and the road-assisted IBL system, for the CVST network.

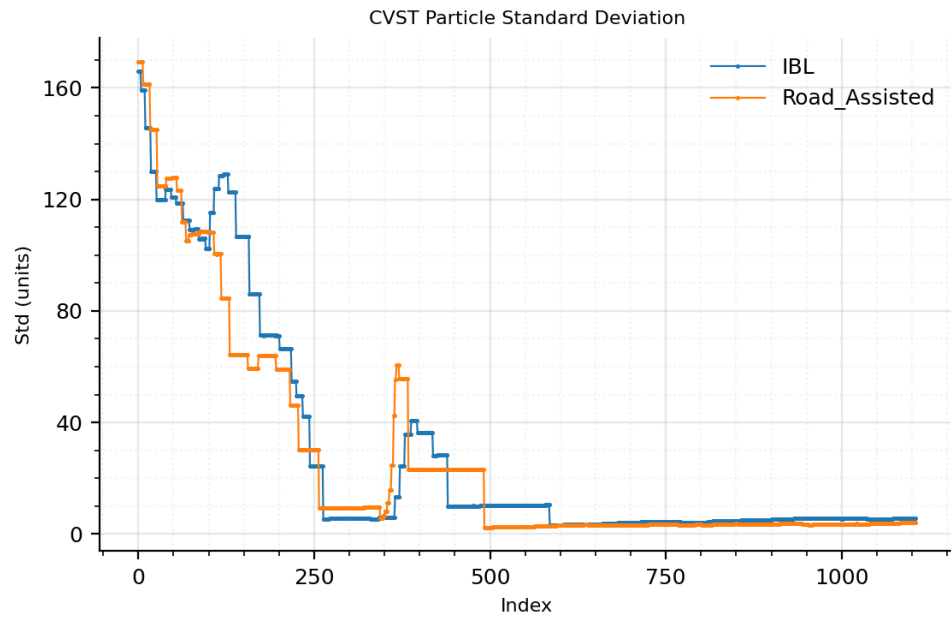


Figure 5.12: Standard-deviation comparison between the pure IBL system and the road-assisted IBL system, for the CVST network.



(a) IBL

(b) Road Assisted

Figure 5.13: Particle Filter converging to the correct position when implemented with CVST. Video of the CVST network applied to a pure IBL system with 1500 particles can be found in the link: https://youtu.be/_0pakpzS_Is. Video of the CVST network applied to a road assisted IBL system with 700 particles can be found in the link: <https://youtu.be/ySoQ5QMKhbI?>

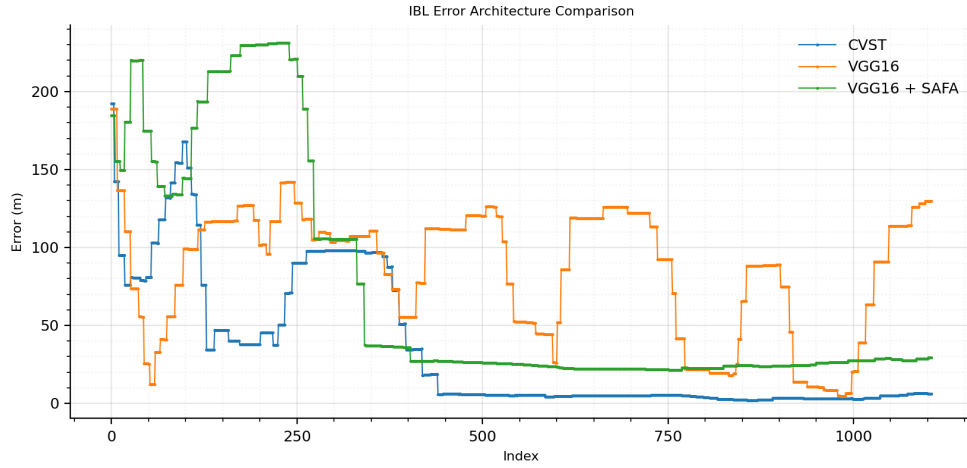


Figure 5.14: Comparison between architectures for the pure IBL system.

The complete system performance is compared against each architecture and find that the accuracy of the system improves when utilising CVST as shown in Figure 5.14.

5.4.2 System Performance

In addition to comparing architectures, tests demonstrate each architecture’s ability to converge while reducing the number of particles. It is found that although the VGG16 with the SAFA module converges, it requires more particles to converge. The minimum number of particles required for the VGG16 with the SAFA module to converge was 1100. While the CVST can operate with 700 particles, allowing the system to be more efficient while keeping image retrieval robust. CVST reduces the number of particles by 36% compared to SAFA. Each system is capable of running one inference in a fraction of a second, making it near-real time. Table 5.1 shows the different timings as a function of particles for the 3D-tree and the map likelihood that includes the encoding and Euclidean distance computation. This is attributed to the integration of the 3D KD-Tree and the particle filter. Without the 3D-tree, inference can take up to 30s with ~ 1000 particles. Our experiments are evaluated on a system with a Ryzen 7 processor, 64 GB of RAM and a 2080Ti GPU.

Performance Comparative Study			
Name	Particles (units)	Tree Search (s)	Map Likelihood (s)
VGG16 RA	1500	0.25	0.107
VGG16 + SAFA $\dagger$ IBL	1500	0.24	0.0107
	1100	0.17	0.0076
VGG16 + SAFA $\dagger$ RA	800	0.125	0.006
CVST $\dagger^o$ IBL	1500	0.2282	0.014
	800	0.122	0.009
CVST $\dagger^o$ RA	300	0.045	0.003

Table 5.1: Near-real time performance of each network tested at different numbers of particles. The symbol $\dagger$ denotes that the model was trained using soft margin loss. o represents networks that have been pretrained with ImageNet weights. (RA) Road Assisted and (IBL) Pure IBL.

From the timings recorded it can be noted that SAFA is capable of operating faster. However, the speed of this model comes at the cost of the number of particles required to converge to the correct pose estimate. To demonstrate this, a study varying the number of particles when $\gamma = 0$ (pure IBL) for VGG16+SAFA and CVST architectures is presented. Findings show CVST can converge to the correct location with 800 particles with a pose

estimate error below 5m. On the contrary, SAFA requires 1100 particles with an error below 50m.

5.4.3 Particle Reduction Study

Figure 5.15 shows that CVST is capable of operating with 800 particles in a pure IBL system, maintaining pose estimate error below 30m. When road assistance is applied, CVST can perform with a minimum of 300 particles, maintaining a pose estimate error below 25m. In pure IBL systems, CVST fails at a minimum of 700 particles; we show a test at 500 particles failing the pose estimate.

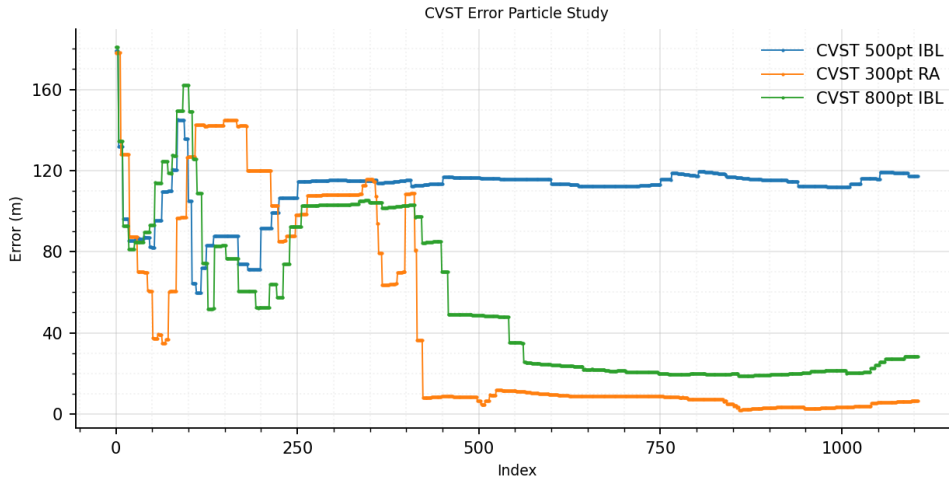


Figure 5.15: Performance of CVST at different numbers of particles.

Figure 5.16 shows that SAFA is capable of operating with 1100 particles in a pure IBL system, maintaining pose estimate error below 30m. When road assistance is applied, SAFA can perform with a minimum of 800 particles, maintaining a pose estimate error below 20m. When 300 particles are used in road assisted IBL systems, SAFA fails to estimate pose, contrary to CVST. In pure IBL systems, SAFA fails at a minimum of 1100 particles. A test with 800 particles in pure IBL application shows failure to estimate pose.

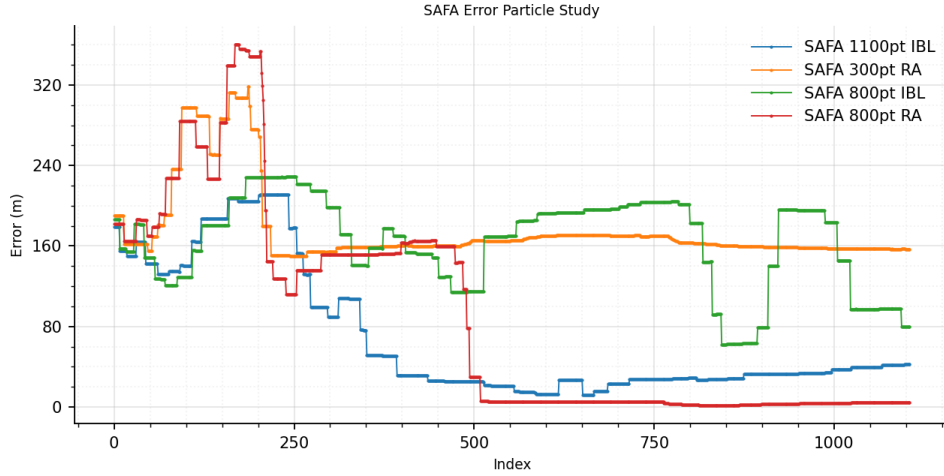


Figure 5.16: Performance of CVST at different numbers of particles.

This study demonstrates the reduction in particles when CVST is applied. Additionally, it also shows how road assistance improves robustness in localisation.

5.4.4 Architecture Error Comparison

Figure 5.17 presents the accuracy of localisation accross the three considered architectures as a function of time (i.e. where index refers to the frame number in the sequence). Here, it can be seen that CVST outperforms the other architectures in terms of localisation time, even when the road assistance is being used. Again, the superiority of CVST aligns with the results presented across the earlier chapters. A video of CVST operating as a pure IBL system is presented in the link: https://youtu.be/_0pakpzS_Is. A video of the CVST network operating with the road assisted likelihood is presented in the link: <https://youtu.be/ySoQ5QMKhbI?>.

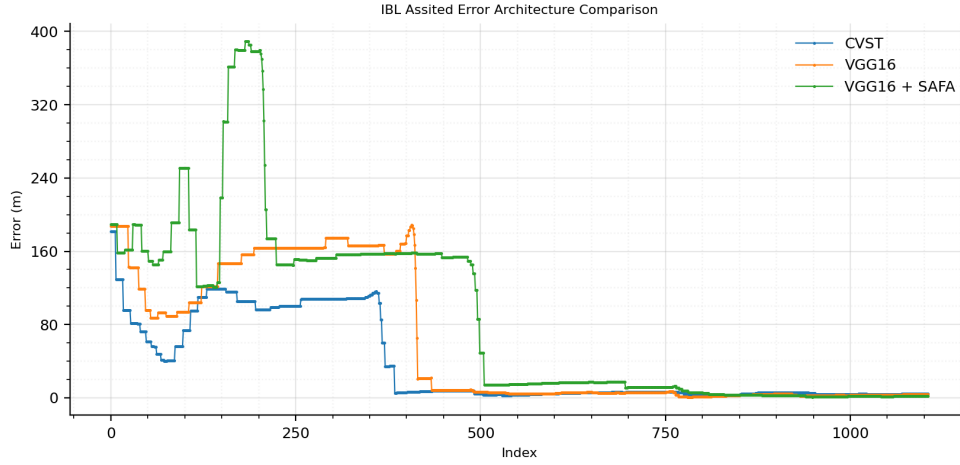


Figure 5.17: Comparison between architectures for the road-assisted IBL system.

5.5 Conclusion

In this chapter, we present an Image-Based Localisation system capable of operating in near-real time with a limited FoV by integrating a particle filter with an efficient data structure (3D KD-Tree). To improve the robustness of the position estimate, a novel particle filter likelihood formulation in the context of IBL is proposed. The likelihood incorporates a weighted approach that combines the probability of the measurement with prior information of the satellite map and the known road network. In addition, we present a novel comparison between different neural network architectures working in conjunction with a particle filter tightly coupled to a 3D-Tree. Finally, we demonstrate the advantages of our novel proposed system that incorporates the use of the known road network, pre-encoded satellite images and the 3D KD-Tree data structure that allows near-real time operation with limited FoV images.

Chapter 6

Conclusion

This thesis explored the development of Image-Based Localisation (IBL) systems for both aerial and ground-based platforms, utilising deep learning methodologies. A novel Cross-View Shared Transformer (CVST) was designed, trained, and rigorously evaluated across aerial and ground vehicle applications. In addition, a near-real time cross-view IBL system was developed which is capable of mitigating perceptual aliasing whilst operating with limited fields of view.

This chapter begins by summarising the research findings, before outlining the principal contributions of the study. It then discusses the limitations encountered and finally proposes directions for future research.

6.1 Summary of Findings

This PhD addresses critical areas of research relating to mobile platform navigation where typical GNSS signal can be affected such as urban canyons or indeed in GPS-denied environments resulting from signal jamming or spoofing. Both aerial and ground robots can enhance localisation robustness by incorporating optical sensor data as an additional information source for position estimation. This approach provides an extra layer of redundancy within navigation systems, contributing to more effective navigation and improved safety—an essential requirement for the deployment and acceptance of autonomous systems within society.

Investigating alternative localisation strategies, therefore, plays a key role in increasing system reliability. By addressing major challenges in IBL, including perceptual aliasing, improving retrieval accuracy, and enabling near-real time performance, this research advances the state of the art and strengthens localisation capabilities for both aerial and ground platforms. Through the examination of Bayesian filtering techniques, deep-learning-based feature extraction, retrieval strategies, and data structures, this thesis develops methods that enhance the robustness of contemporary IBL systems. The main findings of this thesis include;

1. The ability of CVST to improve recall metrics for cross-view image-based localisation using panoramic and limited FoV images.
2. Identifying the recall improvement when sharing weights in siamese networks for image retrieval tasks.
3. Demonstrating the ability of CVST to operate in aerial platforms under urban cross-time datasets and rural datasets.
4. Demonstrating the solution to the kidnapped robot problem via IBL using a database of pre-encoded embeddings that enables near-real time localisation for UAVs.
5. A mathematical formulation that allows dynamic weighting of the particle filter likelihood, incorporating the road network into the estimation process, and resulting in robust localisation despite perceptual aliasing or neural network architecture.
6. The ability of a cross-view IBL system to operate in near-real time with limited FoV images, via the tight coupling between a 3D KD-Tree and a particle filter.
7. The reduction of particles when integrating CVST with the particle filter and efficient data structures.

Taken together, the findings discussed above enable a robust cross-view IBL system capable of operating with limited-FoV ground-level imagery, as well as an aerial IBL framework able to function effectively in challenging rural environments.

6.2 Answering the Research Questions

This section revisits the five research questions (RQ1 - RQ5) posed in Section 1.4 and briefly summarises how each has been addressed.

- **RQ1: What neural network architectures can improve recall for ground-level cross-view IBL?** Chapter 3 introduced the Cross-View Shared Transformer (CVST), a shared-weight transformer that attained the highest recall of all the architectures evaluated on the KITTI and CVUSA benchmarks. The ablation study further showed that weight sharing in Siamese networks and the weighted soft-margin loss are the principal design choices behind this improvement.
- **RQ2: Can neural network architectures be interchanged between cross-view and aerial IBL?** Chapter 4 demonstrated that the same CVST architecture developed for ground-level cross-view localisation transfers to aerial platforms, successfully localising a UAV in both cross-time urban and challenging rural environments.
- **RQ3: What techniques can be implemented to mitigate perceptual aliasing in IBL?** Perceptual aliasing was addressed at two levels: the more discriminative embeddings learned by CVST (Chapter 3) reduce the ambiguity present during retrieval, and the novel particle-filter likelihood that incorporates prior knowledge of the road network (Chapter 5) resolves the perceptual aliasing by constraining the estimate to plausible locations.
- **RQ4: How can particle filters be integrated with deep learning techniques to mitigate visual feature ambiguity while remaining computationally efficient?** Chapter 5 integrated the CVST within a particle filter by pre-encoding the satellite images into database structured as a 3D KD-tree, removing the need for a per-particle neural-network inference and reducing the number of particles required. This tight coupling between the KD-tree and the particle filter enabled the complete cross-view IBL system to operate in near-real time with limited-FoV imagery.
- **RQ5: Can additional estimation features, such as the road network, reduce position uncertainty for cross-view IBL systems?** Chapter 5 showed that

incorporating the road network into the particle-filter likelihood significantly reduces position uncertainty. The road-assisted formulation improves the robustness when estimating the multimodal distributions caused by perceptual aliasing and consistently reduced the localisation error relative to using image similarity alone.

6.3 Contributions

This thesis makes the following contributions to address the research questions outlined in Section 1.4.

1. **Cross View Shared Transformer (CVST):** To enhance recall in cross-view IBL systems, this thesis introduces the CVST and demonstrates its superior performance compared with state-of-the-art architectures. Chapter 3 formulates the IBL task as an image-retrieval problem, in which a query image is geo-localised using a feature embedding extracted by a neural network. The data-collection strategy was described, the deep-learning architectures evaluated, and the training methodology adopted. System performance was assessed using recall metrics for both limited-FoV and panoramic imagery. An extensive ablation study highlighted the advantages of the proposed model for both image types. Additionally, key training insights were presented—specifically, the benefits of weight sharing in Siamese networks—which further improved recall performance (Chapter 3; addresses RQ1 and RQ2).
2. **Aerial IBL in urban and Rural environments:** A new aerial IBL framework was proposed to successfully localise UAVs operating in real-world conditions. This integration enabled near-real time estimation of platform location likelihood and the performance of the CVST on a real rural dataset was evaluated. Within this research, the CVST was trained on UAV data and used to compute similarity measures between UAV feed and satellite images under different environments. While techniques such as Variational Autoencoders (VAEs) and Convolutional Neural Networks (CNNs) have previously been applied to aerial localisation and “return-to-home” tasks [15], Chapter 4 extended the application of CVST to UAV platforms. This chapter demonstrated the localisation of a UAV in a cross-time urban dataset and also within a challenging

rural sequence with varying illumination. Using a triplet-loss function combined with cosine similarity, the CVST was trained on both environments to enable reliable recall across aerial viewpoints (Chapter 4; addresses RQ2 and RQ3).

3. **Robust near-real time cross-view IBL:** To address the perceptual aliasing challenge common to IBL systems, this thesis introduced a novel mathematical framework that fused image-similarity metrics with prior knowledge of the road network, resulting in a robust UGV localisation system. The proposed likelihood formulation incorporated road-network constraints, mitigating perceptual aliasing noise and reducing system convergence time. Chapter 5 presented a near-real time implementation, employing a particle filter that leverages the pre-encoded databased structured in a 3D KD-tree, making the system capable of modelling the non-linear spatial distribution present in the environment estimating the position of the mobile platform (Chapter 5; addresses RQ3, RQ4 and RQ5).

6.4 Limitations of IBL Methodology

Although this research delivers substantial improvements in IBL capability, several limitations remain:

1. Aerial localisation must currently be performed offline, as weight and power limitations prevent the integration of the hardware required for real-time on-board computation.
2. While the cross-view IBL system demonstrates robustness to perceptual aliasing, under extreme visual noise the platform may still converge to an incorrect location. Such conditions may arise from sudden illumination changes or seasonal variations that are not represented in the training data.
3. The susceptibility of IBL to perceptual aliasing restricts reliable operation to relatively small areas, typically only a few square kilometres. For city-scale localisation, significantly longer trajectories are required to ensure convergence.
4. The widespread availability of public datasets raises concerns regarding data privacy. Furthermore, the potential misuse of IBL technology for surveillance or military

applications presents ethical challenges that the research community must continue to address.

6.5 Future Work

Future research can build upon the contributions of this thesis in several aspects:

1. Further experimentation in rural settings, where perceptual aliasing is more pronounced, would help validate and strengthen cross-view IBL performance.
2. The development of extensive datasets containing synchronised, geo-referenced aerial and ground-level imagery would support more robust training, particularly when incorporating seasonal and illumination variations.
3. Investigating the miniaturisation and optimisation of deep-learning models specifically tailored to cross-view localisation with limited-FoV imagery would facilitate deployment on resource-constrained platforms.
4. Continued experimentation and real-world deployment of IBL systems on autonomous robots is essential to bridge the gap between research and practical implementation.

Additional research in these areas will enable large-scale deployment of IBL in autonomous vehicles.

6.6 Conclusion

Image-Based Localisation (IBL) relies on a catalogue of images that act as a visual map of landmarks, which a mobile platform uses to determine its global position. In aerial IBL, localisation is performed by comparing aerial imagery with geo-referenced satellite-derived raster maps. Since the viewpoint between aerial platforms and satellite imagery is typically aligned i.e. planimetric views, the matching process is comparatively straightforward, even in the presence of perceptual aliasing.

In contrast, cross-view IBL is considerably more difficult due to the extreme viewpoint differences between ground-level imagery and satellite references. Previous research has mitigated this challenge through the use of panoramic ground-level images, achieving high recall and improved localisation accuracy, as shown in Chapter 3. However, panoramic cameras are not universally available, and stitching individual images to form panoramas introduces an additional processing stage. As a result, cross-view localisation with limited-field-of-view (FoV) cameras has remained a significant challenge in recent years. While recent studies have shown that localisation with limited-FoV images is feasible, limitations remain. As an example, Kim *et al.* [67] did not achieve near-real time performance, and Downes *et al.* [39] required long trajectories to successfully localise. Overall, the literature demonstrates relatively limited exploration of IBL with restricted FoV imagery due to the complexity it introduces. Furthermore, computational speed remains a fundamental challenge for practical deployment.

Bibliography

- [1] Lost in quantization: Improving particular object retrieval in large scale image databases. pages 1–8. IEEE, 6 2008.
- [2] Spatial Pyramid-Enhanced NetVLAD With Weighted Triplet Loss for Place Recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 31:661–674, 2 2020.
- [3] KITTI-360: A Novel Dataset and Benchmarks for Urban Scene Understanding in 2D and 3D. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 3 2022.
- [4] 3d_molier International / PixelSquid. “Underwater Robot BlueROV2 PNG PSD” [image]. <https://www.pixelsquid.com/png/underwater-robot-bluerov2-2464589410423804951?image=H02>, 2025. Royal-free image from PixelSquid; accessed 31 Oct 2025.
- [5] Amar Ali-Bey, Brahim Chaib-Draa, and Philippe Giguere. MixVPR: Feature Mixing for Visual Place Recognition. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2997–3006. IEEE, 1 2023.
- [6] Sergiu Nedevschi Andra Petrovai, Vlad Miclea. Depth-Aware Panoptic Segmentation with Mask Transformers and Panoptic Bins for Autonomous Driving. In *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2024.
- [7] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:1437–1451, 6 2018.

- [8] Relja Arandjelovic and Andrew Zisserman. All about VLAD. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1578–1585, 2013.
- [9] Eduardo A Avila H, Aidan Magee, Tim McCarthy, and John McDonald. Aerial image based localisation via learned embeddings. In *IET Conference Proceedings CP887*, volume 2024, pages 291–298. IET, 2024.
- [10] Eduardo A Avila H, Tim McCarthy, and John McDonald. Cross View Shared Transformer (CVST) a Shared Weight Siamese Network for Limited FoV Image Based Localisation. In *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1261–1268. IEEE, 2024.
- [11] Udana Bandara, Mikio Hasegawa, Masugi Inoue, Hiroyuki Morikawa, and Tomonori Aoyama. Design and implementation of a bluetooth signal strength based location sensing system. In *Proceedings. 2004 IEEE Radio and Wireless Conference (IEEE Cat. No. 04TH8746)*, pages 319–322. IEEE, 2004.
- [12] Mayank Bansal, Harpreet S. Sawhney, Hui Cheng, and Kostas Daniilidis. Geo-localization of street views with aerial image databases. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1125–1128. ACM, 11 2011.
- [13] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110:346–359, 6 2008.
- [14] Gabriele Berton, Carlo Masone, and Barbara Caputo. Rethinking Visual Geo-localization for Large-Scale Applications. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4868–4878. IEEE, 6 2022.
- [15] Mollie Bianchi and Timothy D. Barfoot. UAV localization using autoencoded satellite images. *IEEE Robotics and Automation Letters*, 6:1761–1768, 4 2021.
- [16] Andrea Borgese, Dario C. Guastella, Giuseppe Sutera, and Giovanni Muscato. Tether-Based Localization for Cooperative Ground and Aerial Vehicles. *IEEE Robotics and Automation Letters*, 7:8162–8169, 7 2022.

- [17] Boston Dynamics, Inc. “Spot Explorer – web image” [image]. <https://bostondynamics.com/wp-content/uploads/2023/05/spot-explorer-web-2-1536x864.jpg>, 2023. Image published May 2023 on Boston Dynamics website, accessed 31 Oct 2025.
- [18] Leana Bouse, Scott A King, and Tianxing Chu. Simplified Indoor Localization Using Bluetooth Beacons and Received Signal Strength Fingerprinting with Smartwatch. *Sensors*, 24(7):2088, 2024.
- [19] Eli Brosh, Matan Friedmann, Ilan Kadar, Lev Yitzhak Lavy, Elad Levi, Shmuel Rippa, Yair Lempert, Bruno Fernandez-Ruiz, Roei Herzig, and Trevor Darrell. Accurate visual localization for automotive applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.
- [20] Andrew Brown, Weidi Xie, Vicky Kalogeiton, and Andrew Zisserman. Smooth-AP: Smoothing the Path Towards Large-Scale Image Retrieval. In *European Conference on Computer Vision (ECCV)*, pages 677–694, 7 2020.
- [21] F. Caballero, L. Merino, J. Ferruz, and A. Ollero. Vision-Based Odometry and SLAM for Medium and High Altitude Flying UAVs. *Journal of Intelligent and Robotic Systems*, 54:137–161, 3 2009.
- [22] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. BRIEF: Binary Robust Independent Elementary Features. In *European Conference on Computer Vision*, pages 778–792, 2010.
- [23] Carlos Campos, Richard Elvira, Juan J. Gómez Rodríguez, José M. M. Montiel, and Juan D. Tardós. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial, and Multimap SLAM. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021.
- [24] Bingyi Cao, Andre Araujo, and Jack Sim. Unifying Deep Local and Global Features for Image Search. 1 2020.

- [25] Francesco Castaldo, Amir Zamir, Roland Angst, Francesco Palmieri, and Silvio Savarese. Semantic Cross-View Matching. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, pages 9–17, 10 2015.
- [26] Yingxiu Chang, Yongqiang Cheng, Umar Manzoor, and John Murray. A review of UAV autonomous navigation in GPS-denied environments. *Robotics and Autonomous Systems*, page 104533, 2023.
- [27] David M. Chen, Georges Baatz, Kevin Koser, Sam S. Tsai, Ramakrishna Vedantham, Timo Pylvanainen, Kimmo Roimela, Xin Chen, Jeff Bach, Marc Pollefeys, Bernd Girod, and Radek Grzeszczuk. City-scale landmark identification on mobile devices. pages 737–744. IEEE, 6 2011.
- [28] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning, PMLR 119*, pages 1597–1607, 2020.
- [29] Zhe Chen et al. Bayesian filtering: From Kalman filters to particle filters, and beyond. *Statistics*, 182(1):1–69, 2003.
- [30] Jae-Young Choi and Sung-Gaun Kim. Collaborative Tracking Control of UAV-UGV. *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, 6:2487–2490, 2012.
- [31] Jae-Young Choi and Sung-Gaun Kim. Study on the Improved Target Tracking for the Collaborative Control of the UAV-UGV. *Journal of Institute of Control, Robotics and Systems*, 19:450–456, 5 2013.
- [32] Ming Dai, Jinglin Huang, Jiedong Zhuang, Wenbo Lan, Yongheng Cai, and Enhui Zheng. Vision-Based UAV Localization System in Denial Environments. 1 2022.
- [33] Fabian Deuser, Konrad Habel, and Norbert Oswald. Sample4geo: Hard negative sampling for cross-view geo-localisation. In *Proc. of the IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pages 16847–16856, 2023.

- [34] Fabian Deuser, Wejdene Mansour, Hao Li, Konrad Habel, Martin Werner, and Norbert Oswald. Temporal resilience in geo-localization: adapting to the continuous evolution of urban and rural environments. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pages 479–488, 2025.
- [35] Fabian Deuser, Martin Werner, Konrad Habel, and Norbert Oswald. Optimizing geo-localization with k-means re-ranking in challenging weather conditions. In *Proceedings of the 2nd Workshop on UAVs in Multimedia: Capturing the World from a New Perspective*, pages 9–13, 2024.
- [36] Lirong Ding, Ji Zhou, Lingxuan Meng, and Zhiyong Long. A Practical Cross-View Image Matching Method between UAV and Satellite for UAV-Based Geo-Localization. *Remote Sensing*, 13:47, 12 2020.
- [37] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. 10 2020.
- [38] Lena M Downes, Dong-Ki Kim, Ted J Steiner, and Jonathan P How. City-wide street-to-satellite image geolocalization of a mobile ground agent. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11102–11108. IEEE, 2022.
- [39] Lena M Downes, Ted J Steiner, Rebecca L Russell, and Jonathan P How. Wide-area geolocalization with a limited field of view camera in challenging urban environments. *arXiv preprint arXiv:2308.07432*, 2023.
- [40] Jos Elfring, Elena Torta, and René Van De Molengraft. Particle filters: A hands-on tutorial. *Sensors*, 21(2):438, 2021.
- [41] E. Foxlin. Inertial head-tracker sensor fusion by a complementary separate-bias Kalman filter. In *Proceedings of the IEEE 1996 Virtual Reality Annual International Symposium*, pages 185–194, 1996.

- [42] Kenjiro Fujii, Wei Wang, Yuto Kaneko, Yoshihiro Sakamoto, Hiroaki Arie, and Shigeki Sugano. Accurate indoor positioning using IMES radio. In *2015 IEEE International Conference on Mechatronics and Automation (ICMA)*, pages 1966–1971. IEEE, 2015.
- [43] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research (IJRR)*, 2013.
- [44] Anand George, Niko Koivumäki, Teemu Hakala, Juha Suomalainen, and Eija Honkavaara. Visual-Inertial Odometry Using High Flying Altitude Drone Datasets. *Drones*, 7:36, 1 2023.
- [45] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.
- [46] Brian Ferris Dirk Hähnelt and Dieter Fox. Gaussian processes for signal strength-based location estimation. In *Proceeding of robotics: science and systems*. Citeseer, 2006.
- [47] Takaya Hakukawa, Kenji Uchiyama, and Kai Masuda. Cooperative System with UAV and UGV for Disaster Area Exploration. pages 117–122. IEEE, 10 2018.
- [48] C. Harris and M. Stephens. A Combined Corner and Edge Detector. pages 23.1–23.6. Alvey Vision Club, 1988.
- [49] Stephen Hausler, Sourav Garg, Ming Xu, Michael Milford, and Tobias Fischer. Patch-NetVLAD: Multi-Scale Fusion of Locally-Global Descriptors for Place Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14141–14152, 6 2021.
- [50] James Hays and Alexei A. Efros. IM2GPS: estimating geographic information from a single image. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 6 2008.

- [51] Xianxu Hou, Linlin Shen, Ke Sun, and Guoping Qiu. Deep feature consistent variational autoencoder. pages 1133–1141. IEEE, 3 2017.
- [52] Xianxu Hou, Linlin Shen, Ke Sun, and Guoping Qiu. Deep feature consistent variational autoencoder. In *2017 IEEE winter conference on applications of computer vision (WACV)*, pages 1133–1141. IEEE, 2017.
- [53] Di Hu, Xia Yuan, Huiying Xi, Jie Li, Zhenbo Song, Fengchao Xiong, Kai Zhang, and Chunxia Zhao. Road Structure Inspired UGV-Satellite Cross-View Geo-Localization. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2024.
- [54] Sixing Hu, Mengdan Feng, Rang M. H. Nguyen, and Gim Hee Lee. CVM-Net: Cross-View Matching Network for Image-Based Ground-to-Aerial Geo-Localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7258–7267. IEEE, 6 2018.
- [55] Sixing Hu and Gim Hee Lee. Image-Based Geo-Localization Using Satellite Imagery. *International Journal of Computer Vision*, 128:1205–1219, 5 2020.
- [56] Yongchang Hu and Geert Leus. Robust differential received signal strength-based localization. *IEEE Transactions on Signal Processing*, 65(12):3261–3276, 2017.
- [57] Yuekun Hu, Yingfan Liu, and Bin Hui. Combining OpenStreetMap with Satellite Imagery to Enhance Cross-View Geo-Localization. *Sensors*, 25(1):44, 2024.
- [58] Martin Humenberger, Yohann Cabon, Nicolas Guerin, Julien Morat, Vincent Leroy, Jérôme Revaud, Philippe Rerole, Noé Pion, Cesar de Souza, and Gabriela Csurka. Robust Image Retrieval-based Visual Localization using Kapture. 7 2020.
- [59] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-To-Image Translation With Conditional Adversarial Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [60] Aman Jain, Milind Mahajan, and Radha Saraf. Standardization of the shape of ground control point (GCP) and the methodology for its detection in images for

- UAV-based mapping applications. In *Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC), Volume 1 1*, pages 459–476. Springer, 2020.
- [61] Dushyant Singh Jasrotia and Manisha J. Nene. Localisation using UAV in RFID and Sensor Network Environment: Needs and Challenges. In *2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, pages 274–279. IEEE, 10 2019.
- [62] John R Jensen and Jungho Im. Remote sensing change detection in urban environments. *Geo-spatial Technologies in Urban Environments: Policy, Practice, and Pixels*, pages 7–31, 2007.
- [63] Edward Johns and Guang-Zhong Yang. Dynamic scene models for incremental, long-term, appearance-based localisation. In *2013 IEEE International Conference on Robotics and Automation*, pages 2731–2736, 2013.
- [64] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3304–3311, 2010.
- [65] Ebrahim Karami, Siva Prasad, and Mohamed Shehata. Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images. 11 2015.
- [66] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in Vision: A Survey. *ACM Computing Surveys*, 54:1–41, 1 2022.
- [67] Dong Ki Kim and Matthew R. Walter. Satellite image-based localization via learned embeddings. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 2073–2080, 2017.
- [68] AS Krishnakumar and Parameshwaran Krishnan. The theory and practice of signal strength-based location estimation. In *2005 International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 10–pp. IEEE, 2005.

- [69] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.
- [70] Marko Krizmancic, Barbara Arbanas, Tamara Petrovic, Frano Petric, and Stjepan Bogdan. Cooperative Aerial-Ground Multi-Robot System for Automated Construction Tasks. *IEEE Robotics and Automation Letters*, 5:798–805, 4 2020.
- [71] Zongdian Li Tao Yu Yonghui Li Kui Wang, Changyang She and Kei Sakaguchi. Roadside Units Assisted Localized Automated Vehicle Maneuvering: An Offline Reinforcement Learning Approach. In *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2024.
- [72] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.
- [73] Alex Lee, Wonho Song, Byeongho Yu, Duckyu Choi, Christian Tirtawardhana, and Hyun Myung. Survey of robotics technologies for civil infrastructure inspection. *Journal of Infrastructure Intelligence and Resilience*, 2(1):100018, 2023.
- [74] Jae-Keun Lee, Hahmin Jung, Huosheng Hu, and Dong Hun Kim. Collaborative control of UAV/UGV. pages 641–645. IEEE, 11 2014.
- [75] Minhyeok Lee. Mathematical analysis and performance evaluation of the gelu activation function in deep learning. *Journal of Mathematics*, 2023(1):4229924, 2023.
- [76] María Leyva-Vallina, Nicola Strisciuglio, and Nicolai Petkov. Generalized Contrastive Optimization of Siamese Networks for Place Recognition. 3 2021.
- [77] Hao Li, Fabian Deuser, Wenping Yin, Xuanshu Luo, Paul Walther, Gengchen Mai, Wei Huang, and Martin Werner. Cross-view geolocalization and disaster mapping with street-view and VHR satellite imagery: A case study of Hurricane IAN. *ISPRS Journal of Photogrammetry and Remote Sensing*, 220:841–854, 2025.

- [78] Yanyu Li, Geng Yuan, Yang Wen, Ju Hu, Georgios Evangelidis, Sergey Tulyakov, Yanzhi Wang, and Jian Ren. EfficientFormer: Vision Transformers at MobileNet Speed. 6 2022.
- [79] Jinliang Lin, Zhedong Zheng, Zhun Zhong, Zhiming Luo, Shaozi Li, Yi Yang, and Nicu Sebe. Joint Representation Learning and Keypoint Detection for Cross-View Geo-Localization. In *IEEE Transactions on Image Processing*, volume 31, pages 3780–3792. Institute of Electrical and Electronics Engineers Inc., 2022.
- [80] Tsung-Yi Lin, Serge Belongie, and James Hays. Cross-View Image Geolocalization. In *2013 IEEE CVPR*, pages 891–898. IEEE, 6 2013.
- [81] Tsung-Yi Lin, Yin Cui, Serge Belongie, and James Hays. Learning deep representations for ground-to-aerial geolocalization. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5007–5015. IEEE, 6 2015.
- [82] Liu Liu and Hongdong Li. Lending Orientation to Neural Networks for Cross-view Geo-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5624–5633, June 2019.
- [83] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022, 2021.
- [84] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.
- [85] David G Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.
- [86] D.G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 1150–1157 vol.2. IEEE, 1999.

- [87] Jiebo Luo, Jie Yu, Dhiraj Joshi, and Wei Hao. Event recognition. In *Proceedings of the 16th ACM international conference on Multimedia*, pages 1071–1080. ACM, 10 2008.
- [88] Will Maddern, Geoff Pascoe, Chris Linegar, and Paul Newman. 1 Year, 1000km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research (IJRR)*, 36(1):3–15, 2017.
- [89] Will Maddern, Geoffrey Pascoe, Matthew Gadd, Dan Barnes, Brian Yeomans, and Paul Newman. Real-time Kinematic Ground Truth for the Oxford RobotCar Dataset. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10697–10704, 2 2020.
- [90] András L Majdik, Charles Till, and Davide Scaramuzza. The Zurich urban micro aerial vehicle dataset. *The International Journal of Robotics Research*, 36:269–273, 3 2017.
- [91] Nathan Michael, Daniel Mellinger, Quentin Lindsey, and Vijay Kumar. The grasp multiple micro-uav testbed. *IEEE Robotics & Automation Magazine*, 17(3):56–65, 2010.
- [92] Takahiro Miki, Petr Khrapchenkov, and Koichi Hori. UAV/UGV Autonomous Cooperation: UAV assists UGV to climb a cliff by attaching a tether. pages 8041–8047. IEEE, 5 2019.
- [93] Krystian Mikolajczyk and Cordelia Schmid. Scale and Affine Invariant Interest Point Detectors. *International Journal of Computer Vision*, 60:63–86, 10 2004.
- [94] Ian D. Miller, Fernando Cladera, Trey Smith, Camillo Jose Taylor, and Vijay Kumar. Stronger Together: Air-Ground Robotic Collaboration Using Semantics. *IEEE Robotics and Automation Letters*, 7:9643–9650, 10 2022.
- [95] Ian D. Miller, Anthony Cowley, Ravi Konkimalla, Shreyas S. Shivakumar, Ty Nguyen, Trey Smith, Camillo Jose Taylor, and Vijay Kumar. Any Way You Look at It: Semantic Crossview Localization and Mapping with LiDAR. In *IEEE Robotics and*

Automation Letters, volume 6, pages 2397–2404. Institute of Electrical and Electronics Engineers Inc., 4 2021.

- [96] Raúl Mur-Artal, J. M. M. Montiel, and Juan D. Tardós. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.
- [97] NASA. Perseverance Rover. <https://mars.nasa.gov/mars2020/>, 2020.
- [98] NASA/JPL-Caltech. “Mars Helicopter Ingenuity (transparent background)” [image]. [https://commons.wikimedia.org/wiki/File:Mars_Helicopter_Ingenuity_\(transparent_background\).png](https://commons.wikimedia.org/wiki/File:Mars_Helicopter_Ingenuity_(transparent_background).png), 2020. Flight model of the Mars 2020 Ingenuity Helicopter; public domain (U.S. Government work). Accessed 31 Oct 2025.
- [99] Archana Nath, Ajit Kumar Behura, Kanak Tara, and Ritesh Kumar. Global Journal Of Engineering Science And Researches, 2019.
- [100] Aleix Paris, Brett T. Lopez, and Jonathan P. How. Dynamic Landing of an Autonomous Quadrotor on a Moving Platform in Turbulent Wind Conditions. pages 9577–9583. IEEE, 5 2020.
- [101] Advait Parulekar, Liam Collins, Karthikeyan Shanmugam, Aryan Mokhtari, and Sanjay Shakkottai. InfoNCE loss provably learns cluster-preserving representations. In *The Thirty Sixth Annual Conf. on Learning Theory*, pages 1914–1961. PMLR, 2023.
- [102] Bhavit Patel, Timothy D. Barfoot, and Angela P. Schoellig. Visual Localization with Google Earth Images for Robust Global Pose Estimation of UAVs. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6491–6497. IEEE, 5 2020.
- [103] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. pages 1–8. IEEE, 6 2007.

- [104] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning CNN image retrieval with no human annotation. *IEEE transactions on pattern analysis and machine intelligence*, 41(7):1655–1668, 2018.
- [105] Krishna Regmi and Ali Borji. Cross-View Image Synthesis using Conditional GANs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3501–3510, 3 2018.
- [106] Krishna Regmi and Mubarak Shah. Bridging the Domain Gap for Ground-to-Aerial Image Matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 470–479, 10 2019.
- [107] Rich RobotsGuide.com. “Boston Dynamics’ Atlas robot, a grey and silver bipedal robot . . .” (photograph). <https://robotsguide.com/robots/atlas>, 2025. Photo: Boston Dynamics, accessed 31 Oct 2025.
- [108] Adrian Rosebrock. Detecting ArUco markers with OpenCV and Python - PyImageSearch, 12 2020.
- [109] Edward Rosten and Tom Drummond. *Machine Learning for High-Speed Corner Detection*. 2006.
- [110] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: An efficient alternative to SIFT or SURF. pages 2564–2571. IEEE, 11 2011.
- [111] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From Coarse to Fine: Robust Hierarchical Localization at Large Scale. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12708–12717. IEEE, 6 2019.
- [112] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning Feature Matching With Graph Neural Networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4937–4946. IEEE, 6 2020.

- [113] Torsten Sattler, Michal Havlena, Konrad Schindler, and Marc Pollefeys. Large-Scale Location Recognition and the Geometric Burstiness Problem. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1582–1590. IEEE, 6 2016.
- [114] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Fast image-based localization using direct 2D-to-3D matching. In *2011 International Conference on Computer Vision*, pages 667–674. IEEE, 11 2011.
- [115] Olivier Saurer, Georges Baatz, Kevin Köser, L’ubor Ladický, and Marc Pollefeys. Image Based Geo-localization in the Alps. *International Journal of Computer Vision*, 116:213–225, 2 2016.
- [116] Davide Scaramuzza and Zichao Zhang. Visual-inertial odometry of aerial robots. *arXiv preprint arXiv:1906.03289*, 2019.
- [117] Patrik Schmuck and Margarita Chli. Multi-UAV collaborative monocular SLAM. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3863–3870. IEEE, 5 2017.
- [118] Johannes L Schönberger and Jan-Michael Frahm. Structure-from-Motion Revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 6 2016.
- [119] Turgay Senlet and Ahmed Elgammal. Satellite image based precise robot localization on sidewalks. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 2647–2653, 2012.
- [120] Zachary Seymour, Karan Sikka, Han-Pang Chiu, Supun Samarasekera, and Rakesh Kumar. Semantically-Aware Attentive Neural Embeddings for Image-based Visual Localization. 12 2018.
- [121] Tixiao Shan and Brendan Englot. LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4758–4765, 2018.

- [122] Yujiao Shi, Liu Liu, Xin Yu, and Hongdong Li. Spatial-Aware Feature Aggregation for Cross-View Image based Geo-Localization. In *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, volume 32, pages 10090–10100. Curran Associates, Inc., 12 2019.
- [123] Yujiao Shi, Xin Yu, Dylan Campbell, and Hongdong Li. Where am I looking at? Joint Location and Orientation Estimation by Cross-View Matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4064–4072, 5 2020.
- [124] Yujiao Shi, Xin Yu, Liu Liu, Tong Zhang, and Hongdong Li. Optimal Feature Transport for Cross-View Image Geo-Localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11990–11997, 4 2020.
- [125] Yujiao Shi, Xin Yu, Shan Wang, and Hongdong Li. Cvlnet: Cross-view semantic correspondence learning for video-based camera localization. In *Asian Conference on Computer Vision*, pages 123–141. Springer, 2022.
- [126] Baigen Cai Jian Wang Debiao Lu Siqi Wang, Jiang Liu. GNSS Spoofing Detection and Elimination for Resilient Train Positioning using Spiking Neural Network and Compressed Sensing. In *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2024.
- [127] Sivic and Zisserman. Video Google: a text retrieval approach to object matching in videos. In *Proceedings Ninth IEEE International Conference on Computer Vision*, volume 2, pages 1470–1477 vol.2. IEEE, 2003.
- [128] Sindre Skrede. Nordlandsbanen: minute by minute, season by season, 1 2013.
- [129] Maarten Speekenbrink. A tutorial on particle filters. *Journal of Mathematical Psychology*, 73:140–152, 2016.
- [130] Bin Sun, Chen Chen, Yingying Zhu, and Jianmin Jiang. GEOCAPSNET: Ground to Aerial View Image Geo-Localization using Capsule Network. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 742–747, 2019.

- [131] Niko Sünderhauf, Sareh Shirazi, Feras Dayoub, Ben Upcroft, and Michael Milford. On the performance of ConvNet features for place recognition. *IEEE International Conference on Intelligent Robots and Systems*, 2015-December:4297–4304, 2015.
- [132] S.L. The Construct Sim. “SUMMIT-XL robot simulation” (image) – Robotnik SUMMIT-XL mobile robot model. <https://www.theconstruct.ai/summit-xl-robot-simulation-2/>, 2025. Accessed 31 October 2025.
- [133] Sebastian Thrun. Particle Filters in Robotics. In *UAI*, volume 2, pages 511–518. Citeseer, 2002.
- [134] Xiaoyang Tian, Jie Shao, Deqiang Ouyang, and Heng Tao Shen. UAV-Satellite View Synthesis for Cross-View Geo-Localization. *IEEE Transactions on Circuits and Systems for Video Technology*, 32:4804–4815, 7 2022.
- [135] Yuxin Tian, Xueqing Deng, Yi Zhu, and Shawn Newsam. Cross-Time and Orientation-Invariant Overhead Image Geolocalization Using Deep Local Features. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2501–2509. IEEE, 3 2020.
- [136] Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. MLP-Mixer: An all-MLP Architecture for Vision. In *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, 2021.
- [137] Akihiko Torii, Relja Arandjelovic, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 Place Recognition by View Synthesis. volume 40, pages 257–271, 2 2015.
- [138] Akihiko Torii, Josef Sivic, Toma Pajdla, and Masatoshi Okutomi. Visual Place Recognition with Repetitive Structures. pages 883–890. IEEE, 6 2013.
- [139] Akihiko Torii, Josef Sivic, and Tomas Pajdla. Visual localization by linear combination of image descriptors. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 102–109. IEEE, 11 2011.

- [140] Fernando Vanegas, Duncan Campbell, Nicholas Roy, Kevin J. Gaston, and Felipe Gonzalez. UAV tracking and following a ground target under motion and localisation uncertainty. In *2017 IEEE Aerospace Conference*, pages 1–10. IEEE, 3 2017.
- [141] Antoni Rosinol Vidal, Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza. Ultimate SLAM? Combining Events, Images, and IMU for Robust Visual SLAM in HDR and High-Speed Scenarios. *IEEE Robotics and Automation Letters*, 3:994–1001, 4 2018.
- [142] Nam N. Vo and James Hays. Localizing and Orienting Street Views Using Overhead Imagery. In *European Conference on Computer Vision (ECCV)*, pages 494–509, 9 2016.
- [143] Shruti Vyas, Chen Chen, and Mubarak Shah. GAMA: Cross-View Video Geo-Localization. In *European Conference on Computer Vision (ECCV)*, pages 440–456, 7 2022.
- [144] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. CosFace: Large Margin Cosine Loss for Deep Face Recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5265–5274. IEEE, 6 2018.
- [145] Shan Wang, Yanhao Zhang, Ankit Vora, Akhil Perincherry, and Hongdong Li. Satellite image based cross-view localization for autonomous vehicle. *arXiv preprint arXiv:2207.13506*, 2022.
- [146] Tingyu Wang, Zhedong Zheng, Chenggang Yan, Jiyong Zhang, Yaoqi Sun, Bolun Zheng, and Yi Yang. Each Part Matters: Local Patterns Facilitate Cross-View Geo-Localization. *IEEE Transactions on Circuits and Systems for Video Technology*, 32:867–879, 2 2022.
- [147] Xiaolong Wang, Runsen Xu, Zhuofan Cui, Zeyu Wan, and Yu Zhang. Fine-grained cross-view geo-localization using a correlation-aware homography estimator. *Advances in Neural Information Processing Systems*, 36:5301–5319, 2023.

- [148] Michael Warren, David Mckinnon, Hu He, and Ben Upcroft. Unaided Stereo Vision Based Pose Estimation. pages 1–3. Australian Robotics and Automation Association, 2010.
- [149] Stephan Weiss, Davide Scaramuzza, and Roland Siegwart. Monocular-SLAM-based navigation for autonomous micro helicopters in GPS-denied environments. *Journal of Field Robotics*, 28(6):854–874, 2011.
- [150] P. Wenzel, R. Wang, N. Yang, Q. Cheng, Q. Khan, L. von Stumberg, N. Zeller, and D. Cremers. 4Seasons: A cross-season dataset for multi-weather SLAM in autonomous driving. In *Proceedings of the German Conference on Pattern Recognition (GCPR)*, 2020.
- [151] Frieder Wittmann, Olivier Lambercy, and Roger Gassert. Magnetometer-Based Drift Correction During Rest in IMU Arm Motion Tracking. *Sensors*, 19:1312, 3 2019.
- [152] Scott Workman, Richard Souvenir, and Nathan Jacobs. Wide-Area Image Geolocalization with Aerial Reference Imagery. pages 3961–3969. IEEE, 12 2015.
- [153] Tete Xiao, Mannat Singh, Eric Mintun, Trevor Darrell, Piotr Dollár, Ross Girshick, and Facebook AI Research. Early Convolutions Help Transformers See Better. In *NeurIPS*, 2021.
- [154] Peng Hang Xiong Lu Jian Sun Xuekai Liu, Qian Chen. A Self-Organizing Cooperative Control Framework for Connected Automated Vehicles at Unsignalized Roundabouts. In *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2024.
- [155] Hongji Yang, Xiufan Lu, and Yingying Zhu. Cross-view Geo-localization with Layer-to-Layer Transformer. *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, 7 2021.
- [156] Amir Roshan Zamir and Mubarak Shah. Image Geo-Localization Based on Multiple-Nearest Neighbor Feature Matching Using Generalized Graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36:1546–1558, 8 2014.

- [157] Richard S. Zemel, Christopher K.I. Williams, and Michael C. Mozer. Lending direction to neural networks. *Neural Networks*, 8:503–512, 1 1995.
- [158] Menghua Zhai, Zachary Bessinger, Scott Workman, and Nathan Jacobs. Predicting Ground-Level Scene Layout from Aerial Imagery. volume 2017-January, pages 4132–4140. IEEE, 7 2017.
- [159] Ji Zhang and Sanjiv Singh. LOAM: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems*, volume 2, pages 1–9. Berkeley, CA, 2014.
- [160] Qingwang Zhang and Yingying Zhu. Aligning geometric spatial layout in cross-view geo-localization via feature recombination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 7251–7259, 2024.
- [161] Wei Zhang and Jana Kosecka. Image Based Localization in Urban Environments. In *Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT’06)*, pages 33–40. IEEE, 6 2006.
- [162] Xiao Zhang, Wenjie Zhao, Changxuan Liu, and Jun Li. Distributed Multi-Target Search and Surveillance Mission Planning for Unmanned Aerial Vehicles in Uncertain Environments. *Drones*, 7(6):355, 2023.
- [163] Zhedong Zheng, Yunchao Wei, and Yi Yang. University-1652: A Multi-view Multi-source Benchmark for Drone-based Geo-localization. pages 1395–1403. ACM, 10 2020.
- [164] Sijie Zhu, Mubarak Shah, and Chen Chen. TransGeo: Transformer Is All You Need for Cross-view Image Geo-localization. In *2022 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1152–1161. IEEE, 6 2022.
- [165] Sijie Zhu, Taojiannan Yang, and Chen Chen. Revisiting Street-to-Aerial View Image Geo-localization and Orientation Estimation. In *2021 IEEE Winter Conf. on Applications of Computer Vision (WACV)*, pages 756–765. IEEE, 1 2021.

- [166] Sijie Zhu, Taojiannan Yang, and Chen Chen. VIGOR: Cross-View Image Geo-localization beyond One-to-one Retrieval. In *2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5316–5325. IEEE, 6 2021.
- [167] Yingying Zhu, Bin Sun, Xiufan Lu, and Sen Jia. Geographic Semantic Network for Cross-View Image Geo-Localization. *IEEE Trans. on Geoscience and Remote Sensing*, 60:1–15, 2022.
- [168] Yingying Zhu, Hongji Yang, Yuxin Lu, and Qiang Huang. Simple, Effective and General: A New Backbone for Cross-view Image Geo-localization. 2 2023.