IMVIP 2017

IRISH MACHINE VISION AND IMAGE PROCESSING

CONFERENCE PROCEEDINGS

30th August – 1st September 2017

Maynooth University,

Maynooth, Co. Kildare, Ireland.

Editors:

John McDonald, Charles Markham and Adam Winstanley

FotoNation® Movidius an Intel company Maynooth University National University of Ireland Maynooth IPRCS Irish Pattern Recognition and Classification Society

Published by the Irish Pattern Recognition & Classification Society

iprcs.org

ISBN 978-0-9934207-2-6

Introduction

The 2017 Irish Machine Vision and Image Processing Conference (IMVIP 2017) was hosted this year at Maynooth University, under the organisation of the Department of Computer Science.

IMVIP is Ireland's primary meeting for those researching in the fields of machine vision and image processing. The conference has been running since 1997 and provides a forum for the exchange of ideas and the presentation of research conducted both in Ireland and worldwide.

IMVIP is a single track conference consisting of high quality previously unpublished contributed papers focussing on both theoretical research and practical experiences in all areas. After a rigorous review process, this year 19 papers were selected for oral presentation at the conference, with a further 22 selected for poster presentation. We wish to sincerely thank the members of the Programme Committee for generously giving their time, effort and expertise in reviewing the submissions.

Continuing the tradition of inviting high-profile speakers to IMVIP, we are delighted to have keynote presentations at IMVIP 2017 from Dr. Thomas Whelan (Oculus Research), Dr. Maurice Fallon (University of Oxford), and Dr. Stefan Leutenegger (Imperial College London).

IMVIP is run in association with the Irish Pattern Recognition & Classification Society (iprcs.org), a member organisation of the International Association for Pattern Recognition (IAPR) and the International Federation of Classification Society (IFCS). In addition to IPRCS, we would like to thank the sponsors of IMVIP 2017 FotoNation, Movidius, as well as Maynooth University Research Development Office.

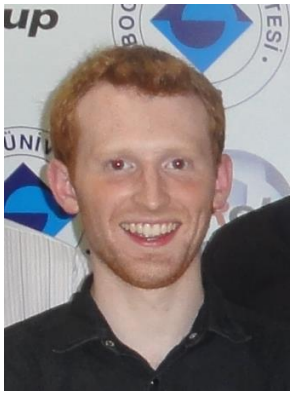
We would like to express our gratitude to a number of people at Maynooth University that helped in the organisation of IMVIP 2017. Firstly thank you to the members of the local organising committee: Patrick Marshall, William Clifford, Louis Gallagher, and Toby Burns. We are also very grateful to Barbara McCormack, Audrey Kinch, and Saoirse Reynolds at the Russell Library for organising the exhibition for the IMVIP delegates. Finally we would like to thank Fiona Smith, Caroline Kingston and all the staff at Maynooth Campus Conference & Accommodation for all of their assistance throughout the year.

John McDonald, Charles Markham & Adam Winstanley
Maynooth University
Ireland
August 2017

 ISBN 978-0-9934207-2-6

Keynote speakers: Thomas Whelan

Sponsored by Maynooth University National University of Ireland Maynooth



Title: *Machine Perception*

Abstract: The advent of modern virtual reality has brought about a host of new and challenging problems in many fields, most notably in computer vision. The future of this technology is not just restricted to the domain of virtual reality, but has far greater reach into more challenging areas including mixed and augmented reality. The Surreal Vision team at Oculus Research focuses on real-time always on embedded machine perception. This talk will present a brief overview of Oculus Research, some of the machine perception challenges involved in mixed reality and some of the existing published solutions to these problems.

About the speaker: Dr. Thomas Whelan is currently a Research Scientist at Oculus Research in Redmond working with the Surreal Vision team. Previous to this he spent one year as a post doctoral research fellow at the Dyson Robotics Laboratory at Imperial College London, lead by Prof. Andrew J. Davison. He was previously a Ph.D. student at the National University of Ireland Maynooth under a 3 year post-graduate scholarship from the Irish Research Council. In 2012 he spent 3 months as a visiting researcher at Prof. John Leonard's group in CSAIL, MIT funded by a Science Foundation Ireland Short-Term Travel Fellowship. He received his B.Sc. (Hons) in Computer Science & Software Engineering from the National University of Ireland Maynooth in 2011. His research focuses on developing methods for dense real-time perception and its applications in SLAM, AR and VR. He is the principal author of two widely used open source real-time dense SLAM systems, ElasticFusion and Kintinuous.

 ISBN 978-0-9934207-2-6

Keynote speakers: Maurice Fallon

Sponsored by FotoNation

Title: *Mapping and Tracking for Legged Robots*

Abstract: In this talk I will present ongoing research on multi-sensor perception for applications to robotic manipulation, navigation and locomotion. The first topic will focus on creating volumetric maps suitable for collision free motion planning by incorporating humanoid proprioception into a surfel-based dense visual mapping system. Visual SLAM systems are notoriously fragile - either reliant on the presence of structure or assuming no dynamics in the scene. A second topic develops methods for high-frequency legged robot state estimation - combining proprioception, vision and LIDAR. We will describe the various challenges of doing so - including dynamic impacts, poor lighting conditions and various sources of sensor noise. Finally we present initial research on visual manipulator tracking. We explore the problem instead by tracking the robot's manipulator using dense vision and present initial research showing how we can use discriminative methods to propose candidate solutions. Example demonstrations will be given using some notable platforms including the Boston Dynamics Atlas, NASA Valkyrie and the HyQ quadruped.

About the speaker: Maurice Fallon is a Departmental Lecturer at the University of Oxford. His research is focused on probabilistic methods for localization and mapping. He has also made research contributions to state estimation for legged robots and is interested in dynamic motion planning and control. Of particular concern is developing methods which are robust in the most challenging situations by leveraging sensor fusion.

Dr. Fallon studied Electronic Engineering at University College Dublin and graduated in 2004. His PhD research in the field of acoustic source tracking was carried out in the Engineering Department of the University of Cambridge within the Signal Processing Group.

 ISBN 978-0-9934207-2-6

Immediately after his Ph.D., he moved to MIT. He worked as a post-doc and a research scientist in the Marine Robotics Group from 2008-2012. From 2012-2015 he was the perception lead of MIT's team in the DARPA Robotics Challenge - a multi-year competition developing technologies for semi-autonomous humanoid exploration and manipulation in disaster situations. The MIT DRC team competed in several phases of the international competition, finishing 7th.

From 2014, he was a Chancellor's Fellow and Lecturer at University of Edinburgh. There he led research in collaboration with NASA's humanoid robotics program before moving to Oxford in April 2017.

 ISBN 978-0-9934207-2-6

Keynote speakers: Stefan Leutenegger

Sponsored by Movidius an Intel company

Title: *Spatial Perception for Mobile Robots*

Abstract: Mobile robots need dedicated sensing and processing for localisation and mapping as well as scene understanding. Recent years have brought tremendous advances in vision sensors (e.g. RGB-D cameras) and processing power (e.g. GPUs) that have led us to design new algorithms that will empower the next generation of mobile robots. With the arrival of deep learning, we are furthermore now in the position to link respective unprecedented performance in scene understanding with 3D mapping. In this talk, I will go through some recent algorithms and software we have developed as well as their application to mobile robots, including drones.

About the speaker: Stefan Leutenegger is a Lecturer (USA equivalent Assistant Professor) in Robotics at Imperial College London. He also Co-leads the Dyson Robotics Lab with Prof Andrew Davison. His research is centered around autonomous robot navigation with a focus on algorithms for real-time on-board localisation inside a potentially unknown environment.

Stefan has received a BSc and MSc in Mechanical Engineering from ETH Zurich in 2006, 2008, respectively, and a PhD in 2014, working at the Autonomous Systems Lab of ETH Zurich on Unmanned Solar Airplanes: Design and Algorithms for Efficient and Robust Autonomous Operation.

 ISBN 978-0-9934207-2-6

Conference Chairs

- General Chair: John McDonald, Maynooth University
- Co-chair: Charles Markham, Maynooth University
- Co-chair: Adam Winstanley, Maynooth University

Local Organising Committee

- William Clifford, Maynooth University
- Louis Gallagher, Maynooth University
- Patrick Marshall, Maynooth University
- Toby Burns, Maynooth University

Programme Committee

- Abdullah Bulbul, Trinity College Dublin
- Aljosa Smolic, Trinity College Dublin
- Andy Shearer, National University of Ireland, Galway
- Anil Kokoram, Trinity College Dublin
- Antonio Fernández, University of Vigo, Spain
- Bob Fisher, University of Edinburgh
- Bryan W. Scotney, Ulster University
- Bryan Gardiner, Ulster University
- Cem Direkoglu, Middle East Technical University, Cyprus
- Derek Molloy, Dublin City University
- Dermot Kerr, Ulster University
- Donald Bailey, Massey University, University of New Zealand
- Fionn Murtagh, University of London, UK
- Francesco Bianconi, University of Perugia, Italy
- François Pitie, Trinity College Dublin
- George Moore, Ulster University
- Guillaume Gales, The Foundry
- Jane Courtney, Dublin Institute of Technology
- Joan Condell, Ulster University
- John Barron, The Univerity of Western Ontario, Canada
- Kathleen Curran, Univerity College Dublin
- Kenneth Dawson-Howe, Trinity College Dublin
- Kevin McGuinness, Dublin City University
- Larbi Boubchir, University of Paris 8
- Nicholas Devaney, National University of Ireland, Galway
- Padraig Corcoran, Cardiff University
- Paul Mc Kevitt, Ulster University
- Paul Miller, Queen's University of Belfast
- Paul Whelan, Dublin City University
- Philip Morrow, Ulster University
- Reyer Zwiggelaar, Aberystwyth University, UK
- Robert Sadlier, Dublin City University

 ISBN 978-0-9934207-2-6

- Rozenn Dahyot, Trinity College Dublin
- Rudi Villing, Maynooth University
- Sally McClean, Ulster University
- Sonya Coleman, Ulster University
- Sudeep Sarkar, University of South Florida, USA
- Tom Naughton, Maynooth University
- Yanpeng Cao, Zhejiang University

 ISBN 978-0-9934207-2-6

Table of Contents

 ISBN 978-0-9934207-2-6

 ISBN 978-0-9934207-2-6

A Modular Scheme for Artifact Detection in Stereoscopic Omni-Directional Images

Sebastian Knorr, Simone Croci and Aljosa Smolic

School of Computer Science and Statistics
Trinity College Dublin, the University of Dublin, Ireland.

Abstract

With the release of new head-mounted displays (HMDs) and new omni-directional capture systems, 360-degree video is one of the latest and most powerful trends in immersive media, with an increasing potential for the next decades. However, especially creating 360-degree content in 3D is still an error-prone task with many limitations to overcome. This paper describes the critical aspects of 3D content creation for 360-degree video. In particular, conflicts of depth cues and binocular rivalry are reviewed in detail, as these cause eye fatigue, headache, and even nausea. Both the reasons for the appearance of the conflicts and how to detect some of these conflicts by objective image analysis methods are detailed in this paper. The latter is the main contribution of this paper and part of long-term research roadmap of the authors in order to provide a comprehensive framework for artifact detection and correction in 360-degree videos. Then, experimental results are demonstrating the performance of the proposed approaches in terms of objective measures and visual feedback. Finally, the paper concludes with a discussion and future work.

Keywords: 360-degree video, omni-directional images, 3D quality assessment, binocular rivalry, conflicts of depth cues

1 Introduction

360-degree video, also called live-action virtual reality (VR), is one of the latest and most powerful trends in immersive media, with an increasing potential for the next decades. In particular, head-mounted display (HMD) technology like e.g. HTC Vive, Oculus Rift and Samsung Gear VR is maturing and entering professional and consumer markets. On the other side, capture devices like e.g. Facebook's Surround 360 camera, Nokia Ozo and Google Odyssee are some of the latest technologies to capture 360-degree video in stereoscopic 3D (S3D).

Figure 1: Example of overlapping errors

However, capturing 360-degree videos is not an easy task as there are many physical limitations which need to be overcome, especially for capturing and post-processing in S3D. In general, such limitations result in artifacts which cause visual discomfort when watching the content with a HMD. The artifacts or issues can be divided into three categories: binocular rivalry issues, conflicts of depth cues and artifacts which occur in both monocular and stereoscopic 360-degree content production (see Section 2 for further details). Issues of the first two categories have been investigated for standard S3D content e.g. for cinema screens and 3D-TV [Knorr et al., 2012], [Vatolin et al., 2016], [Lambooij et al., 2009]. The third category consists of typical artifacts which only occur in multi-camera systems used for panorama capturing. As native S3D 360-degree video production is still very error-prone, especially with respect to binocular rivalry issues, many high-end S3D productions are shot in 2D 360-degree and post-converted to S3D.

 ISBN 978-0-9934207-2-6

This paper is dealing with automatic artifact detection in omni-directional images (ODIs) for quality control within the post-production workflow. To our knowledge, there is no scientific publication in this area and thus an open research field of high importance. Currently, the post-production workflow basically consists of six steps: 1) data ingest, 2) rough stitching of camera views (automatically), 3) fine stitching (manually), 4) color-grading, 5) editing and 6) finishing (rendering). Especially, the fine stitching process, which includes removal of stitching and blending artifacts as well as wire-, rig-, shadow- and contamination removal, is a labor intensive process with many intermediate rendering steps in order to check the quality of the results on HMDs. It is our goal to provide algorithms and tools for automatic detection and, if possible, correction of artifacts in order to give automatic feedback to artists and reduce time and efforts in post.
The paper is structured as follows. In Section 2, the state of the art is reviewed, in particular the technical challenges of 360-degree capturing and the resulting artifacts and issues. Then, in Section 3, we describe the proposed modular system and approaches for color mismatch and geometrical misalignment detection, which is the main contribution of this paper and part of our long-term research roadmap for quality control in 360-degree videos. In Section 4, experimental results for a test dataset of 13 ODIs captured with 7 different 360 capture devices (Google Odyssee, Jaunt, OmniCam-3D, Ozo, Panocam, Surround-360 and a self-constructed system with Mobius cameras) are demonstrating the performance of the proposed approaches in terms of objective measures and visual feedback. Finally, the paper concludes with a discussion and future work in Section 5.

2 State of the Art

Omni-directional image stitching is a process of synthesizing multiple views together on a common virtual surface. The overlapping regions between the cameras are first matched using different planar transformation models (e.g. affine, perspective or cubic transformation models), and the transition between the overlapping parts is estimated via blending parameters. Then, the views are blended and warped onto the omni-directional surface using the estimated geometric relation between the omni-directional surface and the image coordinates.
ODI stitching is challenging as the capturing devices have some inevitable drawbacks, e.g. the optical centers of the individual cameras do not share the same center of projection. However, applying planar transformation models in order to synthesize multiple views together on a common virtual surface is only valid if the captured scene is a planar surface itself or the cameras share the same center of projection [Hartley and Zisserman, 2003].

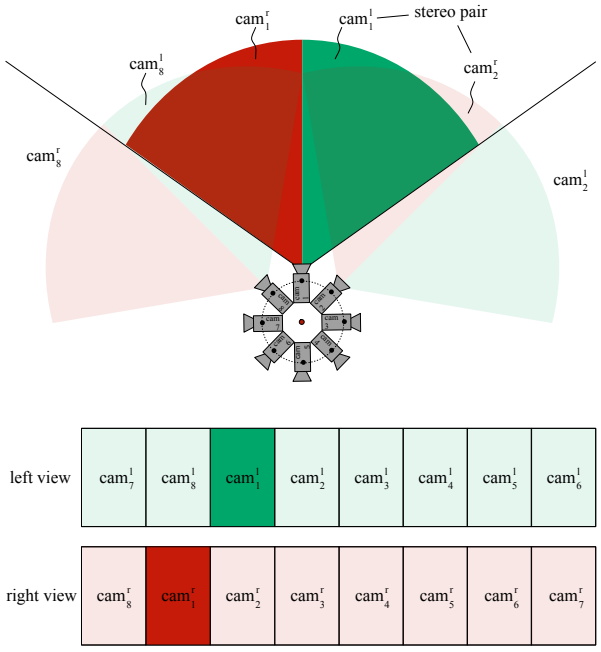


Figure 2: Principles of an off-centered slit camera model for capturing 360-degree in S3D

For off-centered cameras, transformation errors occur which increase with the off-center distance and the amount of depth within the captured scene. Figure 1 shows exemplary the viewport of an ODI with a stitching and blending error caused by a cross-fading in the overlapping area of two adjacent cameras.
In order to reduce stitching errors, the baseline between the cameras should be minimized. On the other side, the baseline between the cameras of different views needs to be increased for S3D content creation as parallax is required for generating a 3D effect, i.e. stitching and blending errors actually increase in S3D 360-degree content.
Figure 2 shows the principles of an off-centered slit camera model for capturing 360-degree in S3D. Half of the field of view of each camera is dedicated to either the left view of a stereoscopic ODI and the other half

 ISBN 978-0-9934207-2-6

is dedicated to the right view of a stereoscopic ODI. At this point, it needs to be mentioned that two adjacent cameras must share at least 50% of their fields of view as the right half of one camera view (e.g. cam_1^l) and the left half of the adjacent view (e.g. cam_2^r) form a stereoscopic image pair as illustrated in Figure 2.

2.1 Common artifacts

Whether native S3D or conversion from 2D to stereoscopic 3D, they both must display a high technical quality, while also taking all aspects of the human binocular visual system into account in order to reduce visual discomfort [Knorr et al., 2012]. This is even more essential, the higher the degree of immersion is, which is the case for VR by using HMDs. Artifacts arising from improper camera alignments, physical limitations of the chosen capture system, errors in post-production or compositing, etc. are still inevitable and can be categorized into three categories: binocular rivalry issues, conflicts of depth cues and artifacts which occur in both monocular and stereoscopic 360-degree content production.
Table 1 gives an overview of the first category. Most of the issues only occur in native S3D productions while issues of the second category (see Table 2) mainly appear in 2D to S3D conversion. Finally, Table 3 details issues which can occur in both 2D and S3D production as well as in native S3D or 2D to S3D conversion.

Artifact/ Issue	**Characteristics**	**Caused by**
Geometrical misalignment	Improper (vertical) alignment of left and right images	Cameras or lenses not properly aligned Tilting head or changing yaw while looking at the pole caps with a HMD
Luminance/ colorimetry	Difference in hue, saturation and/or intensity between left and right image	Cameras not properly matched (e.g. different aperture) Varying lighting conditions at different camera locations
Visual mismatch	Reflections, lens flares, polarization Contamination Missing or different objects in one of the views	Varying lighting conditions at different camera locations Contamination due to environmental conditions (e.g. rain, dust, etc.) Compositing errors in post
Depth of field/ sharpness mismatch	Difference in sharpness or depth of field	Different aperture settings of cameras Focal length of cameras not properly matched
Synchronization	Left and right image sequences are not synchronized	Cameras are not synchronized/ gen-locked Editing errors in post
Hyperconvergence/ hyperdivergence	Objects are too close to or too far from the viewer's eyes	Too much negative or positive parallax between left and right image
Pseudo-3D	Left and right images are swapped	Swapped images in HMDs Editing error in post
Ghosting	Double edges of objects	Stitching and blending artifacts in post

Table 1: Binocular rivalry issues in stereoscopic 360-degree videos

2.2 Quality assessment

Over the last years, many publications focused on the assessment of 3D quality in terms of subjective and objective quality metrics. In [Khaustova et al., 2015], the authors investigated how viewer annoyance depends on various technical parameters such as vertical disparity, rotation and field-of-view mismatches as well as color and luminance mismatches between the views. [Chen et al., 2014] proposed several objective metrics for luminance mismatch and evaluated their correlation with the results of subjective experiments. In [Goldmann et al., 2010], an artifact specific to S3D video is analyzed in depth by evaluating visual discomfort caused by temporal asynchrony. Finally, in [Battisti et al., 2015], a full-reference metric is presented by evaluation of a large variety of measures by taking 2D picture quality, binocular rivalry and depth map degradation into account. The authors maximized the correlation with the mean opinion score (MOS) by using linear regression. However, none of the 3D quality assessment approaches in the literature deal with ODIs.
In this paper, however, the focus lies on artifact detection in order to support the artist by giving direct quality feedback during post-production, in particular for geometrical misalignment and color mismatch. Thus, full-reference objective quality metrics can not be applied in this application. The authors of [Dong et al., 2013] propose a stereo camera distortion detecting method based on statistical models in order to detect vertical misalignment, camera rotation, unsynchronized zooming, and color misalignment in native S3D content.

 ISBN 978-0-9934207-2-6

Depth conflict	Characteristics	Caused by
Vergence vs. accommodation	Eyes accommodate on screen plane but converge or diverge on objects in front or behind the screen plane	Parallax between objects in the left and right view
Stereopsis vs. interposition	Foreground objects are occluded by background objects	3D compositing errors in post
Accommodation vs. depth of field	Eyes accommodate on screen plane but scene or part of scene is out of focus	Wide aperture of cameras
Stereopsis vs. (aerial) perspective	Monocular depth cue "perspective" or "aerial perspective" does not match with binocular depth cue "stereopsis"	3D compositing errors in post
Stereopsis vs. motion parallax	Motion of objects does not match with their distance	3D compositing errors in post
Stereopsis vs. size	Relative or familiar size of objects does not match with their distance	3D compositing errors in post
Stereopsis vs. light and shading	Distance or shape of objects does not match with their shadings	3D compositing errors in post
Stereopsis vs. texture gradient	Texture gradients are not in line with the descending of depth in the scene	3D compositing errors in post

Table 2: Depth conflicts in stereoscopic 360-degree videos

Artifact/ Issue	Characteristics	Caused by
Stitching artifacts	Visible seams and misaligned/ broken edges	Improper camera arrangement Registration and alignment errors in post
Blending artifacts	Visible color- and luminance mismatches of regions within an ODI	Varying lighting conditions at different camera locations Compositing errors in post
Warping artifacts	Visible deformations of objects	Improper camera arrangement Registration and alignment errors in post
Wobbling artifacts	Unsteady scene appearance over time	Temporal inconsistent stitching of camera views (non-stabilized image sequences)

Table 3: Artifacts in both monocular and stereoscopic 360-degree videos

[Voronov et al., 2013] introduce a large variety of artifact detection methods, including color mismatch and vertical disparity. With respect to the color mismatch approach, the RGB color space is used which, in our application, is inappropriate as the HSV color space is usually preferred in post-production.

3 Proposed Approach

Figure 3 shows an overview of the proposed modular system to detect color mismatch and vertical misalignment as part of an overall framework for detecting and correcting artifacts outlined in Subsection 2.1. All of the underlying algorithms are implemented in OFX and thus useable as plugins in professional post-production applications like Nuke, Fusion, Mamba FX or Natron (see Figure 4). The following subsections describe the underlying methods in more detail.

3.1 Geometrical misalignment

Geometrical misalignment, in particular vertical parallax, is present when objects in the scene are not vertically aligned between the left and right stereo images. In order to detect vertical parallax, we first compute sparse

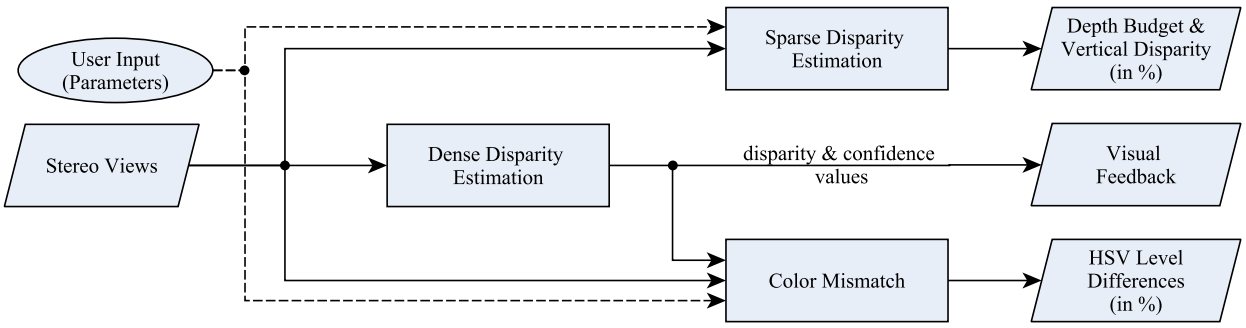


Figure 3: Overview of proposed system to detect color mismatch and vertical misalignment

 ISBN 978-0-9934207-2-6

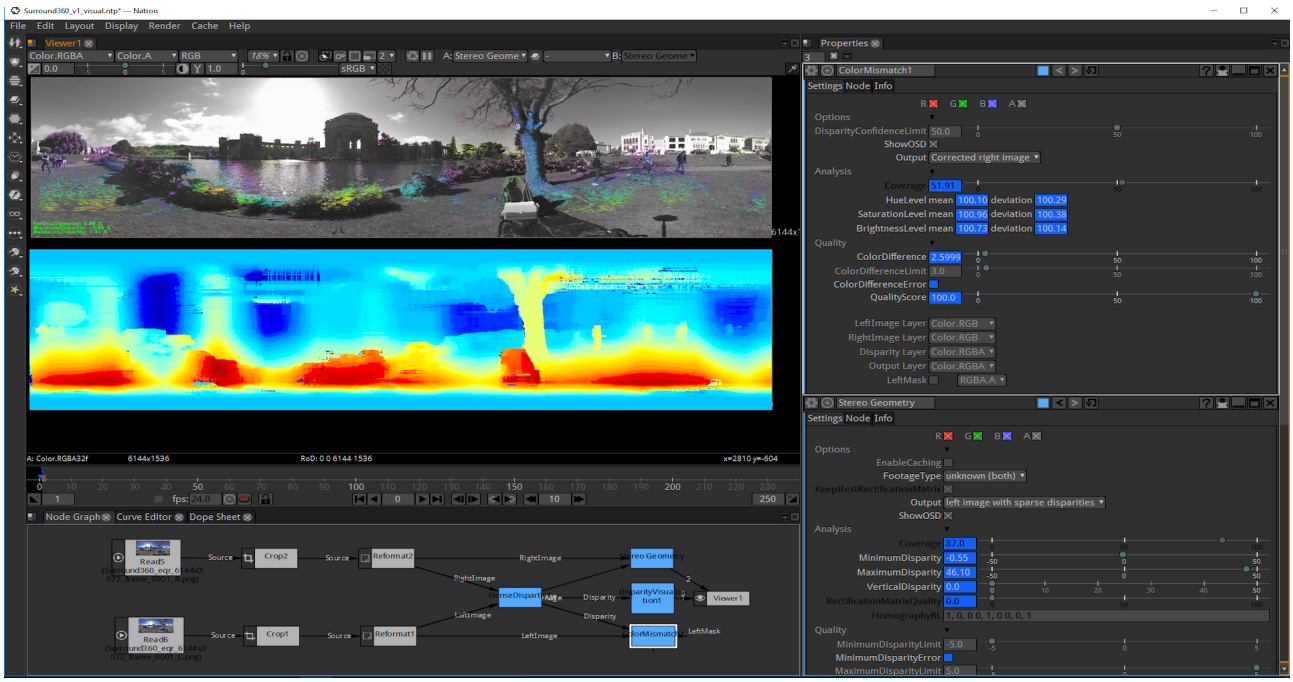


Figure 4: Screenshot of Natron including the node graph with the modules, visualization of sparse and dense disparities, and user input and output parameters for the *lake* ODI captured with the Surround-360 rig

disparities between left and right view. First, distinctive features are extracted in both stereo images using the SURF feature point detector and descriptor SURF [Bay et al., 2006]. Then, for each extracted feature point a descriptor is computed that represents the local properties of the image around the feature point.
The feature points are then matched between the two stereo images as follows. Each feature point in the left image is compared to the feature points in the reference image by calculating the Euclidean distance between their descriptor vectors. A matching pair is detected if its distance is closer than 0.7 times the distance of the second nearest neighbour [Bay et al., 2006]. Unreliable matches are eliminated with RANSAC [Fischler and Bolles, 1981] using the epipolar geometry as validation model. Finally, the vertical component of the disparities of all remaining matches is computed and displayed as percentage of the image width.

3.2 Color mismatch

The color mismatch module compares the color properties between the two stereo images. In the first step, pixels which are present in both the left and right images are detected using the semi-global matching approach for dense disparity estimation as described in [Hirschmuller, 2008]. The confidence maps of the resulting left-to-right and right-to-left disparity maps can be thresholded in order to take only higher reliable disparities into account. In the experimental results in Section 4, we applied a threshold of 50%. For all corresponding pixels above the threshold, the color properties, i.e. mean and standard deviation of the color channels, are computed for the left and right images, respectively, as introduced by [Reinhard et al., 2001]. Instead of using the $l\alpha\beta$ color space, as proposed by the authors, we extract the same statistics but in the HSV color space for reasons mentioned above.
With Ω_l and Ω_r as the sets of corresponding pixels in the left image I_l and right image I_r, and with $I_l(p)$ and $I_r(p)$ as the colors at the pixel p defined in the HSV color space, the means for each channel are defined as

$$\mu_x = \frac{1}{|\Omega_x|} \sum_{p \in \Omega_x} I_x(p), \quad x \in \{l, r\} \tag{1}$$

and the standard deviations for each color channel as

$$\sigma_x = \sqrt{\frac{1}{|\Omega_x|} \sum_{p \in \Omega_x} (I_x(p) - \mu_x)^2}, \quad x \in \{l, r\}. \tag{2}$$

 ISBN 978-0-9934207-2-6

For comparison, we take the left image as reference and compute the mean difference $(\mu_r - \mu_l)/\mu_l$ and the standard deviation difference $(\sigma_r - \sigma_l)/\sigma_l$. Finally, the determined statistics can be used for color transfer and thus applied in order to correct the right or left image, respectively.

4 Experimental Results

In our test scenario, we chose exemplary a dataset consisting of 13 equirectangular stereo ODIs captured with 7 different 360 capture devices (Google Odyssee, Jaunt, OmniCam-3D, Nokia Ozo, Panocam, Facebook's Surround-360 and a self-constructed system with Mobius cameras by Jim Waters[1]) and analyzed the images with respect to geometrical misalignment and color mismatch as proposed in Section 3. As the results will also heavily depend on the captured content and the amount of post-production efforts, they can only be seen as an indiction for the characteristics of the camera systems under test.
In order to have similar conditions for all images, we only selected a vertical field of view of 60 degree instead of 180 degree for the equirectangular ODIs for two reasons: 1) Some of the images do either have no pole caps (like Google Odyssee and OmniCam-3D) or the rig at the nadir has not been removed (Panocam images) and 2) The zenith and nadir have a large degree of distortion within the equirectangular ODI (e.g. the first row represents a single pixel in the sphere).

4.1 Geometrical misalignment

Figure 5 illustrates the vertical parallax for each of the input images in percentage of the image width. It should be noted at this point, that here the image width was chosen to one fourth of the original image width as most of the HMDs only have a horizontal field of view of about 90 degrees.

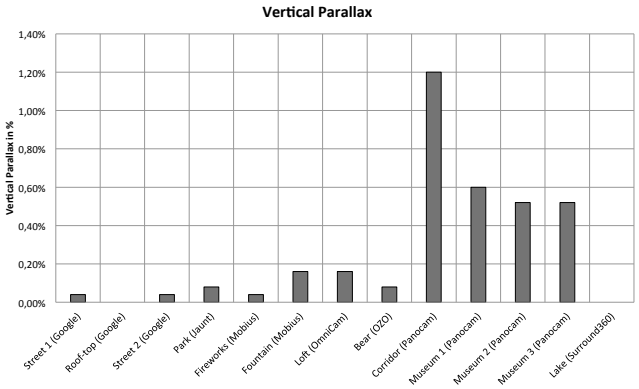


Figure 5: Vertical Parallax

The results show that the ODIs captured with the Panocam have by far the largest geometrical misalignments with up to 1.2% vertical parallax for the *corridor* sequence. In relation to S3D cinema movies, where the vertical parallax ranges between 0 and 0,14% according to the study in [Vatolin et al., 2016], most of the geometrical misalignments are extremely high, and thus would cause a high degree of visual discomfort.

4.2 Color mismatch

Figure 6 illustrates the color mismatches of the right images relative to the left images in terms of mean differences and standard deviation differences for each channel in HSV color space in percentage. The results show that the ODIs captured with the OmniCam have by far the larges color differences with a mean difference of brightness of 19.77%, followed by the *corridor* sequence (Panocam) and the self-constructed rigs with Mobius cameras. Most of the ODIs do only have a minor color mismatch between left and right view. However, we noticed visible color mismatches when applying a toggle view between the left and right images.
The reason for this discrepancy is quite obvious. While the OmniCam, Panocam and Mobius rig capture left and right ODIs independently, color differences between the views are inevitable as described in Section 2. Furthermore, the OmniCam is actually a mirror rig and thus, it is more error-prone to lighting conditions at different viewpoints. All the other ODIs were captured with an off-centered slit camera system as illustrated in Figure 2, i.e. left and right ODIs are captured with the same cameras. Although the overall color differences

[1] `https://photocreations.ca/3D/mobius_camera_rig.html`

 ISBN 978-0-9934207-2-6

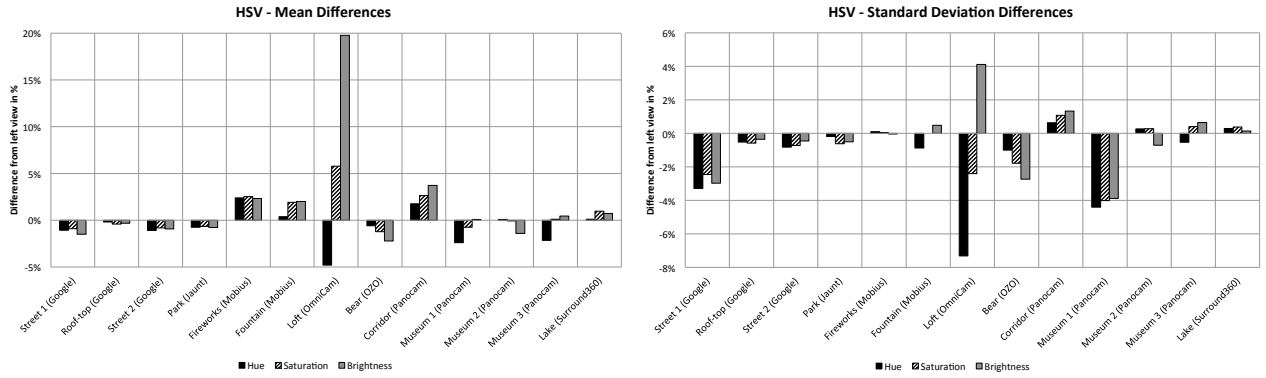


Figure 6: Color difference (mean and standard deviation) of the right image compared to the left image

for these cameras are small, the color differences between left and right images within a viewport of an HMD are still existing as each stereo pair of the input images belong to different cameras.

5 Conclusion and future work

The paper described a modular system for artifact detection in stereoscopic omni-directional images within the post-production, in particular the binocular rivalry issues: geometrical misalignment and color mismatch. The algorithms were exemplary applied to a dataset of 13 stereoscopic ODIs captured with 7 different 360-degree camera rigs. From the results, we can derive a couple of interesting properties and further challenges which need to be addressed. First, vertical parallax seems to be a serious issue in stereoscopic ODIs as it is much larger than in standard 3D cinema or 3DTV footage. Furthermore, capture devices which use independent stereo camera pairs tend to have an even larger vertical parallax. We have to note again, however, that we do not have any knowledge about the amount of post-production efforts spent for each of the ODIs under evaluation.
Secondly, global color mismatches between left and right ODI seem to be relatively small, except for the capture devices which use independent stereo camera pairs like OmniCam-3D, Panocam and the Mobius rig. However, visual inspection of the left and right ODIs often show significant local color mismatches. Thus, more investigation of local color mismatches, i.e. within the viewports is necessary as this is the area which the user actually sees and where binocular rivalry needs to be measured. The main challenges, however, are heavy distortions within each ODI of a stereo pair, in particular stitching, blending and warping artifacts which heavily degrade the dense disparity estimation results, and which affect the subsequent detection modules. One could argue, however, that the modules should support an artist within the fine-stitching process in post. If the distortions are too extreme for confident dense disparity estimation, the artist would probably notice this even without the support of the detection modules and would try to fix it.
The natural evolution of the current work is the extension of the artifact detection tools to the view adapted temporal dimension, i.e. the analysis of the viewports in stereoscopic 360-degree images and videos by also considering saliency as introduced in [Ana De Abreu, Cagri Ozcinar, 2017]. Furthermore, we will extend the system for the detection and possibly correction of other artifacts like e.g. sharpness mismatch. The modules for visual mismatch and pseudo-3D detection are already implemented and under evaluation. Finally, we want to motivate researchers from other research institutes to also focus on this research area in order to improve the quality of 360-degree videos.

Acknowledgments

The authors would like to thank Lutz Goldmann, Sebastian Schmiedecke and Ronald Kluth for their work on 3D quality control plugins.

References

[Ana De Abreu, Cagri Ozcinar, 2017] Ana De Abreu, Cagri Ozcinar, A. S. (2017). Look around you: saliency maps for omnidirectional images in VR applictions. In *9th International Conference on Quality of Multimedia Experience (QoMEX)*.

[Battisti et al., 2015] Battisti, F., Carli, M., Stramacci, A., Boev, A., and Gotchev, A. (2015). A perceptual quality metric for high-definition stereoscopic 3D video. In *Image Processing: Algorithms and Systems XIII, 939916*.

[Bay et al., 2006] Bay, H., Tuytelaars, T., and Van Gool, L. (2006). SURF: Speeded up robust features. In *Lecture Notes in Computer Science*, volume 3951, pages 404–417.

[Chen et al., 2014] Chen, J., Zhou, J., Sun, J., and Bovik, A. C. (2014). Binocular mismatch induced by luminance discrepancies on stereoscopic images. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6.

[Dong et al., 2013] Dong, Q., Zhou, T., Guo, Z., and Xiao, J. (2013). A stereo camera distortion detecting method for 3DTV video quality assessment. In *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 1–4.

[Fischler and Bolles, 1981] Fischler, M. and Bolles, R. (1981). RANdom SAmpling Consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. Assoc. Comp. Mach.*, 24:381–395.

[Goldmann et al., 2010] Goldmann, L., Lee, J. S., and Ebrahimi, T. (2010). Temporal synchronization in stereoscopic video: Influence on quality of experience and automatic asynchrony detection. *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pages 3241–3244.

[Hartley and Zisserman, 2003] Hartley, R. and Zisserman, A. (2003). *Multiple View Geometry in Computer Vision*. Cambridge University Press.

[Hirschmuller, 2008] Hirschmuller, H. (2008). Stereo Processing by Semiglobal Matching and Mutual Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341.

[Khaustova et al., 2015] Khaustova, D., Fournier, J., Wyckens, E., and Le Meur, O. (2015). An objective method for 3D quality prediction using visual annoyance and acceptability level.

[Knorr et al., 2012] Knorr, S., Ide, K., Kunter, M., and Sikora, T. (2012). The Avoidance of Visual Discomfort and Basic Rules for Producing "Good 3D" Pictures. *SMPTE Motion Imaging Journal*, 121(7):72–79.

[Lambooij et al., 2009] Lambooij, M., IJsselsteijn, W., Fortuin, M., and Heynderickx, I. (2009). Visual Discomfort and Visual Fatigue of Stereoscopic Displays: A Review. *Journal of Imaging Science and Technology*, 53(3):030201.

[Reinhard et al., 2001] Reinhard, E., Ashikhmin, M., Gooch, B., and Shirley, P. (2001). Color transfer between images. *IEEE Computer Graphics and Applications*, 21(5):34–41.

[Vatolin et al., 2016] Vatolin, D., Bokov, A., Erofeev, M., and Napadovsky, V. (2016). Trends in S3D-Movie Quality Evaluated on 105 Films Using 10 Metrics. *Proceedings of the SPIE, Stereoscopic Displays and Applications XXVII*, 2016(5):1–10.

[Voronov et al., 2013] Voronov, A., Vatolin, D., Sumin, D., Napadovsky, V., and Borisov, A. (2013). Methodology for stereoscopic motion-picture quality assessment. In *Proceedings of the SPIE, Stereoscopic Displays and Applications XXIV*, volume 8648.

A Robust Quality Measure for Quality-Guided 2D Phase Unwrapping Algorithms Based on Histogram Processing

Ambroise Moreau, Matei Mancas, Thierry Dutoit

Numediart Institute, University of Mons, Belgium

name.surname@umons.ac.be

Abstract

Among two-dimensional phase unwrapping algorithms, the quality-guided strategy offers one of the best trade-off between speed and accuracy. it assigns a quality value, also called reliability, to each pixel and removes the wraps on a path that goes from the highest to the lowest quality region. By doing so, challenging regions are unwrapped last, avoiding error propagation on the entire phase map. Strictly sorting the data by quality value is time-consuming but the process can be accelerated if they are roughly sorted in histogram bins that span the quality range instead. Nevertheless, using this histogram sorting scheme on existing quality measures can lead to erroneous results in some cases such as the presence of physical discontinuities in the true unwrapped phase map. In this paper, we propose a quality criterion that meets the requirements of histogram processing. Our method has been tested on challenging synthetic phase map with physical discontinuities and in the presence of noise. It shows encouraging results for data provided by shape measurement methods like phase-shifting profilometry.

Keywords: 2D Phase Unwrapping, Quality-Guided Phase Unwrapping, Image Processing, Shape Measurement, Phase-Shifting Profilometry.

1 Introduction

Phase unwrapping is an inherent problem in many three-dimensional shape measurement methods such as phase-shifting profilometry [Cong et al., 2015], magnetic resonance imaging [Maier et al., 2015], shearography [Van Brug, 1998] or interferometric synthetic aperture radar [Danudirdjo and Hirose, 2015]. These methods recover the phase of a two-dimensional modulated signal to measure physical quantities like ground elevation or surface profile. The retrieved phase map is usually constrained to its principal interval $(-\pi, \pi]$ and shows discontinuities that are not present in the original data. These discontinuities, or phase wraps, have to be removed through phase unwrapping.

For simple phase maps, phase unwrapping is an easy integration problem: the true phase value of each pixel is found by adding multiples of 2π based on the unwrapped neighbours [da Silva Maciel and Albertazzi, 2014]. For real phase maps, the process may face complications because of true discontinuities in the underlying physical quantity, undersampling in local areas or high local variations of signal-to-noise ratio [Herráez et al., 2002]. In these conditions, phase unwrapping becomes path-dependent. To overcome these difficulties, various methods that can be classified as temporal phase unwrapping or spatial phase unwrapping have been proposed in the past decades. As stated by Zhang et al., temporal algorithms are effective and robust but require several wrapped phase maps along the time dimension while spatial approaches works with a single phase map but do not deal well with disjoint regions and true phase discontinuities [Zhang et al., 2014].

Spatial methods can be further divided into global error minimization algorithms, quality-guided phase unwrapping, branch-cut approaches and region-growing methods [Herráez et al., 2002]. Algorithms falling in

 ISBN 978-0-9934207-2-6

the second category are known to offer the best trade-off between speed and accuracy. They rely on a quality map to measure the reliability of each pixel and unwrap the phase on a path that goes from the highest to the lowest quality region, avoiding error propagation on the entire phase map. Sorting pixels according to their quality value is a time-consuming task. To speed up the process, Lei et al. proposed a novel method based on histogram processing, but depending on the chosen quality criterion, the final result can be erroneous [Lei et al., 2015].

In this paper, we propose a new quality criterion, designed to be unaffected by the histogram sorting process. The remainder of the paper is organized as follows. Section 2 reviews the related work and gives an overview of existing quality criteria. Section 3 presents the new measure of quality. In Section 4, our algorithm is compared with two other quality criteria: the second differences reliability (SDR) [Herráez et al., 2002] and the random tilt reliability (RTR) [Arevalillo-Herráez et al., 2016]. Finally, Section 5 concludes on our work.

2 State of the Art

Many phase-measuring techniques use the phase of a modulated signal to measure a physical quantity such as the shape of an object. For instance, in phase-shifting profilometry, sinusoidal patterns projected on a scene, and deformed by its height, are observed with a camera which enables to recover the wrapped phase map by using the arctangent function on a combination of the captured images [Cong et al., 2015].

Mathematically, the spatial phase unwrapping of a wrapped pixel a, neighbour of an unwrapped pixel b, can be written as:

$$\Phi_a = \phi_a + 2\pi \cdot r\left(\frac{\Phi_b - \phi_a}{2\pi}\right) \tag{1}$$

where ϕ stands for the wrapped phase map, Φ stands for the unwrapped phase and $r(.)$ is a function that rounds its input to the closest integer. As explained by Itoh, the only requirement of Equation (1) is that the difference of the unwrapped phase between neighbouring pixels should fall within the principal interval $(-\pi, \pi]$ [Itoh, 1982]. The simplest unwrapping algorithm would go through the entire phase map, row after row, and remove the wraps using Equation (1). However, in practical cases, the final result could be wrong for four reasons: (i) sharp differences exceeding the principal interval caused by undersampling; (ii) noise leading to wrong unwrapped values; (iii) invalid area like shadows in FPP leading to error propagation; (iv) true discontinuities in the physical quantity mistaken for continuities [Zhang et al., 2014, Zhao et al., 2011]. Moreover, if one pixel is wrongly unwrapped, the subsequent ones will be erroneous too. Phase unwrapping becomes path-dependent. Quality-guided phase unwrapping solves these problems by assigning a quality value to each pixel from the wrapped phase map and following a path that unwraps problematic pixels in the end.

Depending on the application, the quality parameter can be computed from the raw measurement data or from the wrapped phase map itself [Zhao et al., 2011]. The correlation coefficient map in inSAR, the modulation map and the reliability map from least square fitting in phase-shifting profilometry belong to the first category whereas quality measures like the phase derivative variance map and the first and second phase differences maps belong to the second category. In this paper, we present a new quality criterion and compare it with two existing measures: the second differences reliability (SDR) [Herráez et al., 2002] and the random tilt reliability (RTR) [Arevalillo-Herráez et al., 2016].

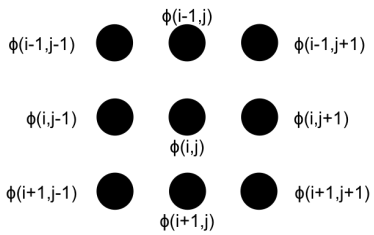


Figure 1: 3x3 window centered on the pixel of interest

SDR is computed on a 3 x 3 window centered on the pixel of interest as illustrated in Figure 1. Its quality,

 ISBN 978-0-9934207-2-6

or reliability, is equal to the sum of the squared second differences in the four possible directions. In this way, if the window contains at least one noisy pixel or goes through a physical discontinuity, the SDR is high and accounts for an unreliable region that should be unwrapped last. Note that the logic is inverted as pixels with high SDR are less reliable. Since SDR is computed on the wrapped phase map, the measure can only take value in the interval $[0, 32\pi^2]$. Equations (2)-(5) give the second differences in the four directions and Equation (6) is the quality formula. $W(.)$ is an operator that removes 2π discontinuity between adjacent pixels.

$$H = W(\phi(i, j-1) - \phi(i, j)) - W(\phi(i, j) - \phi(i, j+1)) \tag{2}$$

$$V = W(\phi(i+1, j) - \phi(i, j)) - W(\phi(i, j) - \phi(i+1, j)) \tag{3}$$

$$D_1 = W(\phi(i-1, j-1) - \phi(i, j)) - W(\phi(i, j) - \phi(i+1, j+1)) \tag{4}$$

$$D_2 = W(\phi(i-1, j+1) - \phi(i, j)) - W(\phi(i, j) - \phi(i+1, j-1)) \tag{5}$$

$$SDR = H^2 + V^2 + D_1^2 + D_2^2 \tag{6}$$

The random tilt criterion relies on the notion of residues, also known as poles. They are associated with local inconsistencies and have been used in quality-guided phase unwrapping and branch-cut methods as well. For two-dimensional signals, residues are defined as regions of 2 x 2 pixels with a non-zero sum of unwrapped values around any closed path. To measure the quality of a pixel, RTR computes the probability that a residue appears when a random tilt is added to the wrapped phase. Again, the logic is inverted and pixels with high RTR are less reliable. The measure can only take value in the interval $[0, 4\pi]$.

Although more than one measure of quality exist, the way the unwrapping path is built does not vary as much and can be summed up in five steps:

1. Compute the quality map.
2. Assign a quality value to each vertical and horizontal edge i.e the separation between two adjacent pixels. Its quality value is equal to the sum of those of the pixels it is made of.
3. Sort all of the edges according to their quality.
4. Assign each pixel to a different group. The initial number of groups is the same as the number of pixels in the phase map.
5. Go through the sorted edges and unwrap the pixels with respect to the two following rules:
   (a) If the pixels forming the current edge belong to two different groups, unwrap the smallest one and merge them into one.
   (b) If the pixels forming the current edge are already in the same group, do nothing.

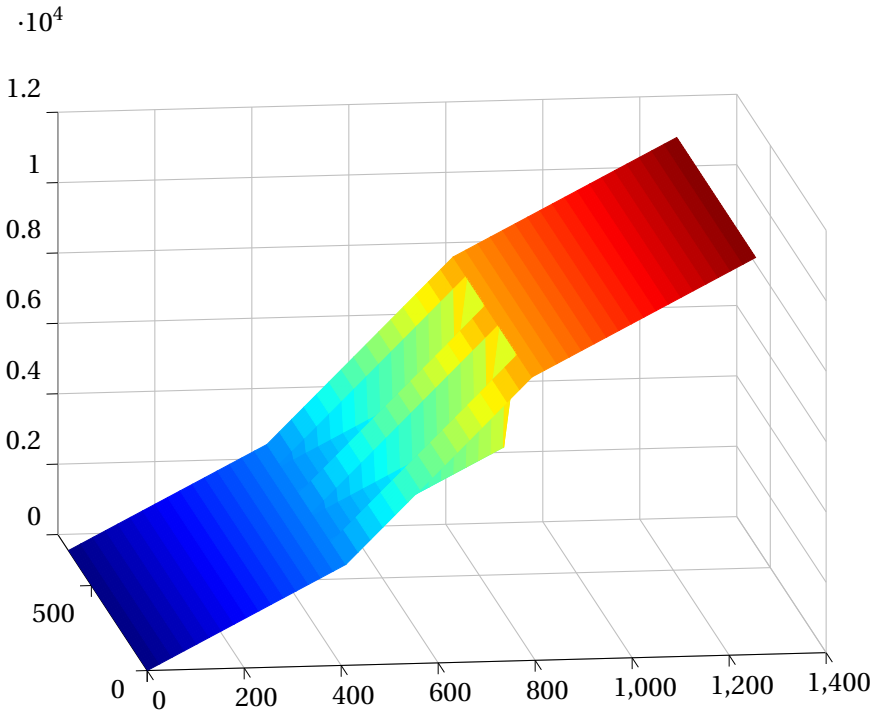
The third step of this process is time-consuming if the edges are strictly sorted but it can be boosted by roughly grouping them in histogram bins, as done in the work of Lei et al [Lei et al., 2015]. They applied this sorting scheme to SDR. Due to the general quality distribution, the sorting histogram bins are not of equal size; the ones smaller than a threshold are narrower to enable a finer sorting. Indeed, it is reasonable to expect the majority of pixels to be reliable. Experiments lead to an empirical value of $3\pi^2$ for the threshold and a number of 100 bins.

In section 3, we show how the histogram sorting can affect the result of phase unwrapping in some cases and propose a criterion designed to fit this fast sorting procedure.

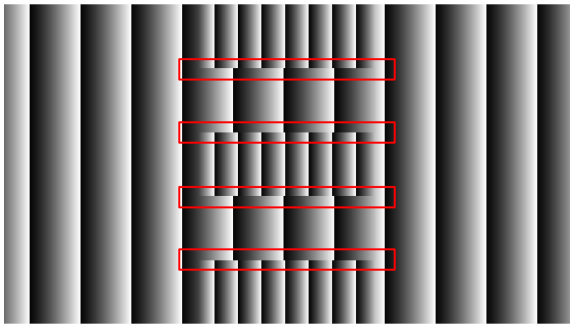
3 Proposed quality measure

Because of the periodic nature of the wrapped phase map, SDR can lead to variable reliability values along physical discontinuities. Some pixels are wrongly labeled as reliable and modify the unwrapping path if edges

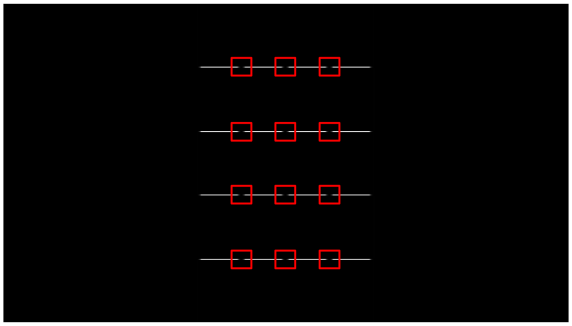
 ISBN 978-0-9934207-2-6



(a) Synthetic surface



(b) wrapped phase map



(c) SDR along the highlighted discontinuity line

(d) Unwrapped phase map with histogram sorting

(e) Unwrapped phase map with strict sorting

Figure 2: Phase unwrapping of a synthetic phase map

 ISBN 978-0-9934207-2-6

are not strictly sorted. To illustrate this behaviour, we applied the two sorting procedures on the synthetic surface shown in Figure 2(a). It contains four lines of discontinuities that should not be crossed by the unwrapping path, but Figure 2(d) shows that it is not the case when edges are sorted in the histogram bins. The twelve falsely reliable spots highlighted in Figure 2(c) are characterized by a SDR value of 0.011 which means they are sorted in the same bin as truly reliable edges. As a result, they are unwrapped too early. Crossing is avoided when edges are strictly sorted, as shown in Figure 2(e), since the falsely reliable regions are unwrapped after all the truly reliable pixels have been visited by the unwrapping path. An efficient way of solving this problem is to use a quality criterion that reduces the variability along physical discontinuities. When taking a closer look at H, V, D_1 and D_2 on one of the discontinuity lines shown in Figure 3, it appears they are all piecewise linear. Therefore, their first derivatives are constant and meet the requirement of low variability along discontinuities when using histogram sorting.

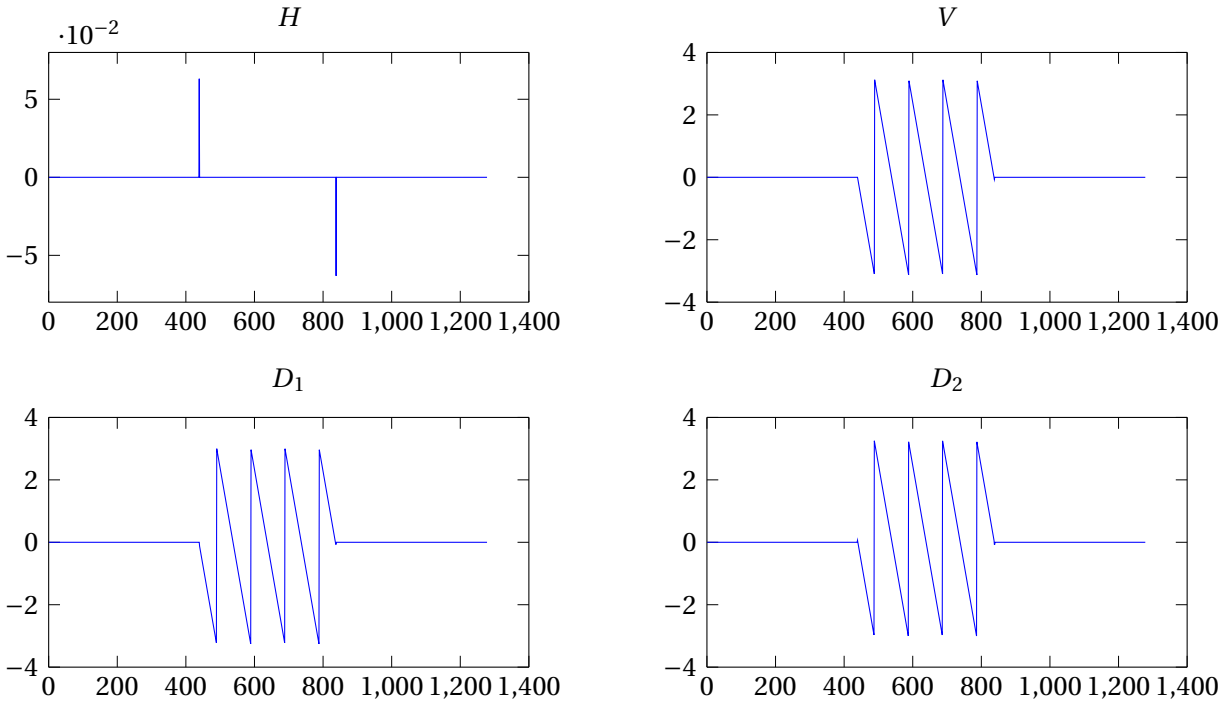


Figure 3: Second differences in the four directions along a discontinuity line

This observation leads to the definition of the *first derivative of second differences reliability* ($FDSDR$) quality criterion given by Equation (7).

$$FDSDR(i,j) = \left|W\left(D_1(i,j+1) - D_1(i,j-1)\right)\right| + \left|W\left(D_2(i,j+1) - D_2(i,j-1)\right)\right| \tag{7}$$

As the information carried by H and V is also found in D_1 and D_2, they are omitted from the measure to reduce computation time. Since D_1 and D_2 are wrapped between $-\pi$ and π, $FDSDR$ takes its value in the interval $[0, 8\pi]$.

4 Experimental results

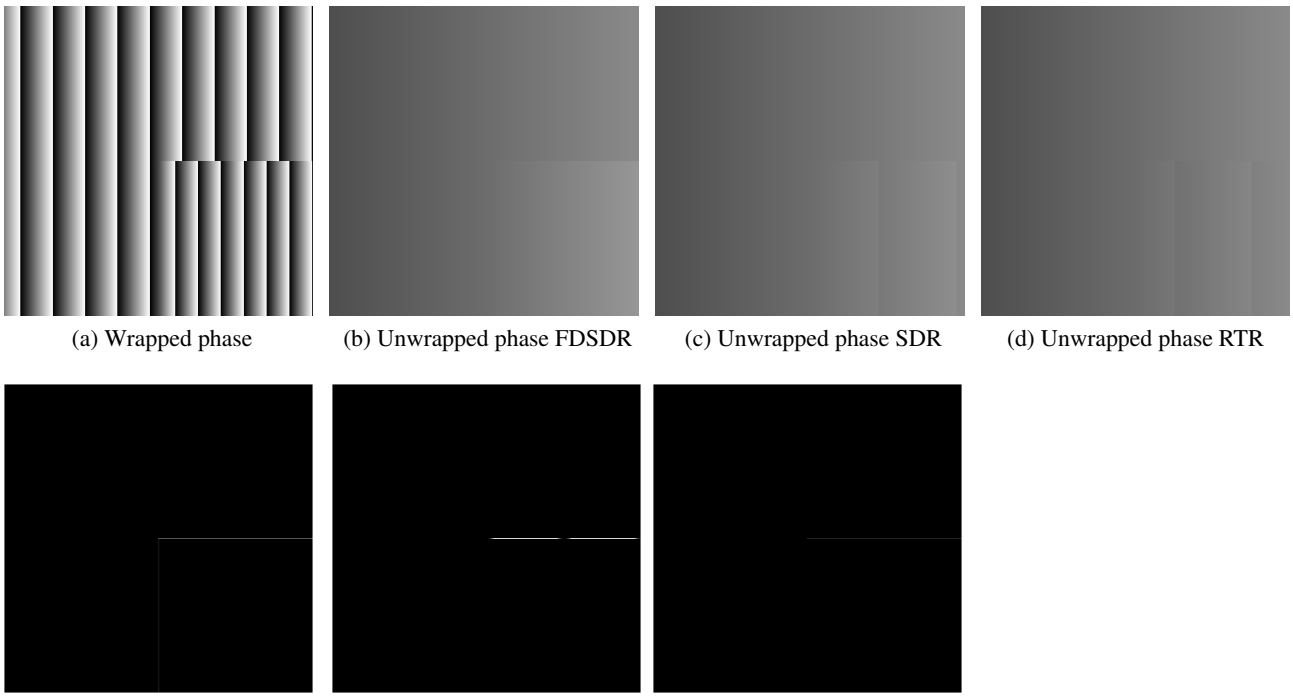
The proposed quality measure has been tested on synthetic data containing discontinuities and noise. SDR, RTR and $FDSDR$ have all been implemented in OpenCV [1]. The hardware used for the tests is a computer with a 2.7 GHz Intel®Core™i5-4210U and 16 Gbytes of RAM.

[1] `https://github.com/opencv`

 ISBN 978-0-9934207-2-6

4.1 Ramps with different slopes and without noise

Our first test was carried out on a phase map of size 720 x 720 divided into two planar regions with different slopes. This profile may seem simple but it is a challenging example that has been used as a reference in some previous work [Bioucas-Dias and Valadao, 2007, Ghiglia and Pritt, 1998]. Figure 4 shows the wrapped phase map (fig. 4(a)) , the result of phase unwrapping with histogram sorting applied on $FDSDR$ (Figure 4(b)), SDR (Figure 4(c)) and RTR (Figure 4(d)) and the corresponding quality maps (Figures 4(e), 4(f), 4(g)). The histogram was divided into twelve small bins and a large one. The threshold was set to π for $FDSDR$, $4\pi^2$ for SDR and 0.5π for RTR. In these conditions, unwrapping failed for SDR and RTR, discontinuities are present in the lower part of the phase maps because the unwrapping path crossed the discontinuity line. On average, unwrapping with FDSDR took 0.48 s when edges were sorted in the histogram while it took 0.59 s when they were strictly sorted. The sorting algorithm we used is the one implemented in the standard C++ library. As seen on Figure 4(e) and 4(g), $FDSDR$ and SDR are constant on the discontinuity line but unwrapping still fails when RTR is used as the quality measure. Indeed, the difference between reliable and unreliable pixels for $FDSDR$ is higher than the same difference for RTR which leads to a better sorting.



(a) Wrapped phase (b) Unwrapped phase FDSDR (c) Unwrapped phase SDR (d) Unwrapped phase RTR

(e) FDSDR map (f) SDR map (g) RTR map

Figure 4: Phase unwrapping of a simulated surface with two planar regions

4.2 Ramps with different slopes and with noise

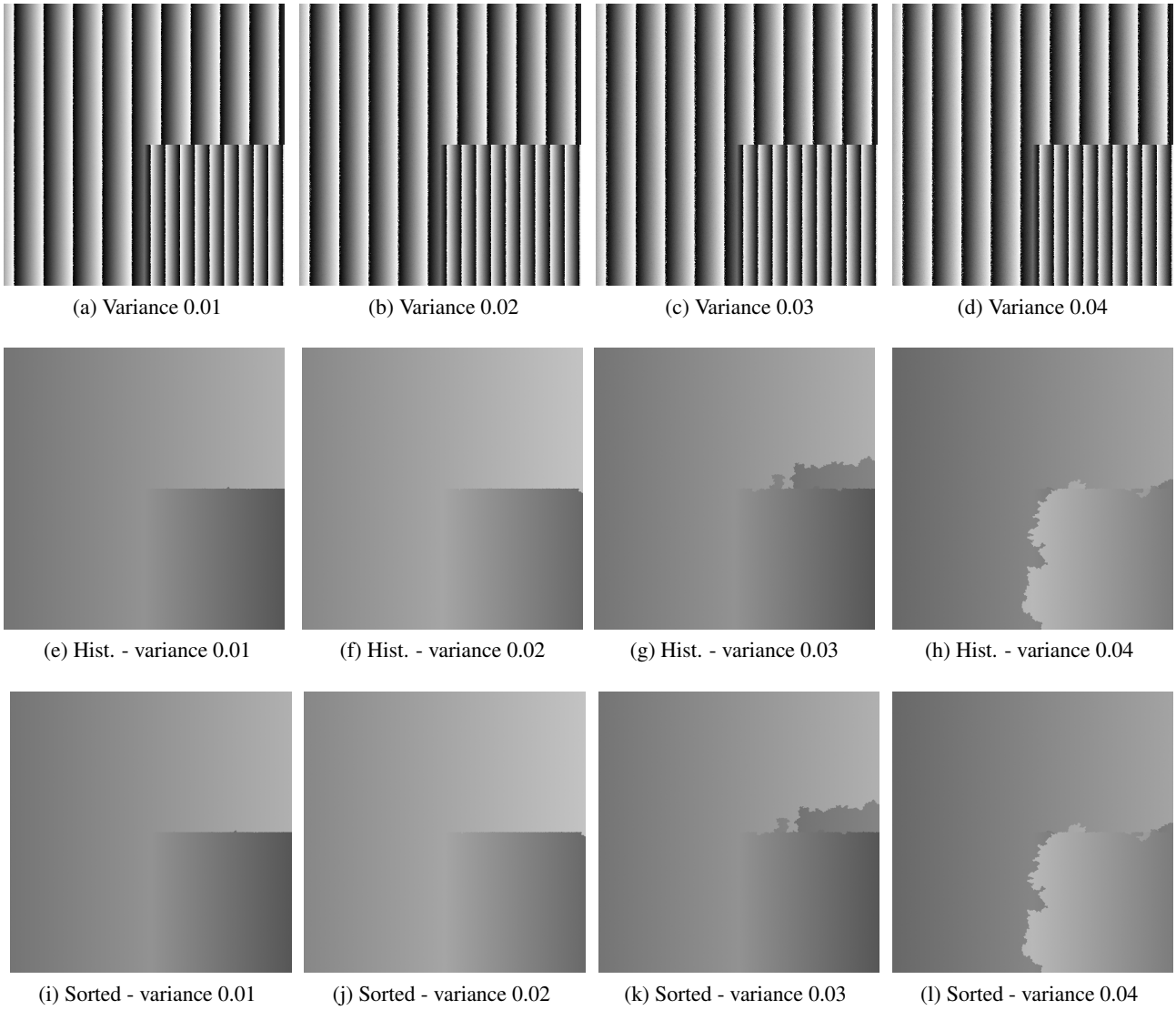
The ability to unwrap a noisy phase map is an important feature for any unwrapping algorithm. The second phase map was also divided into two planar regions with different slopes but this time, Gaussian white noise generated in Matlab was added to the data. Four levels of variance have been tested. Table 1 gives the mean processing time over ten unwrapping of the same phase map along with the minimal number of small bins required to get the best result. Except for the lowest noise level, processing times are equivalent when histogram processing is used. This observation does not hold when edges are strictly sorted. As announced by Lei et al. in their work, the number of small bins required to get the best result increases with the noise level but it does not impact the time performances [Lei et al., 2015]. On the other hand, the number of large bins has no influence on the unwrapped phase map.

 ISBN 978-0-9934207-2-6

Noise level	0.01	0.02	0.03	0.04
$FDSDR$ sorted (s)	0.69	0.70	0.70	0.70
$FDSDR$ histogram (s)	0.50	0.59	0.59	0.59
# of small bins	21	96	130	150

Table 1: Phase unwrapping for different levels of noise

Figure 5 shows the result of phase unwrapping for the four levels of noise. As expected, error propagation also increases with the noise level. An interesting observation is that the unwrapped phase map is the same for the two sorting procedures which means histogram sorting improves the processing speed without compromising the final result.



(a) Variance 0.01 (b) Variance 0.02 (c) Variance 0.03 (d) Variance 0.04

(e) Hist. - variance 0.01 (f) Hist. - variance 0.02 (g) Hist. - variance 0.03 (h) Hist. - variance 0.04

(i) Sorted - variance 0.01 (j) Sorted - variance 0.02 (k) Sorted - variance 0.03 (l) Sorted - variance 0.04

Figure 5: Phase unwrapping of a simulated surface with two planar regions for different levels of noise

5 Conclusion

In this paper, we have presented a new quality measure designed to work with the histogram sorting process proposed by Lei et al [Lei et al., 2015]. Based on the observation that H, V, D_1 and D_2 are piecewise linear, it reduces the quality variability along discontinuities by computing the first derivative of D_1 and D_2. Thus, the unwrapping path is less likely to cross discontinuity lines.

 ISBN 978-0-9934207-2-6

We have successfully tested our method on challenging examples. Histogram processing applied to our quality measure gives as good results as strict sorting while requiring less processing time. Further testing could be made about the histogram distribution since the number of large bins does not seem to impact the outcome of unwrapping. Moreover, the threshold is still empirical.

Finally, histogram processing could be applied to other measures of quality, sharing similar properties with $FDSDR$, meaning a low variability along discontinuity lines and a high difference of quality between reliable and unreliable pixels.

References

[Arevalillo-Herráez et al., 2016] Arevalillo-Herráez, M., Villatoro, F. R., and Gdeisat, M. A. (2016). A robust and simple measure for quality-guided 2d phase unwrapping algorithms. *IEEE Transactions on Image Processing*, 25(6):2601–2609.

[Bioucas-Dias and Valadao, 2007] Bioucas-Dias, J. M. and Valadao, G. (2007). Phase unwrapping via graph cuts. *IEEE Transactions on Image processing*, 16(3):698–709.

[Cong et al., 2015] Cong, P., Xiong, Z., Zhang, Y., Zhao, S., and Wu, F. (2015). Accurate dynamic 3d sensing with fourier-assisted phase shifting. *IEEE Journal of Selected Topics in Signal Processing*, 9(3):396–408.

[da Silva Maciel and Albertazzi, 2014] da Silva Maciel, L. and Albertazzi, A. G. (2014). Swarm-based algorithm for phase unwrapping. *Applied optics*, 53(24):5502–5509.

[Danudirdjo and Hirose, 2015] Danudirdjo, D. and Hirose, A. (2015). Anisotropic phase unwrapping for synthetic aperture radar interferometry. *IEEE Transactions on Geoscience and Remote Sensing*, 53(7):4116–4126.

[Ghiglia and Pritt, 1998] Ghiglia, D. C. and Pritt, M. D. (1998). *Two-dimensional phase unwrapping: theory, algorithms, and software*, volume 4. Wiley New York.

[Herráez et al., 2002] Herráez, M. A., Burton, D. R., Lalor, M. J., and Gdeisat, M. A. (2002). Fast two-dimensional phase-unwrapping algorithm based on sorting by reliability following a noncontinuous path. *Applied Optics*, 41(35):7437–7444.

[Itoh, 1982] Itoh, K. (1982). Analysis of the phase unwrapping algorithm. *Appl. Opt*, 21(14):2470.

[Lei et al., 2015] Lei, H., Chang, X.-y., Wang, F., Hu, X.-T., and Hu, X.-D. (2015). A novel algorithm based on histogram processing of reliability for two-dimensional phase unwrapping. *Optik-International Journal for Light and Electron Optics*, 126(18):1640–1644.

[Maier et al., 2015] Maier, F., Fuentes, D., Weinberg, J. S., Hazle, J. D., and Stafford, R. J. (2015). Robust phase unwrapping for mr temperature imaging using a magnitude-sorted list, multi-clustering algorithm. *Magnetic resonance in medicine*, 73(4):1662–1668.

[Van Brug, 1998] Van Brug, H. (1998). Temporal phase unwrapping and its application in shearography systems. *Applied optics*, 37(28):6701–6706.

[Zhang et al., 2014] Zhang, Y., Wang, S., Ji, G., and Dong, Z. (2014). An improved quality guided phase unwrapping method and its applications to mri. *Progress In Electromagnetics Research*, 145:273–286.

[Zhao et al., 2011] Zhao, M., Huang, L., Zhang, Q., Su, X., Asundi, A., and Kemao, Q. (2011). Quality-guided phase unwrapping technique: comparison of quality maps and guiding strategies. *Applied optics*, 50(33):6214–6224.

 ISBN 978-0-9934207-2-6

Fast and Accurate Optical Flow based Depth Map Estimation from Light Fields

Yang Chen, Martin Alain, and Aljosa Smolic

V-SENSE project
Graphics Vision and Visualisation group (GV2)
Trinity College Dublin

Abstract

Depth map estimation is a crucial task in computer vision, and new approaches have recently emerged taking advantage of light fields, as this new imaging modality captures much more information about the angular direction of light rays compared to common approaches based on stereoscopic images or multi-view. In this paper, we propose a novel depth estimation method from light fields based on existing optical flow estimation methods. The optical flow estimator is applied on a sequence of images taken along an angular dimension of the light field, which produces several disparity map estimates. Considering both accuracy and efficiency, we choose the feature flow method as our optical flow estimator. Thanks to its spatio-temporal edge-aware filtering properties, the different disparity map estimates that we obtain are very consistent, which allows a fast and simple aggregation step to create a single disparity map, which can then converted into a depth map. Since the disparity map estimates are consistent, we can also create a depth map from each disparity estimate, and then aggregate the different depth maps in the 3D space to create a single dense depth map.

1 Introduction

Light fields aim to capture all light rays passing through a given volume of space [1]. Compared to traditional 2D imaging systems which capture the spatial intensity of light rays, a 4D light field contains the angular direction of the rays. Light fields have thus become a topic of growing interest in several research areas such as image processing, computer vision, and computer graphics. Applications include refocusing of an image after capture, rendering new images from virtual points of view, or computational displays for virtual and augmented reality. In this paper, we focus on depth map estimation from a light field.

3D scene reconstruction or depth estimation from light fields is a major topic of interest and many methods have been proposed in the past years. Several methods have been proposed which estimate disparity between views of the light field with respect to the center view using existing stereo-matching techniques [2], [3]. To better exploit the light field structure, novel approaches have been introduced relying either on angular patch analysis [4]–[6] or the Epipolar Plane Images (EPI) representation of light fields [7]–[11].

In this paper, we present a novel pipeline to estimate depth maps from light fields based on optical flow, where the measured displacements correspond to disparity, from which we can then obtain the depth. Our main contributions are: (a): We propose a novel depth map estimation scheme based on an spatio-angular edge aware optical flow [12] applied over an angular dimension of the light field. (b): We propose to combine the spatio-angular optical flow with the state-of-the-art coarse-to-fine patch matching method [13] as initialization, which significantly improves the results without increasing the running time. (c): We propose to directly combine our multiple and consistent depth map estimates in the 3D space to obtain very dense depth maps or point clouds. We show that the proposed approach achieves comparable performances to the best state-of-the-art method in terms of balance between speed and accuracy.

This paper is organized as follows. In section 2 we review the 4D structure of light fields and existing methods to retrieve depth from light fields, as well as the state-of-the-art optical flow estimation techniques. Section 3 describes more in details the proposed approach. Finally, in section 4 we evaluate the performance of our method.

2 Background and related work

2.1 Light field 4D structure

We adopt in this paper the common two-plane parametrization as shown in Figure 1, and a light field can be formally represented as a 4D function $\Omega \times \Pi \rightarrow \mathbb{R}, (x, y, s, t) \rightarrow L(x, y, s, t)$ in which the plane Ω represents the spatial distribution of

 ISBN 978-0-9934207-2-6

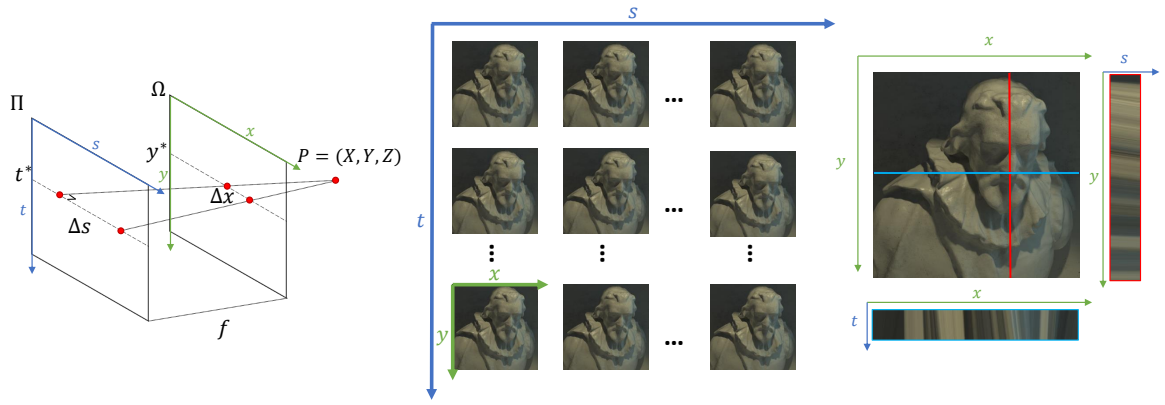


Figure 1: Light field two-plane parametrization (left), matrix of views representation (middle), and Epipolar Plane Images (EPI) representation (right).

light rays, indexed by (x, y), while Π corresponds to their angular distribution, indexed by (s, t).

Perhaps the easiest way to visualize a light field is to consider it as a collection of views, also called sub-aperture images, taken from several view points parallel to a common plane. The light field can then be considered as a matrix of views (see Figure 1). Note that an important assumption when using such representation is that the different views are rectified. Another common representation of light fields are Epipolar Plane Images (EPI), which are 2D slices of the 4D light field obtained by fixing one spatial and one angular dimension (sy- or xt-planes, see Figure 1).

Note that light fields can be captured using lenslet camera such as Lytro [14] or camera arrays, however we use in our experiments synthetic light fields for which the depth map ground truth is known.

2.2 Depth estimation from light fields

A 4D light field implicitly captures the 3D geometry of a scene. As illustrated in Figure 1, the depth Z of a point P in 3D space can be obtained as: $Z = -f\frac{\Delta s}{\Delta x}$, where f and Δs are respectively the focal length and the distance between camera positions, which are known parameters at the time of capture. The depth Z can thus be obtained by estimating the disparity Δx.

Multiple methods taking advantage of the existing literature in stereo disparity estimation have then been proposed to estimate depth from light fields. These techniques rely on various matching approaches to estimate the disparity between views from the light field and a reference view (often the center one). In [2], the authors proposed an accurate block-matching method reaching sub-pixel accuracy based on the Fourier phase-shift theorem. To reduce the complexity, a multi-resolution approach was proposed in [3].

To better take into account the light field structure, extensions of the previous methods have been proposed based on the analysis of texture patches sampled along the angular dimensions instead of the spatial dimensions. These angular patches, also called SCam, were first exploited in [4]. This work was further extended in [5] to be robust to occlusion. More recently, this idea was included in a global optimization framework [6] in order to obtain a dense depth map estimation.

Several techniques also exploit the light field structure through EPI, as in such images the slope of a line has a linear relationship with the depth. In [7], the slope of the epipolar lines are estimated using a structure tensor, while in [8] a spinning parallelogram operator is proposed. In [9], depth from high spatio-angular resolution light fields is obtained by first estimating high confidence depth values on the EPI edges, and then propagating this information to homogeneous regions using a fine-to-coarse approach. In [10], a sparse decomposition of the EPI is performed over a depth-based dictionary built from fixed disparities, and the scene disparity is deduced from the sparse coding coefficients. In [11], defocus and correspondence cues are obtained from the EPI by shearing the epipolar lines. The shearing angle optimizing the multiple cues response gives the slope of the epipolar lines, and thus the depth.

Note that most of the aforementioned methods require an additional regularization or optimization step, which is usually a computationally intensive global process.

2.3 Optical flow

Since Horn and Schunck's pioneer work [15] in variational optical flow estimation, many methods have been proposed exploring their idea based on energy minimization. In this subsection, we will briefly introduce the related literature in

 ISBN 978-0-9934207-2-6

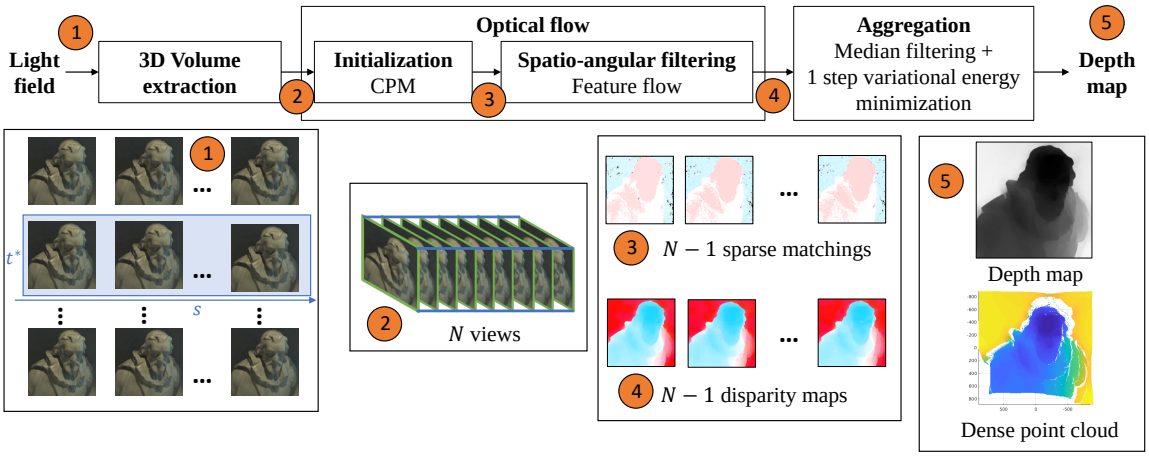


Figure 2: Overview of the proposed approach. The method can be apply on any number of rows or columns of the matrix of views in order to obtain more disparity map estimates.

this field. For a comprehensive survey, please refer to the work in [16], [17].

The original work from [15] often leads to inaccurate estimation of large pixel displacements. To improve the accuracy of such challenging cases, the patch match method [18], initially introduced for nearest neighbor field estimation, has been adapted to the optical flow problem. Patch match provides sparse pixel correspondence, and the final optical flow estimation is then considered as a labeling problem leveraging the coherent information of natural images. To further obtain a dense flow map, Revaud et al. [19] propose an edge-preserving interpolation scheme applied on top of sparse matching correspondences. In this context, Hu et al. [13] propose a coarse-to-fine extension of the basic patch match method, which is proven efficient in finding reliable correspondences on large pixel displacements.

In addition to the spatial accuracy, the optical flow temporal consistency has been an important and challenging research topic. In their early work, Murray et al. [20], proposed to add a temporal smoothness term to improve the temporal consistency. Sliding windows [21] and Kalman filtering [22] based methods have been proposed later, focusing on temporal stability, although their performances highly rely on the selection of the window size. Feature flow [12] proposed a novel local edge-aware filtering to replace the expensive global optimization used in previous work, which significantly reduced the computation cost while performing an accurate estimation. However, this method relies on a sparse correspondence initialization, which has a significant impact on the final result.

3 Depth estimation from light fields using optical flow

3.1 System overview

In this paper, we propose a novel scheme for efficient and accurate estimation of depth maps from the 4D structure of light fields using optical flow. Our approach consists of three main steps, as illustrated in Figure 2. First, a 3D spatio-angular volume is extracted from the light field by taking views along a given angular dimension. Second, an optical flow estimation is performed over this spatio-angular volume. The displacements measured by the optical flow thus correspond to disparity estimates between consecutive views of the light field. The optical flow itself consists of two steps: an initialization with a sparse matching correspondence technique [13] is first performed, then a spatio-angular edge-aware filter is applied on the sparse estimates to obtain a dense flow estimation. For this volumetric filtering we choose the feature flow method [12]. Finally, an aggregation step is performed to obtain the depth map from the multiple disparity map estimates. A common process used in most state-of-the-art methods performs sophisticated weighted median or image guided filtering combined with costly global energy minimization on the disparity map estimates to obtain a single accurate disparity map. Thanks to the the edge-aware filtering along the angular dimension of the optical flow, which enforces the consistency between the different disparity map estimates, we can reduce this step to a simple median filtering followed by a one-step variational energy minimization. For the same reason, we can propose in this paper a novel process where several depth maps are created from each disparity map estimate, and then fused in the 3D space to create a single extra dense point cloud, which enables interesting application scenarios.

 ISBN 978-0-9934207-2-6

3.2 3D spatio-angular volume extraction

To obtain a 3D spatio-angular volume, we fix one of the angular dimensions and extract N views over the remaining dimension. This volume thus consists in a sequence of sub-aperture images, noted $V = \{I_n\}, n = 1 \dots N$. Here and for the rest of this paper, we assume without loss of generality that we fix t^* and take the sub-aperture images over s (see Figure 2).

3.3 Optical flow

The optical flow method used in this paper was selected for its temporal consistency property [12]. In our context, this ensures that the different disparity estimates are consistent over the angular dimension. In addition, we propose to modify the initialization method, as described in the next section.

3.3.1 Initialization: Coarse-to-fine Patch Matching

In this paper, we propose to use a recent extension of the patch match (PM) method [18], the so-called Coarse-to-fine Patch Matching (CPM) technique as initialization [13], as it is both more efficient and accurate than the SIFT flow used in [12].

The well known PM method provides an efficient way to compute sparse matchings between a pair of images. Given two images I_1, I_2, the goal is to find for each patch $p_{1,m}$ in I_1 a corresponding patch $p_{2,m} = \mathrm{M}(p_{1,m})$ in I_2, with $m = 1 \dots M$ where M is the total number of patches in the image. Note that as we only look for sparse matches, M is much lower than the total number of pixels. The core idea of PM is to use random search and propagation between neighbors to speed up the matching search. The matching search itself is conducted as a cost function minimization:

$$\mathrm{M}(p_{1,m}) = \underset{p_{2,i}}{\operatorname{argmin}}\, C(p_{1,m}, p_{2,i}), p_{2,i} \in N_m \tag{1}$$

where the cost function $C(\cdot)$ corresponds to the sum of absolute difference (SAD) of the SIFT descriptors, and N_m is a set comprising all patches contained in a search window centered on $p_{1,m}$. Note that in our context, the sub-aperture images of a light field are rectified, and we can further reduce the complexity of the matching search by limiting the search window to an epipolar line.

The CPM method then consists in applying PM on a hierarchical architecture. A pyramid with k levels is first constructed from the original images with a downsampling factor η. This pyramidal decomposition is noted I_i^l with $i = 1, 2$ and $l = 1 \dots k$. The PM method is first applied on the I_1^k and I_2^k with a random initialization, and then iteratively on I_1^l and I_2^l with $l = k-1 \dots 1$ using the output of the previous level $l+1$ as initialization.

To initialize our optical flow, we apply the CPM method on consecutive pairs of views I_n, I_{n+1} with $n = 1 \dots N-1$ taken from the volume V built previously, and we note f_n^{init} the flow between these views.

3.3.2 Efficient spatio-angular filtering: Feature flow

Once sparse matches are obtained as described in previous section, we performed edge-aware filtering on the spatio-angular volume in order to obtain dense consistent correspondences. We introduce in this section the feature flow method [12], an efficient edge-aware filter, used to diffuse sparse matches with coherent information. One of the main advantage of the feature flow is that the global energy minimization operation used in many optical flow approaches is replaced with a local volumetric edge-aware filtering operation. To properly detect object edges in sub-aperture images and their disparity variations, a domain transform filter [23] is applied iteratively on the 3 spatio-angular dimensions with a fixed width Gaussian kernel. The flow f_n between views I_n, I_{n+1} is obtained from the flow from previous views as $f_n = G * f_{n-1}$ where G is the domain transform filter mentioned above and $*$ is a convolution operator. Note that the filtering along the angular dimension follows "disparity paths" provided by the initial sparse flow estimations f_n^{init}, with $n = 1 \dots N-1$.

To improve the accuracy of the flow estimation, the input sparse flow is weighted with a confidence map, whose weights are computed as the absolute difference between the matching correspondence vectors, thus increasing the contribution of reliable matches.

 ISBN 978-0-9934207-2-6

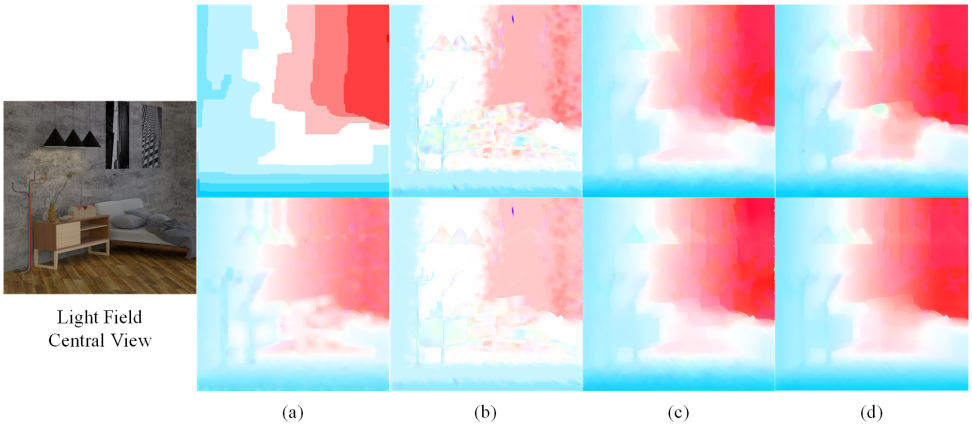


Figure 3: Comparison of optical flow obtained with state-of-the-art methods. **Top row** consists of initialization results with different optical flow methods. **Bottom row** is the results of these initializations + feature flow filter. (a) SIFT Flow [25] (4.9s); (b) EPPM [26] (GPU-based,0.7s); (c) EpicFlow [19] (15s); (d) CPM-Flow [13] (5.3s) **(Best viewed in color)**

3.4 From disparity to depth map

3.4.1 Single accurate disparity map

The first approach to obtain the final depth map is to first compute a single disparity map from all estimates, which is then converted into a depth map Z_{init} using the equation from Section 2.2. This technique is used in many state-of-the-art methods, e.g. based on weighted median or image guided filtering, which efficiently removes outliers, and is often followed by a costly global energy minimization technique.

However, thanks to the angular filtering which enforce the consistency between the different disparity map estimates, we can apply a much simpler and faster aggregation step, using median filtering and a one-step variational energy minimization. Here, the one-step variational energy minimization from the Epicflow [19] method is used to obtain the final depth map Z_f from Z_{init}:

$$Z_f = \underset{Z_i}{\operatorname{argmin}}(E_{data}(Z_i) + \lambda \alpha E_{smooth}(Z_i)) \tag{2}$$

where E_{data} corresponds to a classical color-constancy data term while E_{smooth} corresponds to a gradient-constancy function with a local smoothness term weight $\alpha = exp(-\kappa \|\nabla_2 Z\|)$ [24], where $\kappa = 5$.

3.4.2 Extra dense Point Cloud

Thanks to the consistency of the different disparity estimates, we can use a novel process to create the point cloud, where multiple point clouds are created from each disparity map, and then aggregated in the 3D space to obtain the final extra dense point cloud.

4 Evaluation

In this section, we analyze the results of the proposed approach. All our experiments were run on an Intel Core i7-6700k 4.0GHz CPU. We use the feature flow implementation from [27] and the same parameter setting for all our experiments. For the CPM method, the level of pyramids k is set to 5, the downsampling factor η is set to the 0.5 and the patch size is set to 3x3.

Evaluation of the optical flow. We evaluate here the performance of the proposed optical flow approach against state-of-the art methods. In Figure 3, we show the results of several optical flow initializations in the top row and the results after feature flow filtering in the bottom row. The volumetric filtering using feature flow along the angular dimension of the light field clearly improves the accuracy of the optical flow from any initialization method, significantly improving consistency and continuity of brightness. The proposed method using CPM as initialization achieves the best performance in terms of balance between speed and accuracy.

 ISBN 978-0-9934207-2-6

The importance of the volumetric filtering is also illustrated in Figure 6, where we show the final depth map results for several light fields obtained with our method with and without feature flow. The quality of the depth maps is clearly improved with feature flow for all sequences.

HCI benchmark performance. We evaluate here the accuracy and efficiency of our proposed method against state-of-the-art light field depth map estimation methods [2], [3], [5]–[8], [10] using the recent HCI 4D light field dataset [28] [1]. The accuracy of the depth estimation is evaluated using the Mean Square Error (MSE) * 100 and the computational complexity using the running time in seconds. The results are summed up in the graph of Figure 4, showing the average performances over the HCI dataset. More detailed results will be made available on our web page[2]. Our method achieves comparable perfomance with the best method of the state-of-the-art in terms of balance between accuracy and speed.

In addition to these objective metrics, we show the depth maps obtained from several light fields in Figure 6 and compare against state-of-the-art methods. The final comparison shows better performance for edge preservation of objects (see Cotton column) and also smoother results for noisy scenes (see Backgammon and Dino columns). However, we notice that the proposed local filtering method, which allows considerable speed up, sometime produces less smooth results than a global solution (see background of Dino and Boxes column). The feature flow filter also heavily depends on the quality of the optical flow initialization. If the optical flow method is unable to provide accurate correspondence, it can not be corrected by the filter (see for example the Boxes column).

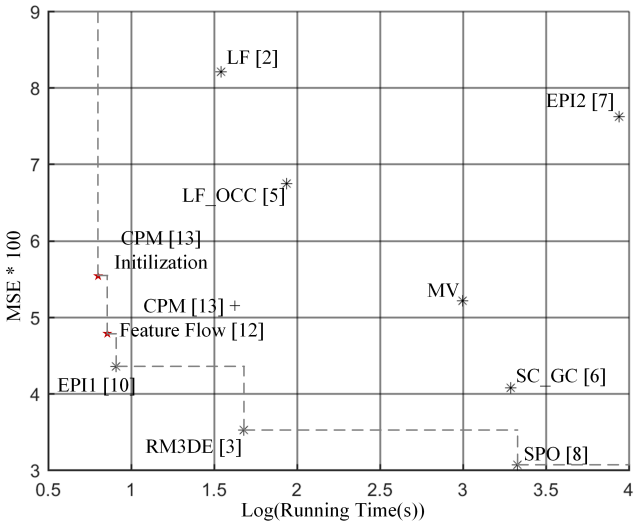


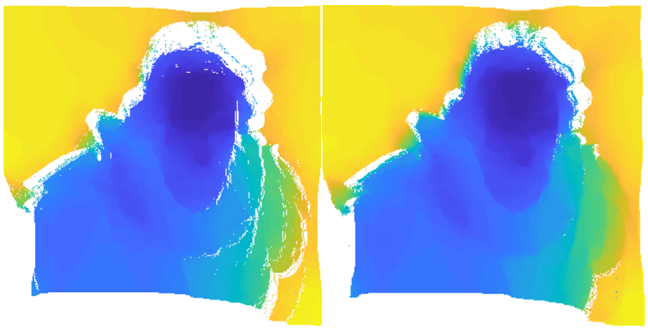
Figure 4: Comparison of our method (red stars) performances against state-of-the-art (blue stars), averaged over all HCI light fields. The results show that we achieve comparable performances to the best state-of-the-art method in terms of balance between speed and accuracy.

Note that at the time of writing, some recent results were added in the HCI benchmark without any attached publications. As we can not fully understand and exploit comparisons with methods which are not described, we choose in this paper to compare our results only against published work.

Extra dense 3D Point Cloud. As mentioned in Section 3.4, a unique feature of our method compared to the state-of-the-art approaches is that we can produce several consistent depth maps which can then be integrated in the 3D space in order to produce extra dense point clouds. Examples of such results are shown in Figure 5 in comparison with point clouds obtained with a single depth map.

5 Conclusion

In this paper, we introduced a novel optical flow-based method to estimate depth maps from light fields. We showed that by extracting a 3D volume consisting of a sequence of views from the 4D light field, and applying a temporally consistent optical flow on this spatio-angular volume, we were able to obtain high-quality depth maps with a reduced complexity. Comparison with state-of-the-art methods on the HCI benchmark showed that we are competitive with the best method in terms of balance between accuracy and speed. Furthermore, thanks to the enforced consistency of the disparity map estimates along the angular dimension, we are able to produce point clouds with a much higher density than in any state-of-the-art method. However, we note that our local filtering sometimes yield inferior results compared to global optimizations. In future work, we plan to investigate different spatio-angular filtering methods in order to improve the



(a) (b)

Figure 5: (a) Single point clouds (about 260k points) v.s. (b) Extra dense point clouds (more than 21 million points). **(Best viewed in color)**

[1] http://hci-lightfield.iwr.uni-heidelberg.de/
[2] https://v-sense.scss.tcd.ie/?p=842

 ISBN 978-0-9934207-2-6

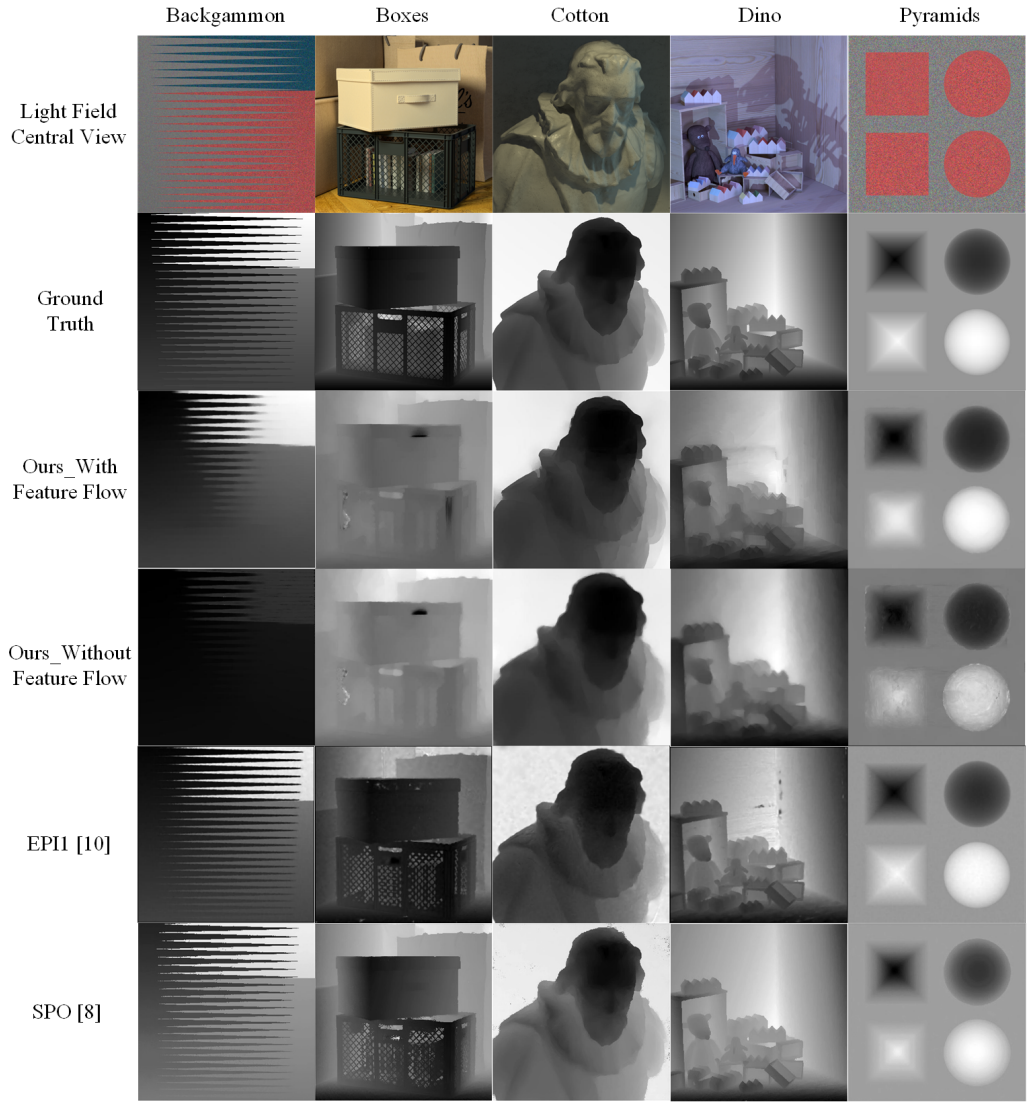


Figure 6: Depth map comparison on HCI dataset

accuracy while keeping a faster running time. Furthermore, we intend to apply our method on dense light fields captured with lenslet cameras such as Lytro [14] to perform 3D reconstruction of real world scenes.

References

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '96, 1996, pp. 31–42.

[2] H. G. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y. W. Tai, and I. S. Kweon, "Accurate depth map estimation from a lenslet light field camera," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June, pp. 1547–1555, 2015.

[3] A. Neri, M. Carli, and F. Battisti, "A multi-resolution approach to depth field estimation in dense image arrays," *Proceedings - International Conference on Image Processing, ICIP*, vol. 2015-Decem, pp. 3358–3362, 2015.

[4] C. Chen, H. Lin, Z. Yu, S. B. Kang, and J. Yu, "Light field stereo matching using bilateral statistics of surface cameras," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1518–1525.

[5] T.-C. Wang, A. A. Efros, and R. Ramamoorthi, "Occlusion-Aware Depth Estimation Using Light-Field Cameras," *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 3487–3495, 2015.

[6] L. Si and Q. Wang, "Dense depth-map estimation and geometry inference from light fields via global optimization," in *Computer Vision – ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part III*. Springer International Publishing, 2017, pp. 83–98.

[7] S. Wanner and B. Goldluecke, "Globally consistent depth labeling of 4D light fields," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2012, pp. 41–48.

[8] S. Zhang, H. Sheng, C. Li, J. Zhang, and Z. Xiong, "Robust depth estimation for light field via spinning parallelogram operator," *Computer Vision and Image Understanding*, vol. 145, pp. 148–159, 2016.

[9] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross, "Scene reconstruction from high spatio-angular resolution light fields," *ACM Trans. Graph.*, vol. 32, no. 4, 73:1–73:12, Jul. 2013.

[10] O. Johannsen, A. Sulc, and B. Goldluecke, "What Sparse Light Field Coding Reveals about Scene Structure," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2016, pp. 3262–3270.

[11] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi, "Depth from combining defocus and correspondence using light-field cameras," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2, pp. 673–680, 2013.

[12] M. Lang, O. Wang, T. Aydin, A. Smolic, and M. Gross, "Practical temporal consistency for image-based graphics applications," *ACM Transactions on Graphics (ToG)*, vol. 31, no. 4, p. 34, 2012.

[13] Y. Hu, R. Song, and Y. Li, "Efficient coarse-to-fine patchmatch for large displacement optical flow," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5704–5712.

[14] *The lytro illum camera*, `https://www.lytro.com/imaging`, accessed: 23-05-2017.

[15] B. Horn and B. Schunck, "Determining optical flow," *Artificial intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.

[16] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *International Journal of Computer Vision*, vol. 92, no. 1, pp. 1–31, 2011.

[17] W. Li, Y. Chen, J. Lee, G. Ren, and D. Cosker, "Robust optical flow estimation for continuous blurred scenes using rgb-motion imaging and directional filtering," in *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*, IEEE, 2014, pp. 792–799.

[18] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," *ACM Transactions on Graphics-TOG*, vol. 28, no. 3, p. 24, 2009.

[19] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Epicflow: Edge-preserving interpolation of correspondences for optical flow," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1164–1172.

[20] D. W. Murray and B. F. Buxton, "Scene segmentation from visual motion using global optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 2, pp. 220–228, 1987.

[21] S. Volz, A. Bruhn, L. Valgaerts, and H. Zimmer, "Modeling temporal coherence for optical flow," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, IEEE, 2011, pp. 1116–1123.

[22] M. Hoeffken, D. Oberhoff, and M. Kolesnik, "Temporal prediction and spatial regularization in differential optical flow," in *International Conference on Advanced Concepts for Intelligent Vision Systems*, Springer, 2011, pp. 576–585.

[23] E. S. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing," in *ACM Transactions on Graphics (ToG)*, ACM, vol. 30, 2011, p. 69.

[24] L. Xu, J. Jia, and Y. Matsushita, "Motion detail preserving optical flow estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1744–1757, 2012.

[25] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. Freeman, "Sift flow: Dense correspondence across different scenes," *Computer vision–ECCV 2008*, pp. 28–42, 2008.

[26] L. Bao, Q. Yang, and H. Jin, "Fast edge-preserving patchmatch for large displacement optical flow," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3534–3541.

[27] J. S. Roo and C. Richardt, "Temporally coherent video de-anaglyph," in *ACM SIGGRAPH 2014 Posters*, ACM, 2014, p. 75.

[28] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A dataset and evaluation methodology for depth estimation on 4d light fields," in *Asian Conference on Computer Vision*, Springer, 2016.

Recognising Fine-Grained Actions by Combining Colour, Depth and Flow

Seán Bruton, Gerard Lacey

School of Computer Science and Statistics
Trinity College Dublin

Abstract

Given the availability of low-cost depth cameras, and the advances made in image recognition through the utilisation of deep convolution neural networks, in this work we explore possibilities for recognising such fine-grained actions through combinations of different image modalities, afforded by depth cameras, including depth and scene flow. Recognition of fine-grained actions, such as those involved in preparing a meal, allows for a number of potential situational support systems to be developed. This work also addresses a problem of some current deep learning approaches to action recognition, whereby it is necessary to train multiple parallel convolutional neural networks for each image modality, such as colour and optical flow. The approach performs an early fusion of data modalities using separable convolution filters, based on the observation that these filters separate the learning of channel-wise features and inter-feature relationships. The recognition results of this fusion approach outperform those of single modalities showing that this early fusion is an effective technique for increasing recognition accuracy of fine-grained actions without the need for training multiple parallel networks.

Keywords: Action Recognition, RGB-D, Convolutional Neural Networks

1 Introduction

With the proliferation of embedded cameras and sensors, there exists potential to utilise these sensors to build applications which can provide monitoring and assistance services for many everyday activities. Such applications may include smart assistance and monitoring systems that provide cognitive situational support for the elderly in performing activities of daily living (ADL). Certain of these activities are composed of a number of subtle interactions with objects. Recognising these actions at a granular level would lead to a more detailed understanding of these activities, and hence provide greater opportunities to provide assistance.

Reliable recognition of these actions is a challenging problem due to a number of factors. One of the principal factors is the complexity of the actions, which are subtle bi-manual interactions with multiple objects. Difficulty also lies in the fact that there are a variety of valid approaches to completing tasks, due to factors such as the handedness of the person. Another complicating factor is that as a result of these actions, the objects being manipulated may change shape and form. These reasons make recognition of fine-grained actions challenging using techniques based on semantic tracking of hands and objects.

For the general problem of action recognition, recent methods [Donahue et al., 2015] often utilise motion information in the form of optical flow to discriminate between different action classes. The importance of high quality optical flow information to learning accurate action models has been shown by Varol et al [Varol et al., 2017]. There exists an analogue of two-dimensional optical flow information for three dimensions, known as scene flow, which extends the flow information to three spatial dimensions.

In this work, we propose the use of scene flow information for the recognition of fine-grained actions in RGB-D data. We leverage a recent method to calculate scene flow information [Jaimez et al., 2015] that utilises

 ISBN 978-0-9934207-2-6

the primal-dual algorithm. The advantage of this method is that it can be parallelised across GPU compute cores to achieve real-time rates, an aspect that makes this method viable as part of a real-time monitoring system.

Many recent methods that perform the general problem of action recognition for colour data, train parallel convolutional neural networks for the colour information and the optical flow information. These deep learning approaches have the benefits of being able to perform inference rapidly, without the need for any semantic tracking of persons or objects, using only the colour and flow information as input features to the network. However, there are a number of drawbacks to this approach. One such drawback is that the training of multiple networks does not scale well when there are a larger number of input image types. Another drawback of this approach is that jointly located information in different modalities may aid in the learning process. For example, in the activity of preparing a meal, complex actions may include chopping and mixing ingredients, with different action classes when the action is performed on different ingredients. It would be difficult to classify these actions based on colour information alone, devoid of the spatio-temporal information, and furthermore, it would be difficult to determine these actions based on motion alone, without being able to identify the ingredient the action is being performed on. Based on this rationale, the co-locality of motion and colour features may aid learning of such joint features. In this work, we explore a potential technique for learning these features jointly via depth-wise separable convolutional features.

2 State of the Art

In this section, we review the latest research as it relates to the problem of fine-grained action recognition. Specifically, we focus on techniques to calculate scene flow information and action recognition in RGB-D data.

2.1 Scene Flow

Scene flow can be defined as the dense or semi-dense field of three-dimensional non-rigid motion of a scene between two distinct times [Vedula et al., 1999]. Early methods of estimating scene flow utilised images from stereo pairs or from moving cameras to estimate scene structure alongside scene flow [Wedel et al., 2008]. However, in recent years, with the availability of consumer depth cameras, there has been interest shown in calculating scene flow based on RGB-D data [Gottfried et al., 2011, Sun et al., 2015]. The primal dual algorithm is utilised in a recent approach [Jaimez et al., 2015], called PD-Flow, to minimise an energy formulation of geometric and luminosity constraints. The advantage of this method over other recent techniques is that the formulation can be efficiently solved using parallelisation techniques. In approaching the problem of developing a situational support action recognition system, it is essential that the actions can be classified at a rate that would allow the system to provide prompts or alerts with minimum delay. For this reason, the PD-Flow method was selected to calculate scene flow using RGB-D data. The authors of this PD-Flow approach provide code which allows for the calculation of scene flow at real-time rates when parallelised on a GPU.

2.2 Action Recognition

2.2.1 2D Action Recognition using Flow Information

Due to the fact that it is a representation of spatio-temporal information, optical flow is pervasive in its use for recognising human actions from video. It has been used as part of global feature approaches, such as histograms of flow, and as part of local feature based approaches such as dense trajectories [Wang and Schmid, 2013], where it is utilised to match local features between frames. In recent deep learning approaches to action recognition, it has also been shown to contain discriminative information [Donahue et al., 2015]. These approaches rely on two parallel convolutional neural networks, the outputs of which are fused through techniques weighting, discriminative classifiers, or through further neural network layers. This imposes a heavy training and computational cost and does not take account of any co-located features across the modalities that may aid in classification. [Feichtenhofer et al., 2016] looked at more optimal techniques for fusion, reporting that fusion at

 ISBN 978-0-9934207-2-6

the latter convolutional layers improved classification results, whilst somewhat reducing the number of parameters of the network. However, the problem of training a network on further image modalities (such as depth, infra-red etc.), without the need for expensive parallel networks, has yet to be tackled.

2.2.2 RGB-D Action Recognition

Many of the techniques for recognition of human action in RGB-D have focussed on learning from pose information. Research has also been focussed on extracting features from depth information such as kernel descriptors [Kong et al., 2015] or occupancy patterns [Oreifej and Liu, 2013]. Work that involved recognition across both colour and depth modalities [Kong and Fu, 2015] observed that there exist latent structures across modalities, and hence by learning how to fuse these modalities a joint representation can give better classification results. Based on these observations, our proposed early fusion of different modalities of image, such as depth and colour, in a deep learning architecture may enable discovery of these joint structures.

2.2.3 Fine-grained Action Recognition

Techniques for recognition of fine-grained actions have relied on different information, such as data extracted from embedded accelerometers [Stein and McKenna, 2013], human pose information [Rohrbach et al., 2015] to object region proposals [Zhou et al., 2015]. A deep learning based technique for fine-grained action recognition [Singh et al., 2016] utilised parallel streams of colour and optical flow over a large temporal window to recognise actions, utilising tracking information to train parallel networks on bounding boxes of areas of motion.

We have yet to observe techniques that utilise scene flow information for the task of fine-grained action recognition as part of a deep learning framework. In this work we examine the applicability of such information and propose early fusion of these different image features, including depth, in a deep network architecture.

3 Method

In this section, we describe each of the components that make up the pipeline for recognising fine-grained actions. We include the algorithm used to calculate scene flow [Jaimez et al., 2015] to ensure that the pipeline description is self-contained.

3.1 PD-flow

A scene flow motion field, $\mathbf{M} : (\Omega \in \mathbb{R}^3) \rightarrow \mathbb{R}^3$, is defined over an image domain, Ω, with respect to a camera reference frame. For each pixel with non-zero depth, $\mathbf{M}$ can be expressed in terms of typical optical flow, u and v, and range flow, w as

$$\mathbf{M} = \Gamma(\mathbf{s}) = \begin{pmatrix} \frac{Z}{f_x} & 0 & \frac{X}{Z} \\ 0 & \frac{Z}{f_y} & \frac{Y}{Z} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix} \tag{1}$$

where $\mathbf{s}^T = (u, v, w)$ and $\Gamma : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ is bijective. The elements of Γ are known quantities, where X, Y and Z are the spatial coordinates of an observed point, and f_x and f_y are the focal lengths of the depth camera. Hence, estimating the scene flow motion field is equivalent to the estimation of the optical flow, u and v, and the range flow, w. The problem of estimating the scene flow can thus be structured as a minimisation problem over $\mathbf{s}$ with penalties to ensure photometric and geometric consistency of the solution, $\min_{\mathbf{s}} \{E_D(\mathbf{s}) + E_R(\mathbf{s})\}$.

Here, the data term, E_D, represents the matching penalty of the intensity and the depth between successive images,

$$E_D(\mathbf{s}) = \int_\Omega \left|\varrho_I(\mathbf{s}, x, y)\right| + \mu(x, y) \left|\varrho_Z(\mathbf{s}, x, y)\right| dxdy \tag{2}$$

 ISBN 978-0-9934207-2-6

where $\varrho_I(\mathbf{s}, x, y) = I_0(x, y) - I_1(x+u, y+v) = 0$ represents the difference in intensity at points in the source image, I_0, and target image, I_1. Similarly for depth, $\varrho_Z(\mathbf{s}, x, y) = Z_0(x, y) - Z_1(x+u, y+v) - w = 0$ takes account of the change in depth, w, between depth frames Z_0 and Z_1. A weighting function, $\mu(x, y)$ is included that weighs depth consistency against brightness consistency.

The regularisation term is included to ensure smooth flow output and to overcome the aperture problem,

$$E_R(\mathbf{s}) = \lambda_I \int_\Omega \left| \left(r_x \frac{\partial u}{\partial x}, r_y \frac{\partial u}{\partial y} \right) \right| + \left| \left(r_x \frac{\partial v}{\partial x}, r_y \frac{\partial v}{\partial y} \right) \right| dxdy + \lambda_D \int_\Omega \left| \left(r_x \frac{\partial w}{\partial x}, r_y \frac{\partial w}{\partial y} \right) \right| dxdy \tag{3}$$

where λ_I, λ_D are constant weights that can be tuned accordingly and $r_x = \left(\frac{\partial X}{\partial x}^2 + \frac{\partial Z}{\partial x}^2 \right)^{-\frac{1}{2}}$, $r_y = \left(\frac{\partial Y}{\partial y}^2 + \frac{\partial Z}{\partial y}^2 \right)^{-\frac{1}{2}}$, where X, Y and Z are the cartesian spatial directions.

The linearity of the data term and the convexity of the total variation regularisation term renders the formulation convex. A primal dual formulation can be solved iteratively and in parallel for each pixel on a GPU.

(a) Colour (b) Depth (c) Scene Flow

Figure 1: A visualisation of the different data modalities used to recognise actions, using an RGB-D frame from the 50 Salads dataset [Stein and McKenna, 2013] with action label *mix ingredients*. Appearing and disappearing depth points can cause incorrect flow estimation, as seen in the bottom-right of scene flow image (c).

3.2 Deep Learning Architecture

The architecture of the deep network employed here to recognise actions is composed of an early fusion component and of a base convolutional network.

Data Fusion Given the availability of multi-modal data from RGB-D cameras, the question remains of how to combine this information in a deep network. Previous methods of fusion have included weighting the classification scores of independent networks that have been trained separately [Donahue et al., 2015]. However, this ignores the possibility of important features being jointly located in colour and depth images. In a usual convolutional layer, the network is tasked with learning convolutional filters that operate across all input channels. Thus the kernels will need to recognise channel-wise features and inter-channel relationships.

An approach that separates these learning tasks is to utilise separable convolution [Sifre and Mallat, 2014]. A separable convolution works by first performing depth-wise 2D convolutions for each image channel. The outputs of this convolutional stage are then combined by a 1x1 convolution, which attempts to learn a linear relationship between the individual pixel outputs of the previous convolutional layer. It is also possible to perform non-linear activation prior to this step. This allows for the task of learning independent features per image channel to be segmented from the task of learning the relationships between these features. The total number of parameters of this separable depth-wise convolutional layer can be less than that of the a normal convolution, dependant on the number of depth-wise features, dm, in Figure 2.

Convolutional Neural Network As the base convolutional neural network, Figure 3, we use a variation of a residual networks, the Wide Residual Network [Zagoruyko and Komodakis, 2016]. At the time of writing, this network achieves state of the art results on the CIFAR10 and CIFAR100 datasets.

 ISBN 978-0-9934207-2-6

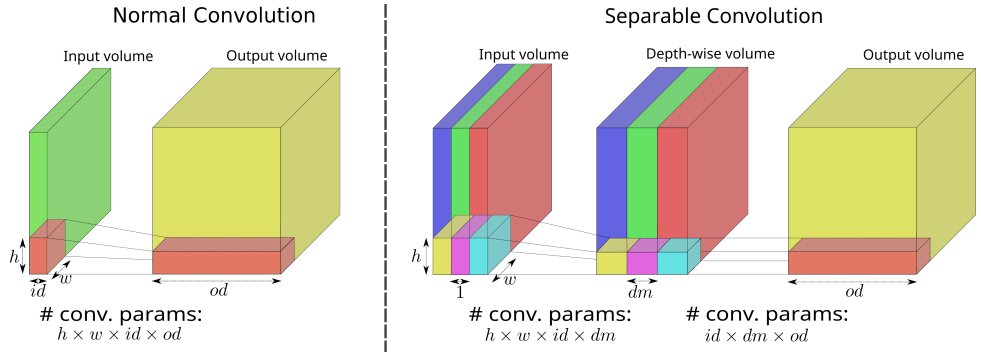


Figure 2: Separable convolution consists of a depth-wise convolution stage followed by a point-wise (1×1) convolution. There are two sets of parameters for separable convolution, the first set is tasked with learning spatial features in each input channel, and the second set is tasked with learning a linear relationship between these depth-wise feature responses.

The outputs of the fusion stage are fed as inputs to this network. The final pooling layer of the network is reduced to a 4×4 pooling as it was found that preserving some spatial information increased classification performance. The output of this pooling layer are fed into a fully connected layer with Relu non-linear activation, and then into a final fully connected layer, with soft-max activation.

This architecture was chosen to provide a baseline performance of single image recognition for different input image types and to examine the performance of the proposed early fusion of these image types via separable convolution. It is adaptable, for example by adding a recurrent layer to the end of the network, to multiple frame inputs to add further temporal information besides that contained in a single scene flow image.

4 Evaluation

4.1 Dataset

There exists a dearth of RGB-D fine-grained action recognition datasets in the literature. The overall aim of the research is to be able to recognise complex actions that involve fine motions and multiple non-rigid objects. An example of a dataset that does exist that matches this criterion is the 50 Salads dataset [Stein and McKenna, 2013]. This dataset consists of 50 RGB-D videos of people preparing a salad in a kitchen environment. It captures 25 different subjects preparing two types of salads in over 4 hours of video. The dataset is labelled at multiple levels of granularity. The granularity at which we are attempting to classify is composed of 18 actions, such as *add oil*, *cut cucumber*, *cut tomato*, *peel cucumber* and *place cheese into bowl*. These actions exhibit similar subtle motions and similar appearance, making them challenging to distinguish. The scene flow images are calculated using the PD-Flow algorithm and the features are stacked along with the colour and depth images. For each of the experiments, the mean training set image is calculated and subtracted from each sample. The samples are also divided by a constant value of half of the largest value encountered in the mean image. This ensures that the data is centred on the origin and mostly lies in the range $[-1, 1]$. A common optimisation is to augment the data in a plausible manner (such as affine transformations and re-colouring) to produce an effectively larger

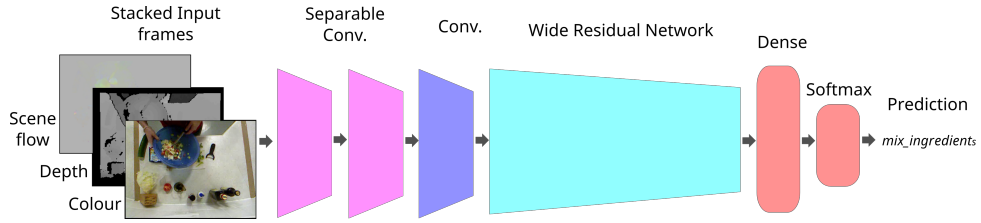


Figure 3: The architecture of the network involves a number of convolutional stages. The first two separable convolutional stages are responsible for the fusion of the features. Following this, there is a convolutional stage before the Wide Residual Network of depth 12.

 ISBN 978-0-9934207-2-6

dataset. The RGB-D images are first downsampled from 640x480 to 320x240 and the scene flow images are calculated at this level. All samples are further downsampled to 160x120. In these experiments, each sample is a central cropping, of size 112x112, shifted by offsets drawn from a uniform distribution on $[-2, 2]$.

4.2 Classifier Training

In order to train our network, a number of different hyperparameters were required to be tuned. The base Wide Residual Network model foregoes any widening in favour of a larger batch size which improved classification results when constrained by GPU memory requirements for processing images of size 112x112.

The loss function to minimise was the categorical cross-entropy loss, also known as the Kullback-Leibler divergence. Given an output probability mass function, $q^i_Y(y)$, across the space, C, of class labels, and the predicted probability mass function, $p^i_Y(y)$, for a frame i, the cross entropy between $p^i_Y(y)$ and $q^i_Y(y)$ is

$$D\left(p^i_Y(y), q^i_Y(y)\right) = -\sum_{c \in C} p^i_Y(c) log\left(q^i_Y(c)\right). \tag{4}$$

The entire neural network can be expressed as a nonlinear differentiable function, $p^i_Y(y) = f\left(\mathbf{X}^i; \{\mathbf{W}_k\}, \{\mathbf{b}_l\}\right)$, where $\{\mathbf{W}_k\}$ and $\{\mathbf{b}_l\}$ are the sets of convolutional weights and biases contained in the network, respectively, and $\mathbf{X}_i$ represents the input image tensor for frame i.

We can concatenate all of the weights and the biases into single vectors, $\mathbf{w}$ and $\mathbf{b}$, respectively, leading to the training set cross-entropy loss to be formulated as

$$J(\mathbf{w}, \mathbf{b}) = \frac{1}{N} \sum_{i \in V} D\left(f(\mathbf{X}^i; \mathbf{w}, \mathbf{b}), q^i_Y(y)\right) + \lambda ||\mathbf{w}||, \tag{5}$$

where V is the set of all frames in the dataset, and $|V| = N$, and we impose a regularization loss controlled by λ. In training, we calculate the loss for a small subset, $B \subset V$, of the training set.

We update the weights and the biases by calculating the gradient of the loss with respect to each parameter and updating them with the equation $\mathbf{w}_i \leftarrow \mathbf{w}_i - \alpha \nabla J(\mathbf{w}, \mathbf{b})$, where the gradient is calculated using the back propagation algorithm.

In order to better monitor the accuracy during training, updates were bundled up into mini-epochs corresponding to 0.04 of the training and the test sets (amounting to 73 training updates per mini-epoch with a batch size of 256). A carefully managed learning rate schedule was used in conjunction with the Stochastic Gradient Descent optimization technique, using Nesterov momentum with a momentum value of 0.9. The learning rate begins at 0.1 and undergoes decay rate of 50 after 75 mini-epochs, and after every ensuing 25 mini-epochs. The same learning schedule was used across each of the experiments. All weights in the network are initialised with He initialisation as per the Wide Residual Network. No biases are used in the convolutional network. To regularise the network an L_2 penalisation value of 0.0005 was used. Dropout was further used to regularize the training of the network. Specific neurons were dropped with a probability of 0.4 between each of the fully connected layers and with a probability of 0.5 after the Relu non-linearity in the residual blocks. The Tensorflow framework was used for all training for its ability to define a network using a simple declarative syntax.

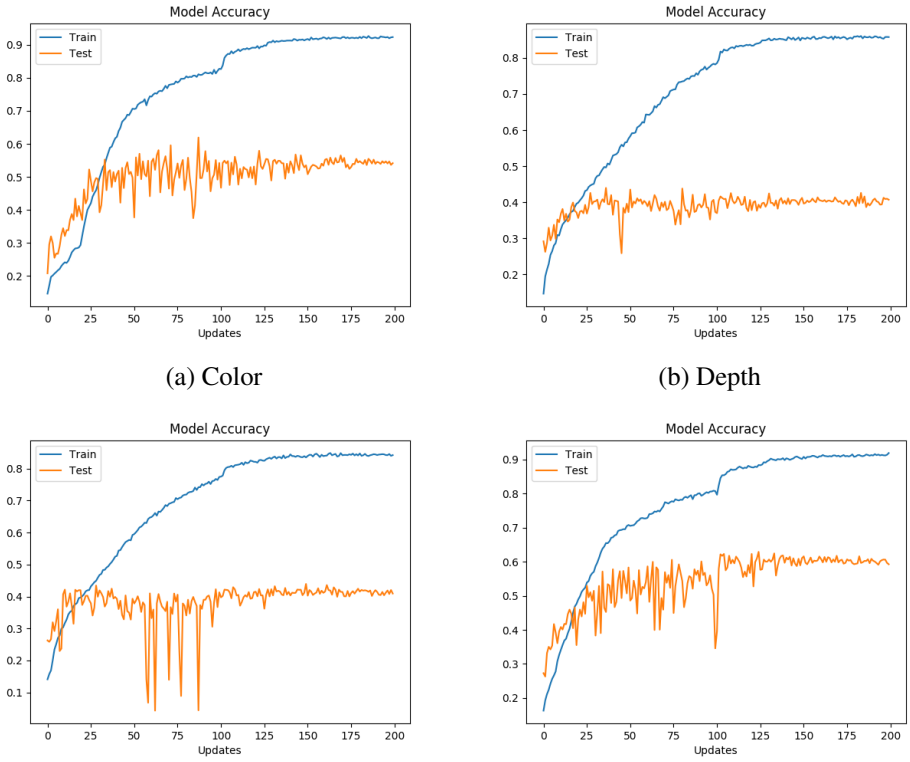
4.3 Results

Figure 4 shows the training profile across the different image feature types as well as across the early fusion method. The accuracy reported is the proportion of correctly classified individual RGBD frames contained in the test split, composed of videos of 5 subjects' performances. It can be seen in Table 1 that the fusion technique outperforms the next best classification accuracy by 5.5%, indicating that the fusion technique is effectively learning joint relationships between colour, depth and scene flow.

Features	Colour	Depth	PD-Flow	Fused
Per-frame accuracy (%)	54.7	40.4	41.4	60.2

Table 1: The learning rate schedule.

 ISBN 978-0-9934207-2-6



(a) Color (b) Depth

(c) PD-Flow (d) Fused

Figure 4: The accuracy during training across the different features and early fusion methods. It can be seen that the separable convolution method outperforms the other fusion methods and suffers less from over-fitting. The x axis represents the number of mini-epochs, equivalent to 0.04 of a full training set epoch.

5 Discussion

In this work, we have shown that it is possible to utilise separable convolution to perform an effective early fusion technique for multi-modal convolutional neural network learning tasks that would typically be performed by training multiple parallel networks. Due to the fact that fine-grained actions are closely reliant on motion and colour information, we argue that early fusion is reasonable in this case. Furthermore, in the case of multiple image modalities, this framework is appealing due to the reduction in parameters and the obviation of the training of multiple separate networks.

In future work, we wish to extend the baseline architecture to take account of temporal information beyond that contained in the scene flow information. This may be achieved by using this architecture to extract features from single multi-modal frames and passing sequences of these features as inputs to a recurrent neural network. Furthermore, we wish to explore potential techniques of fusing the different image types, such as utilising residual blocks to perform the separable convolutional fusion. Techniques of generating synthetic training data, via plausible transformations or adversarial networks, will be explored to improve classification performance.

References

[Donahue et al., 2015] Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., and Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description. pages 2625–2634.

[Feichtenhofer et al., 2016] Feichtenhofer, C., Pinz, A., and Zisserman, A. (2016). Convolutional two-stream network fusion for video action recognition. pages 1933–1941.

[Gottfried et al., 2011] Gottfried, J.-M., Fehr, J., and Garbe, C. S. (2011). Computing range flow from multi-modal kinect data. In *Advances in Visual Computing*, pages 758–767. Springer, Berlin, Heidelberg.

[Jaimez et al., 2015] Jaimez, M., Souiai, M., Gonzalez-Jimenez, J., and Cremers, D. (2015). A primal-dual framework for real-time dense rgb-d scene flow. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 98–104.

[Kong and Fu, 2015] Kong, Y. and Fu, Y. (2015). Bilinear heterogeneous information machine for rgb-d action recognition. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1054–1062.

[Kong et al., 2015] Kong, Y., Satarboroujeni, B., and Fu, Y. (2015). Hierarchical 3d kernel descriptors for action recognition using depth sequences. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 1, pages 1–6.

[Oreifej and Liu, 2013] Oreifej, O. and Liu, Z. (2013). Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 716–723.

[Rohrbach et al., 2015] Rohrbach, M., Rohrbach, A., Regneri, M., Amin, S., Andriluka, M., Pinkal, M., and Schiele, B. (2015). Recognizing fine-grained and composite activities using hand-centric features and script data. *International Journal of Computer Vision*, pages 1–28.

[Sifre and Mallat, 2014] Sifre, L. and Mallat, P. S. (2014). *Rigid-Motion Scattering For Image Classification, Ecole Polytechnique, CMAP PhD thesis*.

[Singh et al., 2016] Singh, B., Marks, T. K., Jones, M., Tuzel, O., and Shao, M. (2016). A multi-stream bi-directional recurrent neural network for fine-grained action detection. pages 1961–1970.

[Stein and McKenna, 2013] Stein, S. and McKenna, S. J. (2013). Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 729–738, New York, NY, USA. ACM.

[Sun et al., 2015] Sun, D., Sudderth, E. B., and Pfister, H. (2015). Layered rgbd scene flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 548–556.

[Varol et al., 2017] Varol, G., Laptev, I., and Schmid, C. (2017). Long-term temporal convolutions for action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1.

[Vedula et al., 1999] Vedula, S., Baker, S., Collins, R., Kanade, T., and Rander, P. (1999). Three-dimensional scene flow. In *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, ICCV '99, pages 722–, Washington, DC, USA. IEEE Computer Society.

[Wang and Schmid, 2013] Wang, H. and Schmid, C. (2013). Action recognition with improved trajectories. In *2013 IEEE International Conference on Computer Vision (ICCV)*, pages 3551–3558.

[Wedel et al., 2008] Wedel, A., Rabe, C., Vaudrey, T., Brox, T., Franke, U., and Cremers, D. (2008). Efficient dense scene flow from sparse or dense stereo data. In *Computer Vision - ECCV 2008*, pages 739–751. Springer, Berlin, Heidelberg.

[Zagoruyko and Komodakis, 2016] Zagoruyko, S. and Komodakis, N. (2016). Wide residual networks. *arXiv:1605.07146 [cs]*.

[Zhou et al., 2015] Zhou, Y., Ni, B., Hong, R., Wang, M., and Tian, Q. (2015). Interaction part mining: A mid-level approach for fine-grained action recognition. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3323–3331.

A Tale of Two Losses: Discriminative Deep Feature Learning for Person Re-Identification.

Alessandro Borgia[1,2], Yang Hua[3,4], Neil M. Robertson[3,4]

[1] *ISSS/EPS, Heriot-Watt University, UK*
[2] *SIP-JRI, University of Edinburgh, UK*
[3] *ECIT, Queen's University Belfast, UK*
[4] *AnyVision, Belfast, UK*

Abstract

The changing camera viewpoint on full-body pedestrians in a multi-camera scenario may be problematic to handle, above all if the fields of view are non-overlapping. A direct effect of the viewpoint variability is that a pair of images of the same person shot by different cameras may appear to be more distant from each other in the feature space than one of them from an image of a different identity captured by the same camera. In order to tackle this problem, we propose to train a state-of-the-art CNN by two new loss functions that jointly increase the inter-class discriminative power of the deep features and their intra-class compactness. In particular, one loss function promotes the aggregation of the feature points around the centres of the view they belong to, within the scope of their own identity. The second loss encourages to push away from each other the feature clusters corresponding simultaneously to different views and different identities. Under the supervision of the two new objectives we achieve state-of-the-art accuracy with ResNet50 on Market-1501 and CUHK03 datasets, beating the performance of the softmax loss.

Keywords: Person re-id, Loss function, Multi-camera net, Changing viewpoint, Discriminative deep features.

1 Introduction

In this paper we investigate the problem of how the changing viewpoint in a multi-camera network with non-overlapping fields of view affects the performance of the person re-identification task. Traditionally, the person re-id problem is tackled using three kinds of approaches: **(1)** *feature design*, dealing either with hand-crafted feature modelling [Zhao et al., 2013, Zhao et al., 2014] or with deep feature extraction approaches [Varior et al., 2016b, Xiao et al., 2016b, Xiao et al., 2016a]. Hybrid solutions are also possible [Wu et al., 2016]. **(2)** *metric learning* [Yi et al., 2014a, Chen et al., 2012, Kulis, 2013, Hoffer and Ailon, 2015], that relies on learning a distance function tuned to a particular task in the feature space. **(3)** *side information*, that is hidden information encoded into the input data. Common strategies include target alignment and full-body pose estimation [Zheng et al., 2017, Pishchulin et al., 2016]. In some cases, strategies of different kind are used together as in [Bak et al., 2015] where target alignment and pose estimation are combined with metric learning to produce better performing pose-driven metric learning schemes. Our method belongs to the first group, embracing the deep learning (DL) approach, by virtue of the relevant impact that it has taken in person re-id. In order to enhance the discriminative power of the deep features, we assert the importance of *influencing the construction itself of the feature space*: we aim to do that by a training loss capable of reproducing the semantic similarity of images in the input space. This is different from using a metric learning approach where a metric is learned in the feature space already generated. The target of making the features extracted by a Convolutional Neural Network (CNN) more discriminative has already been pursued by [Wen et al., 2016] in face recognition. It proposes a centre loss function that, promotes the compactness of the clusters of features around the centre point of each face, thus achieving state-of-the-art results on many benchmarks.

 ISBN 978-0-9934207-2-6

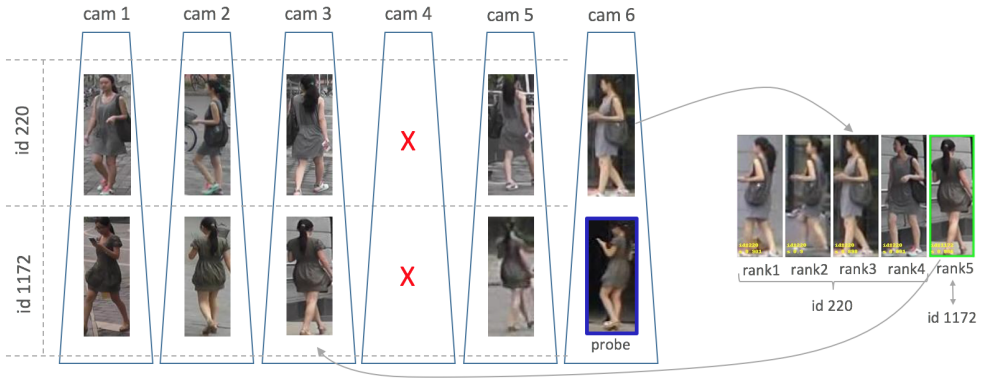


Figure 1: Illustration of the changing viewpoint problem in the multi-camera setting (Market-1501, 6 cameras) when re-id is addressed as a ranking problem. We can see that the image ranked 3^{rd} (which is a false positive since it belongs to a different identity to the probe's) is ranked by the CNN higher than the first right match (ranked only 5^{th}). This happens because the false positive is captured under the same field of view of the probe (cam.6), differently from what happens for the true positive (cam.3).

Respect to face recognition, the person re-identification task, though, is somewhat different: person re-id performance are heavily affected by deceptive viewpoint changes due to the disjoint nature of the multi-camera network for pedestrians, which we do not observe in face recognition. The viewpoint variability may cause images of two different pedestrians observed under the same camera to look (and be ranked as) more similar between each other than respect to their correspondent shots taken under other views. In order to disentangle this kind of ambiguity in person re-id, the capability we need to equip our CNN with is *learning intra/inter-camera relationships* and exploiting them to discriminate better on the base of pedestrian identity. Hence, producing highly discriminative features is our ultimate goal that we pursue by introducing new training objectives. A direct application of the centre loss to the person re-id problem, would not be that beneficial because it does not take into account the information of the field of view under which pedestrian images are captured, which is available in the person-re-id datasets. Starting from this observation, we propose to adapt the centre loss for person re-id by introducing two new loss functions, the *intra-Group Centre Loss* (**intra-GCL**) and the *inter-Group Centre Loss* (**inter-GCL**). The first one represents a direct extension of the centre loss that exploits the field of view information: it penalizes a large distance of each feature point from its related centre point but, instead of considering only one centre point per identity (like centre loss does). It refers as many centres per identity as the number of views available for one identity. Hence, it encourages the intra-identity view-based feature clusters to be compact. On the other side, the second loss function encourages a large inter-subclass separation of the view-field-based clusters belonging to different identities respect to the inter-subclass separability within a single identity, which is aligned with the requirement of person re-id evaluation. The main contributions of this paper can be summarized in the following:

– We propose two new loss functions for learning discriminative deep features that prove to be effective in mitigating the changing viewpoint problem, in the multi-camera setting with disjoint fields of view.

– We achieve state-of-the-art performance on Market-1501 and CUHK03, without employing extra training data (like by training on multiple-datasets as in [Xiao et al., 2016a]) and other side information.

2 Related Work

Hand-Crafted and Hybrid Feature. Before the DL approach became popular, the field of person re-id was dominated by approaches based on designing hand-crafted features that defined the state-of-the-art. LAB colour histograms are extracted from image patches in [Zhao et al., 2013, Zhao et al., 2014] and combined with SIFT

 ISBN 978-0-9934207-2-6

descriptor as a complementary feature. A Bag-of-Words (BoW)-based descriptor is used in [Zheng et al., 2015] in order to bridge the gap between person re-id and image search. Some strengths of this method are that it well accommodates local features and enables fast global feature matching.

Deep Features. Recently, lots of works addressing person re-identification have adopted the DL paradigm, exploiting the availability of new large-scale datasets like Market-1501 ([Zheng et al., 2015]) and CUHK03 ([Wang, 2014]). [Yi et al., 2014b] is the first work to apply DL to person re-id by designing a "siamese" CNN for deep metric learning relying on the cosine similarity as connecting function. Later approaches, following the same direction as [Varior et al., 2016a, Varior et al., 2016b], confirm the success of the siamese CNNs intuitively due to the fact that they force to learn the relationship existing between different camera viewpoints. A new promising view addressing person re-identification in an end-to-end framework is presented in [Xiao et al., 2016b] that jointly handles pedestrian detection and searching in one unified trainable net. Despite it achieves impressive results on real-world street snaps and movies its generic Faster R-CNN-based detector brings some problems because it limits the depth of the following re-id feature extraction subnet. A remarkable approach is introduced by [Zheng et al., 2017] which performs a misalignment correction by Convolutional Pose Machines and combines the corrected and the original features together to enhance the overall discriminative power. The feature extraction part relies entirely two parallel CNNs. In these works the training is performed for identity classification but the learned network is employed for re-id feature extraction according to the transfer learning principle, borrowing the idea from [Sun et al., 2014] in face verification/recognition.

Hybrid Feature. Hybrid approaches combine hand-crafted features with the "learning from data" paradigm: [Wang, 2014] and [Ahmed et al., 2015], for example, design siamese CNNs with constraints on the shape of the objective to learn by adding hand-crafted layers. A different hybrid strategy is applied in [Wu et al., 2016] where a new feature extraction model produces more discriminative features by fusing together convolutional and hand-crafted histogram features, complementary to some degree.

Training Losses. One of the most effective losses used for training CNNs in person re-id is represented by the softmax function when included into the siamese network model. Two recent works exploiting this configuration are [Varior et al., 2016b] and [Varior et al., 2016a] that consider ways of exploiting spatial relations of images (within a single image or between image pairs). A different loss to the one used in siamese CNNs, the "triplet loss" is adopted by the triplet network model in [Hoffer and Ailon, 2015] with the effect of gaining insensitivity (differently from siamese CNNs) to context calibration. On the other side, this is paid in terms of training complexity and slow convergence caused by the explosion of the number of samples. One limitation of both the siamese and triplet models is that they only rely on weak re-ID labels (same id or different id). A modification of the softmax loss, the "random sampling softmax loss, is proposed in "[Xiao et al., 2016b] and allows to supervise the training with sparse and unbalanced labels.

3 Proposed Method

Aiming to mitigate the impact of viewpoint changes on the performance by enhancing the discriminative features of the pedestrian images, we propose the intra-GCL and the inter-GCL. In order to clarify the role that the two objectives play in the training process, let us call *sub-class* each of the feature clusters belonging to an identity that corresponds to a particular field of view (e.g the blue triangles in Fig.2). There are up to 6 sub-classes for an identity in Market-1501 and always 2 in CUHK03).

$$L_{intra} = \frac{1}{2}\sum_{i=1}^{m} \|\boldsymbol{x}_i^{g_i} - \boldsymbol{c}_{y_i}^{g_i}\|_2^2 \qquad (1)$$

$$L_{inter} = \sum_{i}^{m} \frac{\sum\limits_{\substack{g=1 \\ g\neq g_i}}^{s} \|\boldsymbol{x}_i^{g_i} - \boldsymbol{c}_{y_i}^{g}\|_2^2}{\sum\limits_{t=1}^{n}\sum\limits_{\substack{g=1 \\ g\neq g_i}}^{s} \|\boldsymbol{x}_i^{g_i} - \boldsymbol{c}_{t}^{g}\|_2^2} \qquad (2)$$

 ISBN 978-0-9934207-2-6

Intra-Group Centre Loss. It aims to encourage all the multi-dimensional points belonging to a sub-class of an identity to get near to the centre of that sub-class itself, gaining in compactness. It is formulated in Eq. (1) where $\boldsymbol{x}_i^{g_i}$ is a multi-dimensional point in the feature space, y_i is the ground-truth label corresponding to the i^{th} mini-batch point, g_i is the sub-class to which belongs the i^{th} mini-batch point, m is the training mini-batch size. The intra-GCL needs to be applied in joint supervision with the softmax loss: the presence of the softmax avoids that the centres degenerate to the null solution, while the presence of the intra-GCL avoids that the deep features contain too much intra-class variations. A week aspect of the intra-GCL is that, in the multi-camera scenario where the changing viewpoint problem may occur (Fig. 1), it may bring together images of different people sharing the same camera viewpoint, which is the reason why it requires to be balanced by the inter-GCL.

Inter-Group Centre Loss. The inter-GCL aims to penalize the distances of the image representation currently contributing to the training from all the centres of the sub-classes belonging to its identity, except its own. On the other side, it pushes away from the current feature point all the sub-classes referring to a different view and belonging to the rest of the training set identities. The inter-GCL is formulated in Eq. (2) where $\boldsymbol{c}_{y_i}^{g}$ is the centre of the sub-class g of identity y_i, s is the maximum number of cameras in the dataset and n is the number of identities in the training set. The reason why in the summation at the numerator in Eq. (2) we do not include all the terms $\|\boldsymbol{x}_i^{g_i} - \boldsymbol{c}_t^{g_i}\|_2^2$ is because they are already accounted separately in Eq. (1). The underlying assumption about this formulation is that pedestrians captured by the same camera present similar poses: the more it is true for a specific dataset the larger improvement is expected from the joint training supervision. One situation that this loss might not handle properly, though, is a possible initialization where a few image-centre distances (Fig. 2) contributing to the summation in the denominator of Eq. (2) take much larger values than the rest of the summation terms.

Combined With Softmax Loss. The two losses work in combination with the softmax loss. In particular, training involving the inter-GCL requires the concurrent supervision of both the intra-GCL and the softmax loss to avoid the dispersion of the view-field-based sub-classes. Indeed, the inter-GCL only expresses a relative constraint on the intra-class compactness (numerator in Eq. (2)) respect to the inter-class distance (denominator) so it needs to be combined to an absolute distance-based constraint provided by the intra-GCL. The linear combination of the three losses is expressed by the two parameters λ_{intra} and λ_{inter} in Eq. 3 where the first term represents the softmax loss with $\boldsymbol{W}_j$ denoting the j^{th} column of the weights $\boldsymbol{W}$ in the last fully connected layer and $\boldsymbol{b}$ the bias term.

$$L = -\sum_{i=1}^{m} log\frac{e^{\boldsymbol{W}_{y_i}^T \boldsymbol{x}_i + \boldsymbol{b}_{y_i}}}{\sum_{j=1}^{n} e^{\boldsymbol{W}_j^T \boldsymbol{x}_i + \boldsymbol{b}_j}} \quad + \lambda_{intra} L_{intra} \quad + \lambda_{inter} L_{inter} \tag{3}$$

The Stochastic Gradient Descent (SGD) is used for back-propagating the error of the derivatives of the loss with regards to the data $\boldsymbol{x}_i^{g_i}$, for $i = 1, ..., m$ and also respect to the centres of all the sub-classes of all the identities (Appendix A). The centres too, as the rest of the CNN parameters are updated in mini-batches according to the equation $c_j^{t+1} = c_j^t - \alpha \frac{\partial L_{(*)}}{\partial c_j^t}, \ \forall j$, where t is the iteration step, α is a scalar with values in $[0, 1]$ controlling the learning rate of the centres and $L_{(*)}$ denotes either the L_{intra} or L_{inter}, depending on which loss function we are focusing on.

4 Experiment

Database. We perform our experiments against two of the largest datasets for the person re-id task, **CUHK03** ([Wang, 2014]) and **Market-1501** ([Zheng et al., 2015]), since they allow to learn a CNN on a significant number of different views of the same identity. The "labelled" subset of CUHK03 is made up of 1360 different identities, each with up to 10 images, the first 5 seen under one camera and the remaining 5 under a different field of view. We reproduces the setting used in [Wang, 2014]: each of the 20 testsets counts 100 images and

 ISBN 978-0-9934207-2-6

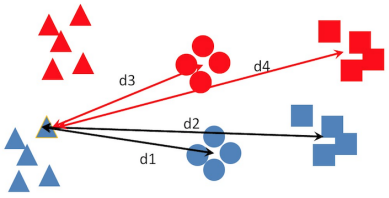


Figure 2: Conceptual scheme of how the inter-GCL function operates. Colour represents identity membership, shape indicates camera view membership. The arrows pointed blue triangle is the image currently contributing to the loss. The function may be re-written like this: $L_{inter}(\boldsymbol{x}_i) = \frac{d_1+d_2}{d_1+d_2+d_3+d_4}$. Best viewed in colour.

the validation set includes 100 identities. Market-1501 has got a larger depth than CUHK03. It exposes each identity up to 6 views and for each view several tens of instances of the same person are present. It consists of a train set of 751 identities shot in 12936 images and a testing set of 750 identities, corresponding to 13115 images. 2798 "distractors" (heavily misaligned detections) are also added to the test set.

Evaluation Metric and Protocols. It is worth noting that in order to measure the performance against the two datasets, the Cumulative Matching Curve (**CMC**) is employed for both datasets. In the case of Market-1501, which provides many ground truths for each query, the mean-Average Precision (**mAP**) is also computed to take into account both precision and recall. For CUHK03 we have adopted the evaluation protocol in [Wang, 2014]. Following this, the first 5 images of each identity (shot by camera 1, 3, or 5) represent view A, the remaining 5 (shot by camera 2, 4 or 6) represent view B. The probe set consists of the images seen in view A. The gallery-set of each probe is made up of 100 randomly chosen images seen in view B, one per each of the 100 identities in the testset. The gallery images selection is repeated 100 times, the CMC is calculated each time and, finally, the mean CMC curve is reported. The evaluation protocol we have implemented in Python for Market-1501 is compliant with the one used in [Zheng et al., 2015] according to which each of the 3368 queries is to be tested against its own gallery-set. The gallery set is formed of all the testing images except the ones having a file-name starting with '-1' (ID identifier) and the ones belonging to the probe's "junk set" comprising all the test images having the probe's same identity and field of view.

Implementation Details. One Caffe layer has been implemented separately for each loss and concatenated to the softmax loss layer. The SGD proceeds on the base of mini-batches and, since the two losses are added separately as two different layers to the Caffe training prototxt file (defining the structure of the CNN at trainaing time), two independent systems of centres are generated for the two layers which progressively converge. We experimented the two training objectives to train ResNet50 ([Zheng et al., 2017]), a residual learning-based state-of-the-art CNN ([He et al., 2016]) formed by 53 convolutional layers, feeding it with RGB images resized to 224x224 pixels. It is first trained for the identity classification task and, at testing stage, the deep features are extracted from the fully connected layer *pool5* for ResNet50, with dimension 2048. The training was extended up to 15000 iterations in all our simulations in the configuration softmax + intra-GCL + inter-GCL. The better identity classification performance are reached, the better re-id accuracy is achieved (Table 2). We ran our deep learning experiments on a single machine equipped with one NVIDIA GeForce GTX Titan X GPU and an Intel Core i7-5960X CPU @ 3.00GHz, 64.0 GB RAM. The training takes 4 hours for 15000 iterations.

Experimented Results. We report in Table 2 (with related graphs in Fig. 3) the results achieved on CUHK03 and Market-1501 with ResNet50 supervised by the softmax loss in linear combination with the intra-GCL and inter-GCL against the results achieved by the usual training relying only on the softmax loss. The study has been carried out by parametrizing the performance with regards to the two scaling factors λ_{intra} of the intra-GCL and λ_{inter} of the inter-GCL. By varying $(\lambda_{intra}, \lambda_{inter})$ in the range $[10^{-5}, 10^{-2}]$, it comes out that the point of maximum for the rank 1 re-id accuracy is $(\lambda_{intra}, \lambda_{inter}) = (5 * 10^{-4}, 10^{-4})$. Table 1 reports the performance by

 ISBN 978-0-9934207-2-6

	CUHK03	Market-1501	
L	rank1	rank1	mAP
0	51.60	73.02	47.62
0.00001	63.66	76.22	53.39
0.00005	62.35	76.48	53.95
0.0001	60.85	76.51	53.43
0.0005	61.57	75.89	53.26
0.001	59.27	75.83	53.46
0.005	57.30	75.27	52.92
0.01	51.25	74.61	50.86

Table 1: Performance (%) under the combined losses supervision for $\lambda_{intra} = 0.0005$ and λ_{inter} changing.

	CUHK03		Market-1501		
	rank1	id acc	rank1	mAP	id acc
Bow+Kissme [Zheng et al., 2015]	-	-	44.42	20.76	-
Null Space [Zhang et al., 2016]	54.7	-	55.43	29.87	-
LSTM Siamese [Varior et al., 2016b]	57.3	-	61.6	35.3	-
Gated Siamese [Varior et al., 2016a]	61.8	-	65.88	39.55	-
Baseline(R,pool5) [Zheng et al., 2017]	51.60	94.23	73.02	47.62	91.19
ours	**63.66**	96.79	**76.51**	**53.43**	93.59

Table 2: Softmax vs combined losses supervision for ResNet50. Results (%) at the point of maximum in the $(\lambda_{intra}, \lambda_{inter})$ plane. The accuracy (*id acc*) of the identity classification task is measured at iteration #15000.

varying λ_{inter} when λ_{intra} is fixed at 0.0005. The same data are plotted in Fig. 3. For Market-1501 a rank 1 accuracy of **76.51**% is achieved, improving the baseline result (72.41%) in [Zheng et al., 2017] of 5.66% as shown in Fig.3. The correspondent mAP value is **53.43**%, that improves the $Baseline(R, pool5)$ result (46.79%) of 14.19%. For CUHK03 the improvement is 23.37%. Furthermore, Table 2 shows that our method achieves better performance than many state-of-the-art approaches like [Varior et al., 2016b] or [Varior et al., 2016a].

5 Conclusion

In this paper we have proposed two new loss functions for training a state-of-the-art CNN, re-formulating the centre loss for the person re-identification task, in order to get more discriminative features that could mitigate the effects of camera viewpoint changes on pedestrians. The experiments presented showed that the supervision of the two losses, combined with the softmax loss, helps significantly the performance in the disjoint multi-camera scenario, beating several state-of-the-art approaches on CUHK03 and Market-1501.

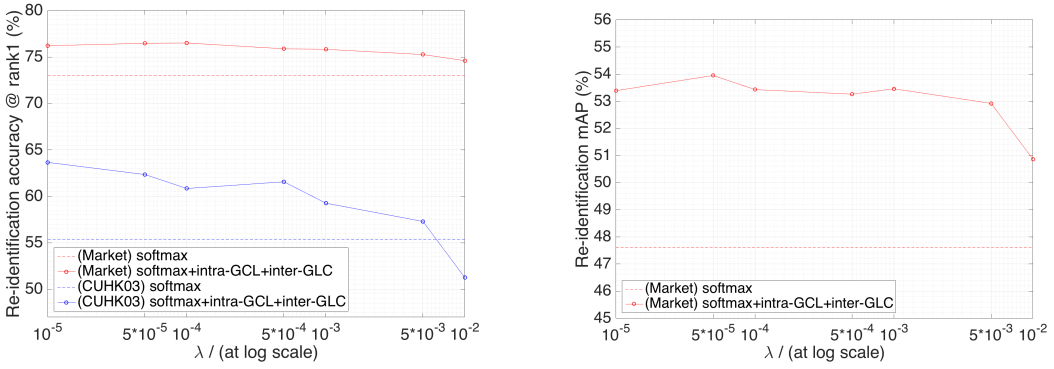


Figure 3: Performance improvement that our method achieves on ResNet50. Best viewed in colour.

 ISBN 978-0-9934207-2-6

A APPENDIX

$$\frac{\partial L_{inter}}{\partial \boldsymbol{x}_i^{g_i}} = 2\frac{\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}(\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_j^{g})*\sum\limits_{t=1}^{n}\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}\|\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_t^{g}\|_2^2-\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}\|\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_j^{g}\|_2^2*\sum\limits_{t=1}^{n}\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}(\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_t^{g})}{(\sum\limits_{t=1}^{n}\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}\|\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_t^{g}\|_2^2)^2}\delta(y_i=j) \tag{4}$$

$$\frac{\partial L_{inter}}{\partial \boldsymbol{c}_q^{k}} = \begin{cases} 2\sum\limits_{i=1}^{m}\frac{(-\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}\|\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_j^{g}\|_2^2)*(\boldsymbol{c}_q^{k}-\boldsymbol{x}_i^{g_i})}{(\sum\limits_{t=1}^{n}\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}\|\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_t^{g}\|_2^2)^2}\delta(y_i=j), & \text{for } q\neq j, k\neq g_i \\ 2\sum\limits_{i=1}^{m}\frac{(\sum\limits_{t=1}^{n}\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}\|\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_t^{g}\|_2^2-\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}\|\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_j^{g}\|_2^2)*(\boldsymbol{c}_q^{k}-\boldsymbol{x}_i^{g_i})}{(\sum\limits_{t=1}^{n}\sum\limits_{\substack{g=1\\g\neq g_i}}^{s}\|\boldsymbol{x}_i^{g_i}-\boldsymbol{c}_t^{g}\|_2^2)^2}\delta(y_i=j), & \text{for } q=j, k\neq g_i \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

References

[Ahmed et al., 2015] Ahmed, E., Jones, M., and Marks, T. K. (2015). An improved deep learning architecture for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3908–3916.

[Bak et al., 2015] Bak, S., Martins, F., and Bremond, F. (2015). Person re-identification by pose priors. In *SPIE/IS&T Electronic Imaging*, pages 93990H–93990H. International Society for Optics and Photonics.

[Chen et al., 2012] Chen, D., Cao, X., Wang, L., Wen, F., and Sun, J. (2012). Bayesian face revisited: A joint formulation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7574 LNCS(PART 3):566–579.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778.

[Hoffer and Ailon, 2015] Hoffer, E. and Ailon, N. (2015). Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*, pages 84–92. Springer.

[Kulis, 2013] Kulis, B. (2013). Metric Learning: A Survey. *Foundations and Trends® in Machine Learning*, 5(4):287–364.

[Pishchulin et al., 2016] Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P. V., and Schiele, B. (2016). Deepcut: Joint subset partition and labeling for multi person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4929–4937.

[Sun et al., 2014] Sun, Y., Wang, X., and Tang, X. (2014). Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1891–1898.

[Varior et al., 2016a] Varior, R. R., Haloi, M., and Wang, G. (2016a). Gated siamese convolutional neural network architecture for human re-identification. In *European Conference on Computer Vision*, pages 791–808. Springer.

[Varior et al., 2016b] Varior, R. R., Shuai, B., Lu, J., Xu, D., and Wang, G. (2016b). A siamese long short-term memory architecture for human re-identification. In *European Conference on Computer Vision*, pages 135–153. Springer.

[Wang, 2014] Wang, X. (2014). DeepReID : Deep Filter Pairing Neural Network for Person Re-Identification. *Cvpr*, pages 1–8.

[Wen et al., 2016] Wen, Y., Zhang, K., Li, Z., and Qiao, Y. (2016). A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, pages 499–515. Springer.

[Wu et al., 2016] Wu, S., Chen, Y.-C., Li, X., Wu, A.-C., You, J.-J., and Zheng, W.-S. (2016). An enhanced deep feature representation for person re-identification. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–8. IEEE.

[Xiao et al., 2016a] Xiao, T., Li, H., Ouyang, W., and Wang, X. (2016a). Learning deep feature representations with domain guided dropout for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1249–1258.

[Xiao et al., 2016b] Xiao, T., Li, S., Wang, B., Lin, L., and Wang, X. (2016b). End-to-end deep learning for person search. *arXiv preprint arXiv:1604.01850*.

[Yi et al., 2014a] Yi, D., Lei, Z., Liao, S., and Li, S. Z. (2014a). Constrained Deep Metric Learning for Person Re-identification. *2014 22nd International Conference on Pattern Recognition*, (1):34–39.

[Yi et al., 2014b] Yi, D., Lei, Z., Liao, S., and Li, S. Z. (2014b). Deep metric learning for person re-identification. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 34–39. IEEE.

[Zhang et al., 2016] Zhang, L., Xiang, T., and Gong, S. (2016). Learning a discriminative null space for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1239–1248.

[Zhao et al., 2013] Zhao, R., Ouyang, W., and Wang, X. (2013). Unsupervised salience learning for person re-identification. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3586–3593.

[Zhao et al., 2014] Zhao, R., Ouyang, W., and Wang, X. (2014). Learning mid-level filters for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 144–151.

[Zheng et al., 2017] Zheng, L., Huang, Y., Lu, H., and Yang, Y. (2017). Pose invariant embedding for deep person re-identification. *arXiv preprint arXiv:1701.07732*.

[Zheng et al., 2015] Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., and Tian, Q. (2015). Scalable person re-identification: A benchmark. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1116–1124.

On using CNN with DCT based Image Data

Matej Ulicny & Rozenn Dahyot

School of Computer Science and Statistics
Trinity College Dublin, Ireland

Abstract

This paper investigates the use of Convolutional Neural Networks (CNN) to classify images encoded in compressible form using Discrete Cosine Tranform (DCT) as an alternative to raw image format. We show experimentally that DCT features, that are directly available from JPEG format for instance, can be processed as efficiently as raw image data using the same CNN architectures.

Keywords: Classification, Deep Learning, CNN, DCT.

1 Introduction

In the past decade, convolutional neural networks (CNN) have grown in popularity for performing image processing tasks. In CNNs, convolutional filters are trained to extract relevant features and shapes from images to perform classification for instance (c.f. Figure 1). Convolutions by small size filters have already been widely adopted for instance in VGG [Simonyan and Zisserman, 2014] or ResNet [He et al., 2015]. A small filter covers only a small part of an image and thus has to be applied numerous times. In early stages of convolutional networks learning from large images, the spatial resolution of the feature space is rather large and the neural network has to perform a vast amount of operations. This paper harnesses the idea to exploit broadly used image compression techniques, in particular JPEG compression format for classification (see Section 3). Standard image classification techniques use CNNs on spatial representation of data, for example RGB pixel intensities, in contrast images in JPEG format are mapped to the frequency domain, which applicability for CNNs is explored in the paper. Section 2 introduces first related research works, Section 3 gives detailed explanation of the proposed approach, and we show several experimental results in Section 4 that confirms that Convolutional Neural Networks can be applied efficiently to image data in frequency domain.

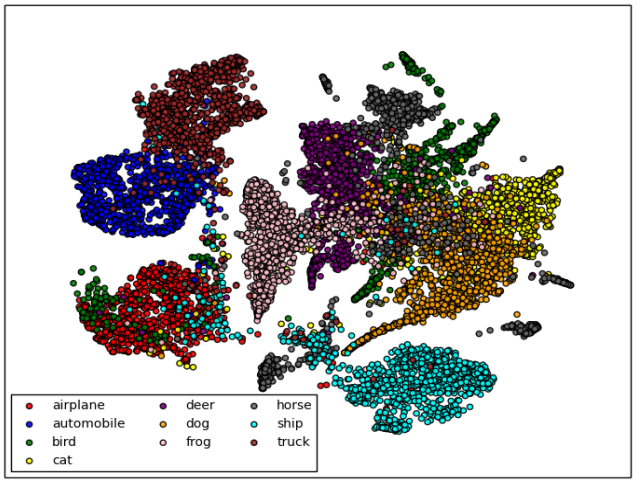


Figure 1: Separability of high level features learned on a datasets of raw RGB images capturing 10 object classes (CIFAR-10).

2 Related Work

Several papers addressed learning from frequency data. Zou et al. applied DCT on images of handwritten digits and used restricted number of frequency coefficients to train a deep belief network [Zou et al., 2014]. Er et al. used DCT features to train Radial Basis Function network for face recognition [Er et al., 2005].

 ISBN 978-0-9934207-2-6

In contrast, our approach works with frequency information of the image subregions, preserving their global localization. Under such setting, approaches that exploit spatial dependencies of the data, can be used, and in particular convolutional neural networks. CNNs have been successful for processing data that have underlying Euclidean or a regular grid-like structure (e.g. pixel grid) and, the size and structure of input data is expected to be fixed to feed into the networks used [Bronstein et al., 2017]. We present next our approach that takes advantage of JPEG compressed image format for creating CNN compliant input data to feed into CNN based classifiers.

3 Method

Our method is motivated by reusing well performing and broadly used JPEG image compression. Firstly we will briefly illustrate how the compression works. JPEG uses YCbCr color scheme, consisting of luma component (Y), representing luminance, and two chroma components, (Cb and Cr), capturing the blue and the red color difference. Color channels are sometimes subsampled for larger compression ratio. Each image channel is split into 8x8 pixel regions, 128 is subtracted from each pixel value, and finally the regions are transformed by 2 dimensional Discrete Cosine Transform (DCT).

Discrete Cosine Transform (DCT): 2-dimensional DCT transform on an input X with dimension N to the output Y is defined as

$$Y = C^N \cdot X \cdot (C^N)^T \tag{1}$$

where coefficient of the transform matrix is defined by:

$$C^N_{jk} = \sqrt{\frac{\alpha_j}{N}} \cos\left(\frac{\pi\,(2k+1)\,j}{2N}\right) \tag{2}$$

given $\alpha_j = 1$ for $j = 0$, and $\alpha_j = 2$ if $j > 0$.

Result of the DCT transform is mapping of the sub-region to the frequency domain. Upper-left corner of the sub-region contains low frequencies while the high frequencies occur in lower-right part. If the compression is lossy, the sub-regions are factorized by 8x8 matrix to discard high frequencies.

Finally, transformed and quantized sub-regions are converted into one dimensional Huffman code (entropy encoding), ordering pixels in an antidiagonal order starting from top-left corner (the lowest frequency). Given the non-homogeneous structure of the Huffman code, its variable length output is causing a challenge to fit the data to a fixed-sized input network such as CNN based architectures. Our experiments are focused here on using fixed size outputs of the DCT transform as inputs for our classifier as shown on the Figure 2.

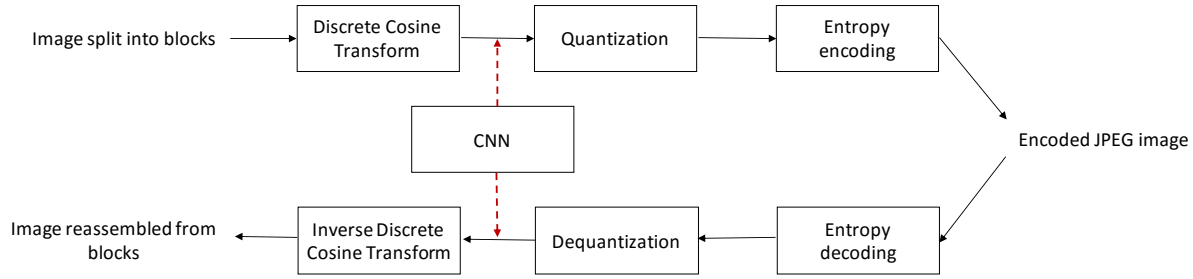


Figure 2: Flowchart showing JPEG (de)compression steps: Red dashed arrows indicate where CNN classifier is plugged-in.

We consider an image split to sub-regions or windows defined by the window size t. Hence windows of arbitrary size are considered, given JPEG compression splits image only to windows of size 8, the compressed data is not used, but original image data of each window is transformed into DCT space. Alternatively, if $t = 8$ is desired, entropy coded images can be decoded and dequantized, a process already performed when

 ISBN 978-0-9934207-2-6

extracting RGB infromation from JPEG images, avoiding aditional computational costs. The output of DCT transform is a set of frequency coefficients, each connected to a particular DCT basis function. A set of these coefficients represents a feature space of an image window. The filter of the proposed approach is designed to cover 2 adjacent windows in each image direction, forming a $2t \times 2t$ kernel. A weight of the filter is responsible for a particular frequency in the feature space. To avoid application of weights to coefficients with different frequencies, the convolutional filter slides along the image with a stride of one region width t. For a window size $t = 8$, a filter of 16×16 is used. Note that for inference purpose, the window does not need to have a square size, it can have an arbitrary shape composed of t^2 values (e.g. a 1 dimensional vector $1 \times t^2$ as long as relative position of the windows stays intact). The approach outline is depicted on Figure 3.

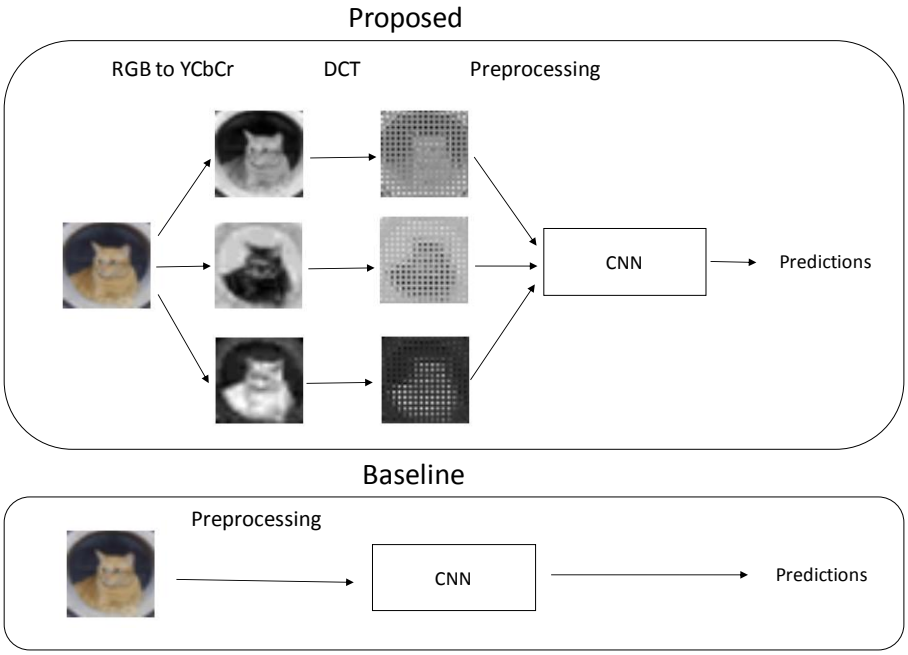


Figure 3: Flowchart of the proposed approach compared to the baseline, first converting RGB to YCbCr, then slicing image to windows and transforming each window separately.

4 Experimental Results

To evaluate the suitability of frequency data representation for training a CNN, experiments are performed on 2 well-known public datasets: CIFAR10 [Krizhevsky, 2009] and MNIST [LeCun et al., 1998].

4.1 CIFAR10

The dataset CIFAR10 of colored images of size 32×32 is used to compare low level features learned from frequency in contrast with original raw image intensity. This experiment is conducted for two values of t: 4 and 8. For each window size a different shallow network is used, further referred to as CNN-A ($t = 4$) and CNN-B ($t = 8$) with architectural details in Table 1. CNN-A and CNN-B are trained on the output of window-based DCT of the train set consisting of 50000 images. Features learned by convolutional layers are transformed by batch normalization [Ioffe and Szegedy, 2015] and activated by ReLU [Nair and Hinton, 2010]. CNN-A further performs average pooling to downsample the feature space, while CNN-B uses dropout technique to reduce overfitting [Srivastava et al., 2014].

In all our experiments we use weight decay $\lambda = 0.0001$, and batch size 256 except for MNIST experiments where the batch size is 128. Both shallow architectures (CNN-A, CNN-B) are trained on the original raw RGB data, raw YCbCr data representation and on its compressed DCT transform. Stochastic gradient descent

 ISBN 978-0-9934207-2-6

4x4 (CNN-A)			8x8 (CNN-B)		
layer name	kernel type	output size	layer name	kernel type	output size
conv	[8x8, 4x4]	7x7, 64	conv	[16x16, 8x8]	3x3, 64
avg-pool	[3x3, 2x2]	3x3, 64	dropout	p=0.25	3x3, 64
softmax		1, 10	softmax		1, 10

Table 1: The specification of CNN-A and CNN-B, a shallow convolutional networks for learning low level CIFAR10 features. A kernel definition of [8x8, 4x4] denotes a filter with spatial size 8×8 with stride 4×4. Output size of $3 \times 3, 64$ represent 64 feature maps with size 3×3. Dropout layer probability is defined by p.

with momentum is used to train the network, with learning rate starting at 0.1, reduced by factor 10 after 40 and 60 epochs for a total length of 80 epochs. Different input preprocessing techniques are tested and two well performing ones are reported: per feature normalization, noted as *mean/std* (for frequency data after performing DCT on non-centered data) and the *center/max* preprocessing by subtraction of 128 from the image pixel values and division by the original maximum value (255 for colors, $t \cdot 256$ for frequency data after performing DCT on centered images). Using the whole test set of 10000 images, the highest classification accuracy was observed for models trained on DCT data with *center/max* preprocessing. The results of 20 runs for each setting, depicted in Table 2 demonstrate, the window based DCT transform facilitates learning of more discriminative low-level features.

window size	4x4 (CNN-A)		8x8 (CNN-B)	
preprocessing	mean/std	center/max	mean/std	center/max
RGB	66.82 ± 0.39	66.96 ± 0.36	60.36 ± 0.37	60.20 ± 0.31
YCbCr	65.57 ± 0.36	67.07 ± 0.24	59.60 ± 0.34	60.25 ± 0.27
DCT	66.84 ± 0.23	$\mathbf{67.24 \pm 0.26}$	60.25 ± 0.28	$\mathbf{60.87 \pm 0.26}$

Table 2: Classification scores (mean ± std %) of shallow CNN-A and CNN-B networks over 20 runs on CIFAR10.

Empiric results indicate the standard *mean/std* preprocessing technique for RGB data is not well suited for data in YCbCr format. We suspect the imbalance between standard deviations of luma and chroma channels scales down the more important luma channel that is then not exploited sufficiently by a layer normalized with L2 norm.

The low level features show encouraging results, motivating experiments on a deeper network. A network CNN-C with details in Table 3 is trained with both previously mentioned preprocessing approaches for all 3 tested data formats. Each layer with trainable parameters (except for the softmax layer) is batch normalized and activated by ReLU. Excluding the first layer, all convolutional layers preserve spatial dimensionality of the features. Due to a low resolution of the dataset, window size t is set to 2 to prevent drastic downsampling after the first convolutional layer. We use the same training method as for the shallow network with difference in the learning rate scheduler: the network is trained for a length of 300 epochs, having initial learning rate 0.1 reduced by factor 5 after 90, 180 and 240 epochs.

The Table 4 contains median and mean accuracy on the test set after 300 epochs, trained both without augmentation and with simple augmentation. The images are augmented by random horizontal flipping and random shifting by multiples of t, at most by $2t$, filling missing pixels by zeros. When augmentation is not used, the network trained on DCT achieves slightly higher accuracy, however, RGB representation benefits from augmentation more than other representations.

We conduct a visual evaluation of CNN-C network features learned on RGB and DCT data. Firstly, low level first layer activations of the model (*mean/std* and *center/max* preprocessing for RGB and DCT data respectively) are rendered on Figure 4 that confirms both networks learned similarly looking features. Furthermore, discriminating properties of the high level features of CNN-C model trained on DCT is demonstrated by mapping activations of the dense layer ("dense1" in Table 3) to the 2D space (Figure 5) via t-distributed stochastic neighbor embedding (t-SNE) [van der Maaten and Hinton, 2008]. Class compactness in the 2D projection is visually similar to the projection of features of the same network trained on raw RGB data (Figure 1).

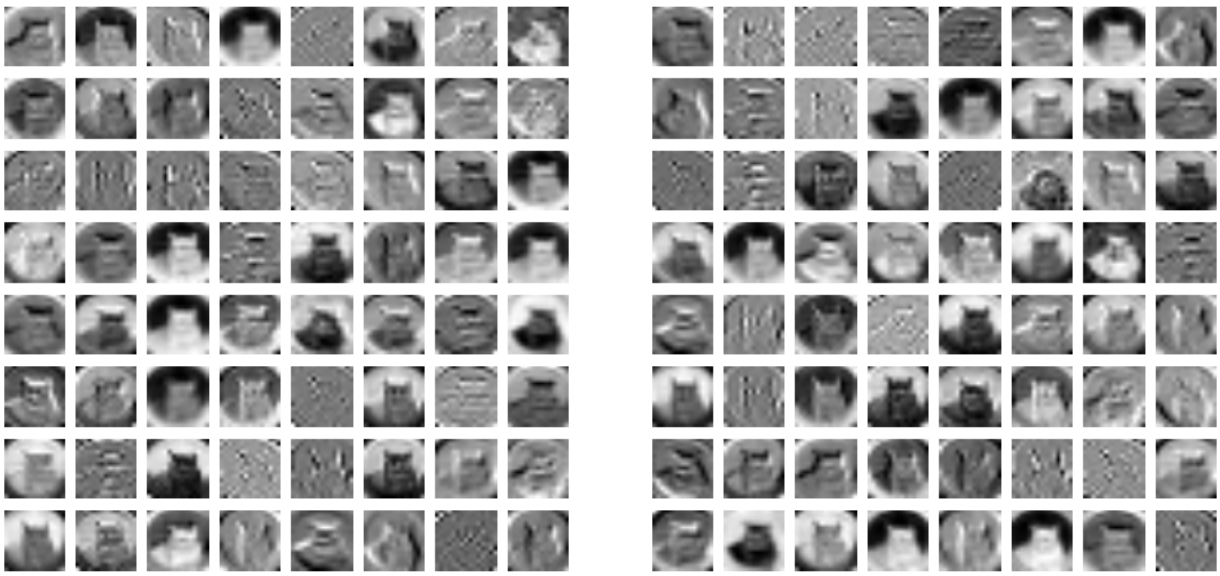
 ISBN 978-0-9934207-2-6

layer name	kernel type	output size
conv1	[4x4, 2x2]	15x15, 64
conv2	[3x3, 1x1]	15x15, 64
dropout1	p = 0.25	15x15, 64
conv3	[3x3, 1x1]	15x15, 64
max-pool1	[3x3, 2x2]	7x7, 64
conv4	[3x3, 1x1]	7x7, 128
max-pool2	[3x3, 2x2]	3x3, 128
dropout2	p = 0.25	3x3, 128
dense1		1, 512
dropout3	p = 0.5	1, 512
softmax		1, 10

Table 3: Convolutional network architecture (CNN-C) used on CIFAR10.

augmentation	no augmentation		real-time augmentation	
preprocessing	mean/std	center/max	mean/std	center/max
RGB	86.11 (86.22 ± 0.25)	86.13 (86.21 ± 0.29)	**90.49 (90.49 ± 0.21)**	90.35 (90.30 ± 0.12)
YCbCr	85.77 (85.47 ± 0.41)	85.96 (86.13 ± 0.49)	89.98 (90.08 ± 0.31)	90.28 (90.18 ± 0.27)
DCT	**86.35 (86.25 ± 0.42)**	86.30 (86.27 ± 0.19)	89.97 (90.07 ± 0.27)	90.27 (90.28 ± 0.13)

Table 4: Classification scores computed by median and (mean ± std %) in brackets over 5 runs on CIFAR10 dataset for CNN-C.



(a) (b)

Figure 4: First layer activations of a median score CNN-C model trained with augmentation on RGB data (**a**) and DCT representation (**b**), created by inferring the image from Figure 3.

A comparison of common mistakes made by models trained on RGB and DCT data points out that both networks are making similar mistakes. Figure 6 presents the confusion matrices for both networks, showing the most common mistake for both models was confusing a dog for a cat. The findings further support the claim that both networks learn similar representations.

4.2 MNIST

We perform a similar experiment on MNIST dataset, training a CNN-D network similar (see Table 5) to the one used on CIFAR10 data. Unlike CIFAR10, MNIST contains only 1 color channel with dimension 28x28. Here

 ISBN 978-0-9934207-2-6

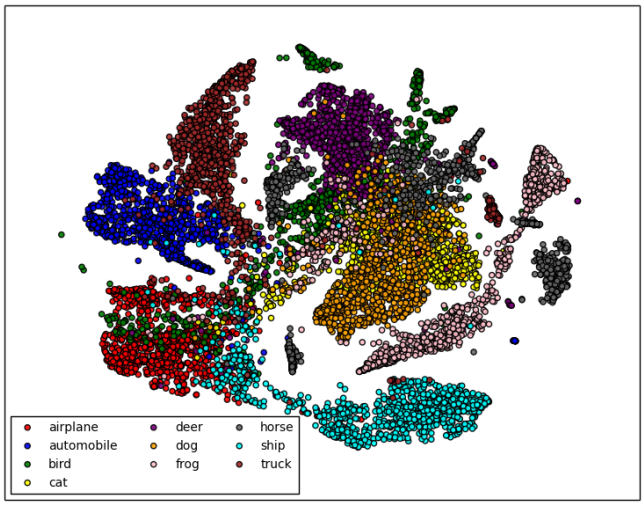


Figure 5: Mapping of high-level features of a network trained on DCT CIFAR10 representation into 2D plane.

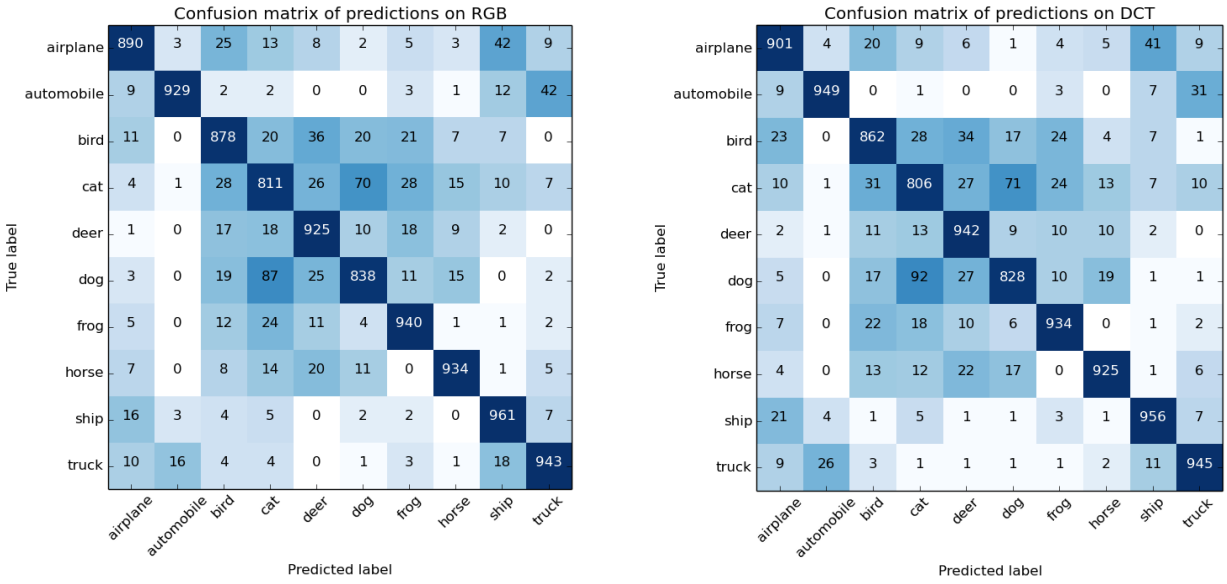


Figure 6: A confusion matrix of predictions by CNN-C network trained on RGB (**left**) and DCT representation (**right**) of CIFAR10.

we use $t = 2$ and train the network on the whole 60000 image large train set for 30 epochs (the whole dataset passes) with stochastic gradient descent. The initial learning rate is 0.1, which is every 10 epochs reduced by factor 10, and we use momentum of 0.9.

layer name	kernel type	output size
conv1	[4x4, 2x2]	13x13, 64
conv2	[3x3, 1x1]	13x13, 64
max-pool1	[3x3, 2x2]	6x6, 64
conv3	[3x3, 1x1]	6x6, 128
max-pool2	[2x2, 2x2]	3x3, 128
dropout1	p = 0.25	3x3, 128
dense1		1, 512
dropout2	p = 0.5	1, 512
softmax		1, 10

Table 5: Convolutional network architecture (CNN-D) used on MNIST.

 ISBN 978-0-9934207-2-6

Multiple preprocessing approaches are used for original intensity values and for DCT transform of the data. Table 6 lists average errors over 20 runs on the whole test set of 10000 images for each preprocessing method. The most successful preprocessing technique for raw data was subtraction of mean value and scaling down by

preprocessing	Orig. error (%)	preprocessing	DCT error (%)
/255	0.4455 ± 0.0499	DCT/512	0.4245 ± 0.0332
-128/128	0.4415 ± 0.0273	-128 DCT/256	0.4405 ± 0.0458
-128/255	0.4355 ± 0.0213	-128 DCT/512	$\mathbf{0.4105 \pm 0.0383}$
-mean/std	0.4445 ± 0.0407	DCT-mean/std	0.4320 ± 0.0499
-mean/128	$\mathbf{0.4245 \pm 0.0322}$	DCT-mean/256	0.4360 ± 0.0306
-mean/255	0.4425 ± 0.0360	DCT-mean/512	0.4385 ± 0.0432
		-mean DCT/std	0.4390 ± 0.0391
		-mean DCT/256	0.4460 ± 0.0434
		-mean DCT/512	0.4300 ± 0.0453

Table 6: Classification scores as the mean ± std % over 20 runs on MNIST dataset. Original raw data is compared to its DCT transform for different preprocessing techniques: "-" a constant or "mean" value represents subtraction of specified value from every image pixel, "DCT" stands for a performing a discrete cosine transform at the particular step, and "/" by a constant or "std" refer to scaling the image by specified value.

128 with error 0.4245%, followed by method that was reported in previous subsection on CIFAR10, subtracting 128 from pixels and scaling by 255, achieving 0.4355% error. The lowest average error of 0.4105% is obtained with DCT representation when subtracting 128 from the original image before performing the DCT and scaling the transformed data with a constant 512 depicted as *center/max* preprocessing in Section 4.1. The difference in error in favor of DCT representation is not significant however, given noticeably well performing baseline network. There is therefore not much space to observe an improvement.

4.3 Implementation details

All networks used in the experiments are modeled and trained in Keras deep learning framework that uses TensorFlow backend. Models were trained on NVIDIA GTX770 GPU with 2GB of memory. The CNN-C network with roughly 750 thousand parameters with training time of 20 seconds per epoch is fully trainined in less than 2 hours regardless of input representation, which is passed to the graphic card in form of 32bit precision floating point tensor. DCT transform is computed via provided implementation in opencv library. Time to transform the whole train set is platform dependent, on Intel processor with 3.7GHz frequency the transform with window size 2 applied to nearly 50 million sub-windows takes around 4 minutes, while using sizes 4 and 8, substantially less windows are processed leading to roughly 1 minute and 20 seconds long execution time respectively.

5 Conclusion & Future work

In this paper we presented an approach for adopting convolutional networks to learn on frequency representations of the data, with motivation to deploy this approach to compressed image data. Empirically, we showed that low level features learned from window based discrete cosine transform coefficients are comparably or more discriminative than those learned from raw data. High level feature analysis shows deeper networks trained on data transformed by DCT learn similar representations and make similar mistakes as their raw data counterparts. Our study encourages further experiments on high resolution images and videos in compressed format.

 ISBN 978-0-9934207-2-6

Acknowledgments

The ADAPT Centre for Digital Content Technology is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

References

[Bronstein et al., 2017] Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., and Vandergheynst, P. (2017). Geometric deep learning: going beyond euclidean data. Technical report, `https://arxiv.org/pdf/1611.08097.pdf`.

[Er et al., 2005] Er, M. J., Chen, W., and Wu, S. (2005). High-speed face recognition based on discrete cosine transform and rbf neural networks. *Trans. Neur. Netw.*, 16(3):679–691.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, abs/1512.03385.

[Ioffe and Szegedy, 2015] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167.

[Krizhevsky, 2009] Krizhevsky, A. (2009). Learning multiple layers of features from tiny images.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 11(86):2278–2324. `http://yann.lecun.com/exdb/mnist/`.

[Nair and Hinton, 2010] Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In FÃijrnkranz, J. and Joachims, T., editors, *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814. Omnipress.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

[Srivastava et al., 2014] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

[van der Maaten and Hinton, 2008] van der Maaten, L. and Hinton, G. E. (2008). Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

[Zou et al., 2014] Zou, X., Xu, X., Qing, C., and Xing, X. (2014). High speed deep networks based on discrete cosine transformation. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 5921–5925. IEEE.

Deep convolutional neural networks and digital holographic microscopy for in-focus depth estimation of microscopic objects

Tomi Pitkäaho,[1] Aki Manninen,[2] and Thomas J. Naughton[1]

[1] *Department of Computer Science, Maynooth University–National University of Ireland Maynooth, Maynooth, County Kildare, Ireland*

[2] *Biocenter Oulu, Oulu Center for Cell-Matrix Research, Faculty of Biochemistry and Molecular Medicine, University of Oulu, Finland*

Abstract

Deep artificial neural network learning is an emerging tool in image analysis. We demonstrate its potential in the field of digital holographic microscopy by addressing the challenging problem of determining the in-focus reconstruction depth of an arbitrary epithelial cell cluster encoded in a digital hologram. A deep convolutional neural network learns the in-focus depths from half a million hologram amplitude images. The trained network correctly determines the in-focus depth of new holograms with high probability, without performing numerical propagation. To our knowledge, this is the first application of deep learning in the field of digital holographic microscopy.

Keywords: Imaging, Digital Holographic Microscopy, Autofocusing, Deep Learning, Machine Learning

1 Introduction

Deep learning [LeCun et al., 2015] is a technique for solving hitherto open problems in image analysis and other fields that is starting to have an impact in the field of biomedical optics, for example OCT [Prentašić et al., 2016, Abdolmanafi et al., 2017, Karri et al., 2017] and other forms of microscopy [Cireşan et al., 2012, Wang et al., 2014, Rezaeilouyeh et al., 2016, Gopakumar et al., 2017]. In the context of this work, deep learning is defined to be a deep convolutional neural network with multiple hidden layers that perform convolution operations on its multi-dimensional input. Image-based applications of deep learning [Russakovsky et al., 2015] are characterised by neural networks with at least eight hidden layers, at least tens of thousands of images, at least hundreds of images per class, at least millions of learned parameters, and training times of at least weeks if run on a single-processor personal computer. This type of network has been used successfully in various different visual object recognition and object detection applications [Krizhevsky et al., 2012, Simonyan and Zisserman, 2014].

A digital hologram is an efficient encoding of a diffraction volume. An obvious requirement in digital holography would be the possibility to edit the hologram directly, in order to effect some semantic change in the diffraction volume, or even more simply, analyse the hologram directly in order to construct an understanding of the 3D scene. Unfortunately, in the general case this has eluded researchers in digital holography, including the authors. Researchers are limited to sampling the reconstruction volume (i.e. using numerical propagation to reconstruct from the digital hologram at a plurality of depths) before they can understand the encoded 3D scene. A handful of notable exceptions exist, such as the landmark papers by Vikram and Billet [Vikram and Billet, 1984] and Onural and Özgen [Onural and Özgen, 1992], and subsequently others over the past decade [Soulez et al., 2007, Cheong et al., 2010, Yevick et al., 2014, Schneider et al., 2016] whose work allows one to determine the size and position of individual particles based on an analysis of the

 ISBN 978-0-9934207-2-6

hologram directly. However, these approaches are limited to the special case of idealized spherical particles. Here, we consider significantly more complicated multi-cellular partially-transparent objects.

Holography has a history as an enabling technology for artificial neural networks [Psaltis and Farhat, 1985], and conventional artificial neural networks have been applied before in the fields of digital holographic microscopy [Kamilov et al., 2015, Schneider et al., 2016] and more generally digital holography [Frauel and Javidi, 2001, Shortt et al., 2006].

In this paper, we demonstrate that it is possible to design a deep convolutional neural network to predict the in-focus distance of a living cell cluster from the digital hologram plane amplitude only. With deep learning, we propose that digital holographic microscopy (DHM) researchers now have a tool at their disposal that is a major step towards removing the need to perform any propagation steps in order to determine the in-focus distance.

2 Deep learning

Deep convolutional neural networks (CNN), that are one form of deep learning, were discovered by LeCun et al. [LeCun et al., 1989]. CNN is an artifical neural network where some of the layers perform convolution operations on their multi-dimensional input. An image convolution is an operation where values of a vector (or pixels of an image) are multiplied by a weights of a sliding vector (or 2D matrix with 2D data such as an image) called a kernel. In convolutional neural network these weights are learned during the training. Four main ideas behind convolutional neural networks are: local connections, shared weights, pooling and the use of many layers [LeCun et al., 2015]. The basic principle of convolutional neural networks with images is that lower levels detect coarse features like edges that are combined by higher levels to parts and further to objects.

Each convolution operation produces a feature map that shares a same kernel, and different feature maps in a layer use a different kernel. The kernel is applied to the feature map of a previous layer, or at the beginning to the input image. Number of feature maps and therefore learned kernels is determined as part of a network architecture design. Output of a convolution is passed through an activation function such as rectified linear unit (ReLU). A standard principle is that higher convolution layers have higher depth (number of feature maps increases).

Pooling layers merge semantically similar features to one reducing dimensionality of feature maps. Pooling is based on small patches of a feature map of which for example a maximum or average is found/calculated and stored as a new value of a sub sampled feature map. Pooling increases the shift invariance [LeCun et al., 2015].

Deep convolutional neural networks that are one form of deep learning have been used successfully in various different visual object recognition and object detection applications [Krizhevsky et al., 2012, Simonyan and Zisserman, 2014]. Some of the layers in these networks perform convolution operations on its multi-dimensional input.

3 Digital holographic microscopy

DHM overcomes a problem present in optical microscopes of a shallow depth-of-field, allowing one to change the in-focus plane after hologram capture. A magnified digital hologram that is formed of reference, R, and object, O, waves as $H(x,y) = |R|^2 + |O|^2 + R^*O + RO^*$ can be propagated to any depth z using the Fresnel approximation [Goodman, 2005]

$$U(x,y;z) = \frac{-i}{\lambda z}\exp(\mathrm{i}kz)\, H(x,y) \otimes \exp\left(\mathrm{i}\pi\frac{x^2+y^2}{\lambda z}\right), \tag{1}$$

where λ is the wavelength of the light, $\otimes$ denotes a convolution operation and $k = 2\pi/\lambda$. The amplitude component of the complex-valued reconstruction is defined as

$$A(x,y;z) = \{\mathrm{Re}[U(x,y;z)]^2 + \mathrm{Im}[U(x,y;z)]^2\}^{0.5}, \tag{2}$$

 ISBN 978-0-9934207-2-6

where Re and Im are extracting real and imaginary components, respectively. However, since each in-focus plane has a narrow depth of field, the object of interest is in focus only at a small range of reconstruction depths. The problem of determining the most appropriate in-focus depth is essential for applications such as autofocusing, extended focus imaging, and segmentation. The critical importance of this problem to digital holography researchers is evidenced by the regularity of newly proposed focus metrics to apply amplitude and phase reconstructions such as self-entropy [Gillespie and King, 1989], phase changes [Ferraro et al., 2003], wavelet analysis [Liebling and Unser, 2004], integrated amplitude modulus [Dubois et al., 2006], gray-level variance [McElhinney et al., 2007], power spectra [Langehanenberg et al., 2008], Tamura co-efficient [Memmolo et al., 2012], multi-wavelength Fourier phase [Dohet-Eraly et al., 2016], cosine score [He et al., 2017], structure tensor [Ren et al., 2017], and magnitude differential [Lyu et al., 2017] among others. However, each suffers from the same drawback: a stack of reconstructed images must be computed, and the focus metric must be applied to each reconstruction. This time-consuming drawback is compounded by the fact that the whole procedure must be applied to each new hologram. The greatest benefit of the deep learning method outlined in this paper is that after training, the in-focus depth can be obtained from the hologram plane intensity directly, and in constant time, without any numerical propagation.

4 Experimental results

We chose a convolutional neural network approach to tackle the autofocusing problem still existing in digital holographic microscopy of transparent samples. The architecture of the network is based on the AlexNet architecture that won the Large Scale Visual Recognition Challenge 2012 [Krizhevsky et al., 2012], and was used as a benchmark for subsequent network architectures. One inducement of its use, within the application reported here, is the fact that this paper and results thereof can be considered as a proof of concept with a well known network architecture. AlexNet has 5 convolution layers, 3 fully-connected layers, and uses convolutional filters up to 11×11 pixels in size (Fig. 1).

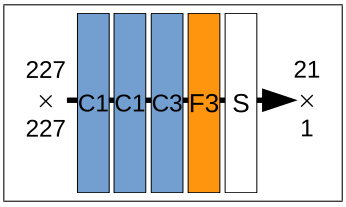


Figure 1: Network architecture. C, convolution block; F, fully-connected block; input size, 227×227 pixels. Numbers show amount of layers in each block. Each convolution block is followed by a maxpooling layer with kernel size of 3 and stride of 2.

4.1 Training

A total of 494 hologram of semitransparent Madin-Darby canine kidney (MDCK) epithelial cell clusters were captured using an off-axis Mach Zehnder digital holographic microscope (Lyncée Tec T1000, Lyncée Tec SA, Lausanne, Switzerland). The microscope comprises a 660 nm laser source, a 1024×1024 pixel CCD camera with 6.45 μm square pixels, and a 40X microscope objective with 0.7 numerical aperture (Leica HCX PL Fluotar). To obtain the ground truth data, one of the authors (T.P.) manually determined for each hologram the z (at 1 mm resolution) in Eq. 1 that brings the middle region of each cell cluster into focus. The middle region was considered to be in-focus when edges of cell cluster were estimated to display the lowest diffraction (caused by the top and bottom halves of the sample).

The holograms were used to generate a database of images as follows. An amplitude reconstruction was obtained from each hologram at each of 21 depths distributed equally over the range ±100 mm centred on the in-focus plane (see examples in Fig. 2). Through different combinations of rescaling and cropping, each re-

 ISBN 978-0-9934207-2-6

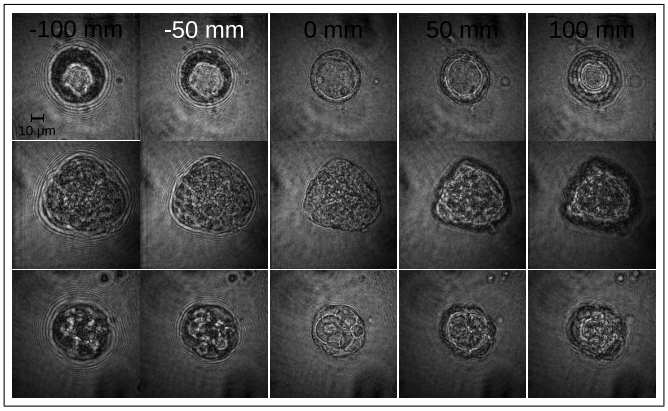


Figure 2: Example training images: each row shows amplitude reconstructions from one hologram (at the in-focus plane, and at the distances ±100 mm, ±50 mm from the in-focus plane).

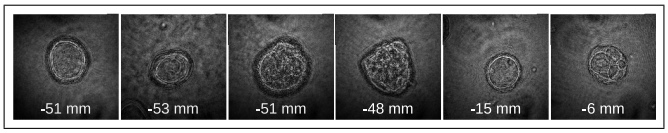


Figure 3: Example amplitudes of the twelve dc- and twin-free holograms used for testing (with ground truth in-focus distance shown).

construction was used to generate six similar but distinct 227×227 pixel rescaled and cropped images. This size was chosen as it is the size the network was originally designed for. Each such image was rotated through each of three distinct 90 deg. rotations, and each resulting image was further augmented through horizontal mirroring. This formed a database of 497 952 images. From this database, all augmented images from the twelve hand-picked holograms (comprising 12 096 images, 2.4% of the set) were set aside as test data (examples shown in Fig. 3). The remaining images were partitioned randomly into training (87.8%, 437 271) and validation (9.8%, 48 585) data. Finally, a mean image (calculated from the training data only) was subtracted from each image.

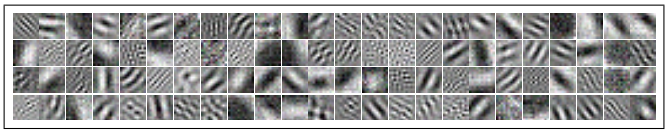


Figure 4: The learned filters from the first convolution layer: 96 11×11 pixel filters

The actual training used Nvidia's Deep Learning GPU Training System (DIGITS) software with Nvidia Titan X ('Maxwell') graphics card. Learning rate was fixed at 0.001. The network was trained for 100 epochs (16 hours) with a stochastic gradient descent solver. The loss function was categorical cross entropy and the network output a 21 element vector containing class probabilities.

The network was trained for 60 epochs with a stochastic gradient descent solver. The minibatch size was set to 100. When the training was finished, training loss was 2.497×10^{-2}, validation loss 0.129, and validation accuracy 95.6%. The learned filters from the first convolution layers are shown in the Fig. 4, allowing one to infer the basic features that the network learned to extract from an image for analysis in subsequent layers.

4.2 Testing

Testing was performed on a separate computer without a discrete graphics card to demonstrate the portability of deep learning. As an operating system this computer was running Ubuntu 14.04, had Intel Core i5 processor,

 ISBN 978-0-9934207-2-6

and 16 GB memory. The trained model (with size 227.4 MB) was imported into the Caffe [Jia et al., 2014] deep learning framework using the general-purpose Python programming language. The run time (mean of 200 holograms) was 247 ms. Using the same PC that was used for training (with a GPU support), run time was 4 ms. For comparison, a single Tamura coefficient calculation (including aberration removal, reconstruction, phase unwrapping and Tamura coefficient calculation) is 932 ms (aberration removal 380 ms, reconstruction 318 ms, phase unwrapping 231 ms, Tamura coefficient calculation 3 ms).

Testing was performed by classifying holograms that were not used in training or validation, as explained. Of the 12 096 test images, 99.9% were classified within one class of the ground truth depth (see Fig. 5). Although the depth classes are completely unrelated as far as the network is concerned, it is a remarkable indicator of robustness that where model incorrectly classifies an input, it invariably chose a neighbouring depth class instead.

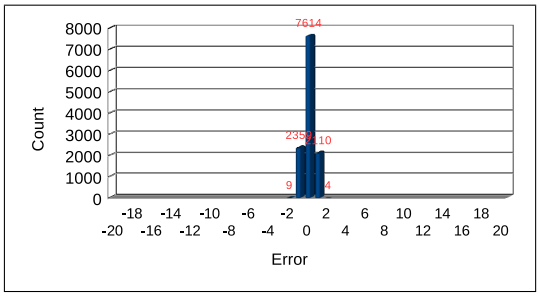


Figure 5: Classification errors with the testing data. In 62.9% of test cases the correct depth class was predicted. In 99.9% of test cases a correct depth is within one depth class.

Table 1: **Test results using the 12 holograms from Fig. 3, showing classification result from the network**

Groundtruth (mm)	Top two predictions (mm,mm)	Confidence values (%,%)
-51	-40,-50	100,0
-53	-50,-40	89,11
-51	-60,-50	99,1
-48	-40,-30	91,9
-15	-10,-20	100,0
-6	-10,0	100,0
1	0,10	100,0
7	0,10	100,0
0	0,-10	56,44
-2	0 -10	100,0
-74	-80,-70	100,0
-65	-60,-50	100,0

To examine how the network responded to holograms that may have an in-focus distance not a multiple of the 10 mm discretization used in training, the holograms from Fig. 3 were used directly (see Table 1). The two top predictions for the network typically straddle the correct answer. The network typically classifies with high confidence holograms with an in-focus distance close to a multiple of 10 mm. The mean absolute error over the 12 holograms was 5.01 mm.

Systematic testing was then performed with the holograms from Fig. 3 over the range ± 100 mm centred on the in-focus depth, but this time with a finer depth resolution of 1 mm. For a system to generalise well outside the discrete set of 21 in-focus depth classes with which it was trained, the shape of the scatter plot should exactly be a staircase with a linear trend. The network generalized well with each test hologram, and a typical example is shown in Fig. 6(a). To push the network past its designed capabilities, the network was tested (over the same depth range and resolution) with a human cell line sample captured with the same DHM hardware. The

 ISBN 978-0-9934207-2-6

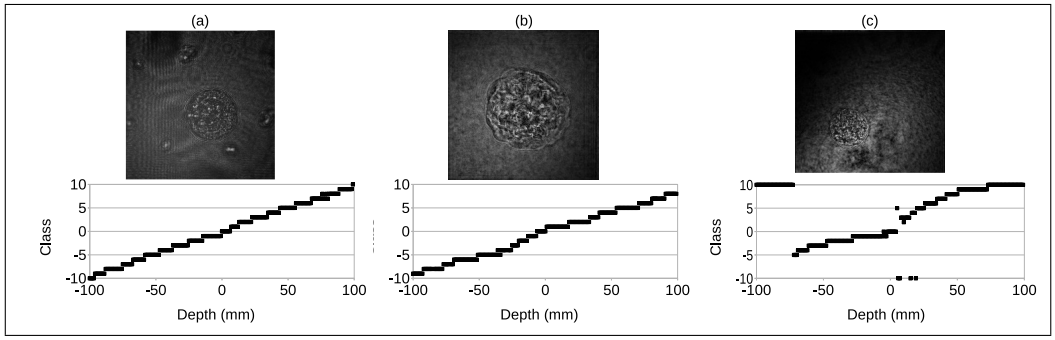


Figure 6: Fine (1 mm) depth-resolution test results (a linear staircase indicates perfect classification): (a) same DHM hardware and same sample type as in training, (b) same DHM hardware but different cell line, (c) different DHM illumination, magnification, and chemically fixed sample from a different cell line.

network performed surprisingly well with this sample, however classification, while largely monotonic, is no longer linear [see Fig. 6(b)]. Later, this exact sample was chemically fixed and captured using a different digital holographic microscope with different illumination and a lower magnification 20X microscope objective with 0.5 numerical aperture (Leica HCX PL Fluotar 20x). The network fails at some depths which is not surprising given the enormous differences in scale and optical density from the training set, overall the network performs well with this data [see 6(c)].

5 Conclusions

In this paper, to our knowledge the first application of deep learning to digital holographic microscopy, we show that an artificial neural network can be designed to learn the appropriate in-focus depth of an arbitrary MDCK cell cluster encoded in a digital hologram. Its greatest benefit is that the in-focus depth can be obtained from the hologram plane intensity in constant time without any numerical propagation. It generalises well to in-focus depths different from its training set, and there is evidence that it will degrade gracefully with differences in cell line, in fixing conditions, and in DHM architecture, from that used in training. As has been discovered in recent years in other fields of microscopy, we believe that deep learning has the potential to become an important tool for DHM.

Acknowledgments

This publication has emanated from research conducted with the financial support of an Irish Research Council (IRC) Postgraduate Scholarship and of Science Foundation Ireland (SFI) under grant no. 13/CDA/2224.

References

[Abdolmanafi et al., 2017] Abdolmanafi, A., Duong, L., Dahdah, N., and Cheriet, F. (2017). Deep feature learning for automatic tissue classification of coronary artery using optical coherence tomography. *Biomed. Opt. Express*, 8(2):1203–1220.

[Cheong et al., 2010] Cheong, F. C., Krishnatreya, B. J., and Grier, D. G. (2010). Strategies for three-dimensional particle tracking with holographic video microscopy. *Opt. Express*, 18(13):13563–13573.

[Cireşan et al., 2012] Cireşan, D., Giusti, A., Gambardella, L. M., and Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. *Advances in Neural Information Processing Systems*, 25:2843–2851.

[Dohet-Eraly et al., 2016] Dohet-Eraly, J., Yourassowsky, C., and Dubois, F. (2016). Fast numerical autofocus of multispectral complex fields in digital holographic microscopy with a criterion based on the phase in the Fourier domain. *Opt. Lett.*, 41(17):4071–4074.

[Dubois et al., 2006] Dubois, F., Schockaert, C., Callens, N., and Yourassowsky, C. (2006). Focus plane detection criteria in digital holography microscopy by amplitude analysis. *Opt. Express*, 14(13):5895–5908.

[Ferraro et al., 2003] Ferraro, P., Coppola, G., Nicola, S. D., Finizio, A., and Pierattini, G. (2003). Digital holographic microscope with automatic focus tracking by detecting sample displacement in real time. *Opt. Lett.*, 28(14):1257–1259.

[Frauel and Javidi, 2001] Frauel, Y. and Javidi, B. (2001). Neural network for three-dimensional object recognition based on digital holography. *Opt. Lett.*, 26(19):1478–1480.

[Gillespie and King, 1989] Gillespie, J. and King, R. A. (1989). The use of self-entropy as a focus measure in digital holography. *Pattern Recogn. Lett.*, 9(1):19–25.

[Goodman, 2005] Goodman, J. W. (2005). *Introduction to Fourier Optics*. Roberts and Company Publishers.

[Gopakumar et al., 2017] Gopakumar, G., Babu, K. H., Mishra, D., Gorthi, S. S., and Subrahmanyam, G. R. K. S. (2017). Cytopathological image analysis using deep-learning networks in microfluidic microscopy. *J. Opt. Soc. Am. A*, 34(1):111–121.

[He et al., 2017] He, G., Xiao, W., and Pan, F. (2017). Automatic focus determination through cosine and modified cosine score in digital holography. *Opt. Eng.*, 56(3):034103.

[Jia et al., 2014] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*.

[Kamilov et al., 2015] Kamilov, U. S., Papadopoulos, I. N., Shoreh, M. H., Goy, A., Vonesch, C., Unser, M., and Psaltis, D. (2015). Learning approach to optical tomography. *Optica*, 2(6):517–622.

[Karri et al., 2017] Karri, S. P. K., Chakraborty, D., and Chatterjee, J. (2017). Transfer learning based classification of optical coherence tomography images with diabetic macular edema and dry age-related macular degeneration. *Biomed. Opt. Express*, 8(2):579–592.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25:1097–1105.

[Langehanenberg et al., 2008] Langehanenberg, P., Kemper, B., Dirksen, D., and von Bally, G. (2008). Autofocusing in digital holographic phase contrast microscopy on pure phase objects for live cell imaging. *Appl. Opt.*, 47(19):D176–D182.

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

[LeCun et al., 1989] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551.

[Liebling and Unser, 2004] Liebling, M. and Unser, M. (2004). Autofocus for digital Fresnel holograms by use of a Fresnelet-sparsity criterion. *J. Opt. Soc. Am. A*, 21(12):2424–2430.

[Lyu et al., 2017] Lyu, M., Yuan, C., Li, D., and Situ, G. (2017). Fast autofocusing in digital holography using the magnitude differential. *Appl. Opt.*, 56(13):F152–F157.

[McElhinney et al., 2007] McElhinney, C. P., McDonald, J. B., Castro, A., Frauel, Y., Javidi, B., and Naughton, T. J. (2007). Depth-independent segmentation of macroscopic three-dimensional objects encoded in single perspectives of digital holograms. *Opt. Lett.*, 32(10):1229–1231.

[Memmolo et al., 2012] Memmolo, P., Iannone, M., Ventre, M., Netti, P. A., Finizio, A., Paturzo, M., and Ferraro, P. (2012). On the holographic 3d tracking of in vitro cells characterized by a highly-morphological change. *Opt. Express*, 20(27):28485–28493.

[Onural and Özgen, 1992] Onural, L. and Özgen, M. T. (1992). Extraction of three-dimensional object-location information directly from in-line holograms using Wigner analysis. *J. Opt. Soc. Am. A*, 9(2):252–260.

[Prentašić et al., 2016] Prentašić, P., Heisler, M., Mammo, Z., Lee, S., Merkur, A., Navajas, E., Beg, M. F., Šarunić, M., and Lončarić, S. (2016). Segmentation of the foveal microvasculature using deep learning networks. *J. Biomed. Opt.*, 21(7):075008.

[Psaltis and Farhat, 1985] Psaltis, D. and Farhat, N. (1985). Optical information processing based on an associative-memory model of neural nets with thresholding and feedback. *Opt. Lett.*, 10(2):98–100.

[Ren et al., 2017] Ren, Z., Chen, N., and Lam, E. Y. (2017). Automatic focusing for multisectional objects in digital holography using the structure tensor. *Opt. Lett.*, 42(9):1720–1723.

[Rezaeilouyeh et al., 2016] Rezaeilouyeh, H., Mollahosseini, A., and Mahoor, M. H. (2016). Microscopic medical image classification framework via deep learning and shearlet transform. *J. Med. Imaging*, 3(4):044501.

[Russakovsky et al., 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet large scale visual recognition challenge. *Int. J. Comput. Vision*, 115(3):211–252.

[Schneider et al., 2016] Schneider, B., Dambre, J., and Bienstman, P. (2016). Fast particle characterization using digital holography and neural networks. *Appl. Opt.*, 55(1):133–139.

[Shortt et al., 2006] Shortt, A. E., Naughton, T. J., and Javidi, B. (2006). Compression of optically encrypted digital holograms using artificial neural networks. *J. Display Technol.*, 2(4):401–410.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[Soulez et al., 2007] Soulez, F., Denis, L., Fournier, C., Thiébaut, É., and Goepfert, C. (2007). Inverse-problem approach for particle digital holography: accurate location based on local optimization. *J. Opt. Soc. Am. A*, 24(4):1164–1171.

[Vikram and Billet, 1984] Vikram, C. S. and Billet, M. L. (1984). Far-field holography at non-image planes for size analysis of small particles. *Applied Physics B*, 33(3):149–153.

[Wang et al., 2014] Wang, H., Cruz-Roa, A., Basavanhally, A., Gilmore, H., Shih, N., Feldman, M., Tomaszewski, J., Gonzalez, F., and Madabhushi, A. (2014). Mitosis detection in breast cancer pathology images by combining handcrafted and convolutional neural network features. *J. Med. Imaging*, 1(3):034003.

[Yevick et al., 2014] Yevick, A., Hannel, M., and Grier, D. G. (2014). Machine-learning approach to holographic particle characterization. *Opt. Express*, 22(22):26884–26890.

 ISBN 978-0-9934207-2-6

Automated Identification of Trampoline Skills Using Computer Vision Extracted Pose Estimation

Paul W. Connolly, Guenole C. Silvestre and Chris J. Bleakley

School of Computer Science, University College Dublin, Belfield, Dublin 4, Ireland.

Abstract

A novel method to identify trampoline skills using a single video camera is proposed herein. Conventional computer vision techniques are used for identification, estimation, and tracking of the gymnast's body in a video recording of the routine. For each frame, an open source convolutional neural network is used to estimate the pose of the athlete's body. Body orientation and joint angle estimates are extracted from these pose estimates. The trajectories of these angle estimates over time are compared with those of labelled reference skills. A nearest neighbour classifier utilising a mean squared error distance metric is used to identify the skill performed. A dataset containing 714 skill examples with 20 distinct skills performed by adult male and female gymnasts was recorded and used for evaluation of the system. The system was found to achieve a skill identification accuracy of 80.7% for the dataset.

1 Introduction

Originating in the 1930s, trampolining became a competitive Olympic sport in Sydney 2000. In competition, athletes perform a routine consisting of a series of skills performed over a number of jumps. The skills are scored by human judges according to the Trampoline Code of Points [FIG, 2017]. Although more explicit and objective judging criteria have been introduced in recent years, the scores awarded can still vary between judges leading to highly contentious final decisions. Eliminating human error by means of reliable, automated judging of trampoline routines is desirable. Herein, we describe a first step towards this goal: a novel automated system for identification of trampoline skills using a single video camera. Identification of skills is necessary prior to judging since different skills are scored in different ways.

While still a challenging problem, identification of trampoline skills from video has been enabled by recent advances in human pose estimation. In [Andriluka et al., 2014], improved accuracy over model-based approaches was achieved with the introduction of convolutional neural network (ConvNet) based estimation. Estimators such as this rely on new ConvNet algorithms coupled with recent gains in GPU performance. In addition, the introduction of larger, more varied general pose datasets [Sapp and Taskar, 2013, Johnson and Everingham, 2011], leveraging crowd-sourced annotation, has vastly increased the quantity of training data available.

To the best of the authors' knowledge, no previous work has been reported on identification of trampolining skills from video. The closest previous work on identification of trampoline skills required the gymnast to wear a full-body motion capture suit containing inertial sensors [Helten et al., 2011]. Wearing special suits is cumbersome and is not allowed in competition due to the strict rules regarding gymnast attire [FIG, 2017]. Previous work on automated judging of rhythmic gymnastics from video was reported in [Díaz-Pereira et al., 2014]. However, their method differs from the method used in this work.

The algorithm proposed herein consists of a number of stages. The bounding box of the gymnast is extracted using conventional image processing techniques. The pose of the athlete is subsequently determined, allowing body orientation and joint angles to be estimated. The angle trajectories over time are compared with those obtained for reference skills. The skill performed is identified as the nearest neighbour in the reference dataset

 ISBN 978-0-9934207-2-6

based on a mean square error metric. The system was evaluated using a large number of video recordings capturing the movements of male and female gymnasts performing trampoline routines. A wide variety of skills, lighting conditions, and backgrounds were recorded. The gymnasts did not wear special clothes or markers. The camera was placed side-on to the performance, in the same position as a human judge.

The structure of the paper is as follows. In section 2, background information on trampolining is given. In section 3, further detail is provided on approaches to analysis of sporting movement and pose estimation using video recordings. The proposed algorithm is described in section 4. Section 5 discusses the experimental procedure and organisation of the dataset. The experimental results and discussion are provided in section 6. Conclusions, including future work, follow in section 7.

2 Background

A trampoline routine consists of a sequence of high, continuous rhythmic rotational jumps performed without hesitation or intermediate straight bounces. The routine should show good form, execution, height, and maintenance of height in all jumps so as to demonstrate control of the body during the flying phase. A competition routine consists of 10 such jumps, referred to, in this work, as skills. For simplicity, a straight bounce is taken to be a skill. A competitor can perform a variable number of straight bounces before the beginning of a routine (so called in-bounces) while an optional straight bounce (out-bounce) can be taken after completing a routine, to control height before the gymnast is required to stop completely.

Skills involve take-off and landing in one of four positions: feet, seat, front, or back. Rotations about the body's longitudinal and lateral axes are referred to as twist and somersault rotations, respectively. Skills combine these rotations with a body shape: tuck, pike, straddle, or straight. These take-off and landing positions and shapes are illustrated in Figure 1.

The score for a performance is calculated as the sum of four metrics: degree of difficulty (tariff), execution, horizontal displacement, and time of flight. Degree of difficulty is scored based on the difficulty of the skill performed. For example, a full somersault is awarded more points than a three-quarter somersault. The tariff assigned is found by a simple look-up based on skill identification. Examples of tariff scores can be seen in Table 2. The execution score is allocated based on how well the skill was judged to be performed. The horizontal displacement and the time of flight are measured electronically using force plates on the legs of the trampoline.

3 Related Work

One of the problems with the capture of trampoline skills is the large performance space. Elite performers can reach up to 10m in height. Tracking such a large volume is prohibitively difficult for many existing motion capture solutions including RGB-D devices such as the Microsoft Kinect.

In [Helten et al., 2011], inertial sensors were used to measure body point acceleration and orientation. The gymnast was required to wear a body suit containing ten inertial measurement units. The sensor data streams were transformed into a feature sequence for classification. For each skill, a motion template was learned. The feature sequence of the unknown trampoline motions were compared with a set of skill templates using a variant of dynamic time warping. The best accuracy achieved was 84.7% over 14 skill types.

A survey of vision-based methods for general human motion representation, segmentation and recognition can be found in [Weinland et al., 2011]. In [Díaz-Pereira et al., 2014], judging of rhythmic gymnastics skills from video was investigated. The movement of the gymnast was tracked using optical flow. Velocity field information was extracted across all frames of a skill and projected into a velocity covariance eigenspace. Similar movements were found to trace out unique, but similar, trajectories. New video recordings were classified based on their distance from reference trajectories. The system's specificity was approximately 85% and the sensitivity was approximately 90% for the 3 skills considered.

 ISBN 978-0-9934207-2-6

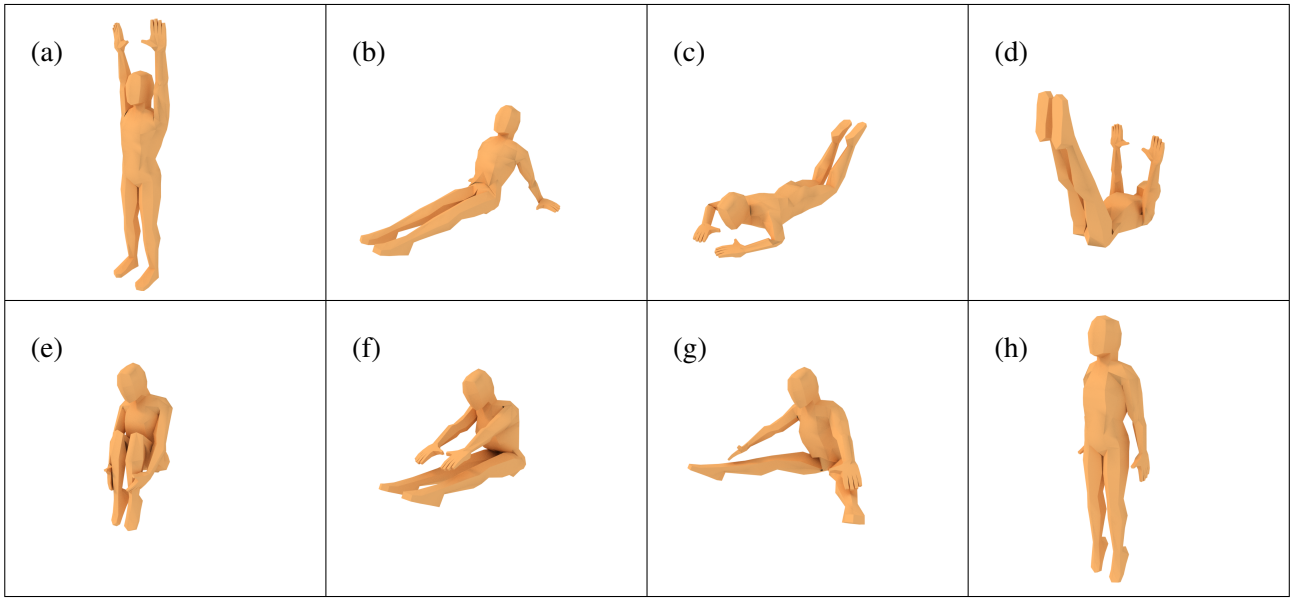


Figure 1: Take-off and landing positions: (a) feet, (b) seat, (c) front and (d) back. Trampoline shapes: (e) tuck, (f) pike, (g) straddle and (h) straight.

Human pose estimation is the process of estimating the configuration of the body, typically from a single image. Robust 2D pose estimation has proven to be a powerful starting point for obtaining 3D pose estimates for human bodies. An overview of the 2D pose estimation problem and proposed methods can be found in [Sigal, 2011, Poppe, 2007]. Model-based methods have been successful for images in which all the limbs of the subject are visible. However, they are unsuitable for the side-on view of a trampoline routine where self-occlusions are inherent. ConvNet based systems do not assume a particular explicit body model since they learn the mapping between image and body pose. These machine learning based techniques provide greater robustness to variations in clothing and accessories than model-based approaches. The MPII benchmark [Andriluka et al., 2014] has been used to access the accuracy of pose estimators. The model-based approach described in [Pishchulin et al., 2013] achieved an accuracy of 44.1% whereas the ConvNet based method proposed in [Newell et al., 2016] achieved 90.9%.

The work described herein differs from previous work in that the system performs skill identification for trampolining using a single monocular video camera. The work takes advantage of recently developed, high accuracy, open source ConvNet based pose estimators. The Stacked Hourglass Network [Newell et al., 2016] and MonoCap [Zhou et al., 2016] methods were selected for estimation and filtering of the 2D pose, respectively.

In the Stacked Hourglass Network, 2D pose estimates are provided by a ConvNet architecture where features are processed across all scales and consolidated to best capture the spatial relationships of the body parts. Repeated steps of pooling and upsampling, in conjunction with intermediate supervision, out-perform previous methods. In MonoCap, 3D pose is estimated via an expectation-maximization algorithm over a sequence of images with 2D pose predictions. Conveniently, the 2D joint location uncertainties can be marginalized out during inference.

4 Proposed Algorithm

The complete algorithm is illustrated in Figure 2. Video is recorded and pre-processed to reduce resolution and remove audio. After pre-processing, the body extraction stage identifies and tracks the convex hull of the athlete over all video frames. The video is segmented according to the detected bounces. The feature extraction stage estimates the pose of the athlete and from this, the body orientation and joint angles in each frame. Based on the extracted feature angles, classification is performed to identify the skill. In our experiments, the accuracy

 ISBN 978-0-9934207-2-6

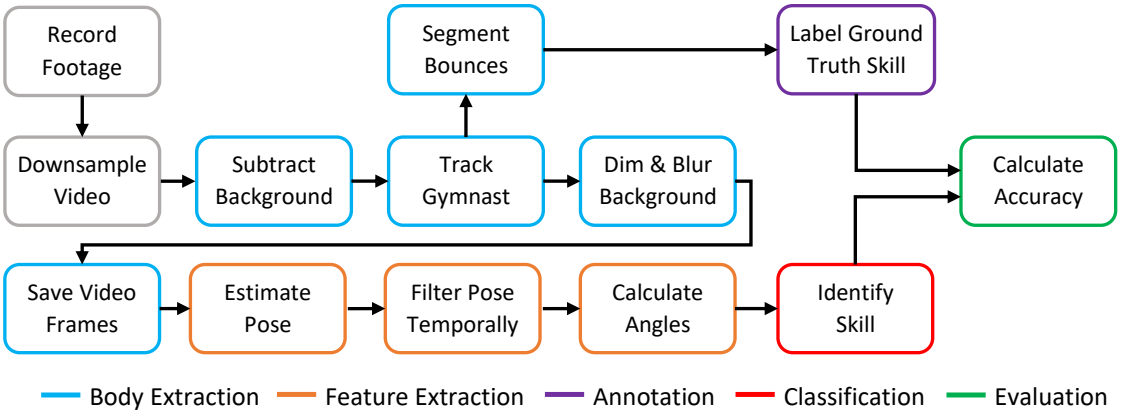


Figure 2: Flow chart illustrating the proposed method.

of the system was evaluated by comparing the detected skills to manually marked ground truth. The algorithm stages are explained in more detail in the following sections.

4.1 Body Extraction

The top of the trampoline is identified based on its hue characteristics and is presented as a best guess on a user interface that allows the position to be fine tuned. The gymnast is tracked by assuming that they are the largest moving object above the trampoline. A background subtractor generates a foreground mask for each frame. All static image components over multiple frames are taken to be part of the background. The camera is assumed to be static without changing focus during the recording. The foreground mask is eroded for one iteration and dilated for ten iterations with a 2×2 kernel. The largest segment of this morphed mask is taken to be the silhouette of the gymnast. The method of moments is used to determine the centroid of this silhouette. The video is segmented into individual skills based on the position of this centroid. A peak detection algorithm identifies the local minima of the vertical position of the centroid. These local minima are taken to indicate the start and end frames of a skill. A threshold is applied to peaks between the local minima to identify the start and end jumps of the routine. The convex hull of the silhouette is used to generate a bounding box for the athlete's image. The bottom of the bounding box is compared to the position of the top of the trampoline to detect the contact phase of a bounce. Examples of the application of this method are shown in Figure 3.

Images of the body are saved for frames in which the athlete is not in contact with the trampoline. The maximum size of the bounding box across all frames of the routine is found. Each image is squarely cropped to this size, centred on the centroid of the gymnast. Based on the extracted foreground mask, the background of each image is blurred and darkened. This helps to reduce the number of incorrect pose estimates.

(a) (b) (c) (d)

Figure 3: Processed images. (a) Original frame. (b) Background model. (c) Foreground mask. (d) Body silhouette and convex hull after erosion and dilation.

 ISBN 978-0-9934207-2-6

i	1, 2	3, 4	5, 6	7, 8	9, 10	11	12
$\theta_i(t)$	R, L Elbow	R, L Shoulder	R, L Hip	R, L Knee	R, L Leg	Torso	Twist

Table 1: Feature angles by name and index.

4.2 Feature Extraction

The Stacked Hourglass Network and MonoCap are used for 2D pose estimation and filtering, respectively. The 2D pose estimator generates pose predictions for 16 joint locations. The 3D pose estimator is then used to filter the 2D pose predictions across the sequence of images. From the smoothed 2D pose, the 2D joint angles and orientation angles that represent the athlete's body position are calculated. These feature angles are denoted as $\boldsymbol{\theta}_i$ for $i \in [1 \ldots M]$ where $M = 12$ is the total number of feature angles. Each of the M feature angles is part of a time series $\theta_i(t)$, where t is the frame number, $t \in [1 \ldots T]$. The angles are listed in Table 1 and example trajectories can be seen in Figure 4.

Twist around the body's longitudinal axis is estimated from the 2D distance between the pose points labelled as right and left shoulder. The shoulder separation in the image is at a maximum when the gymnast's back or front is facing the camera and is approximately zero when sideways to the camera. By finding the maximum 2D separation over the whole routine, the separation can be normalised to a value between 0°and 180°. In this way, the angle does not depend on the size of the performer.

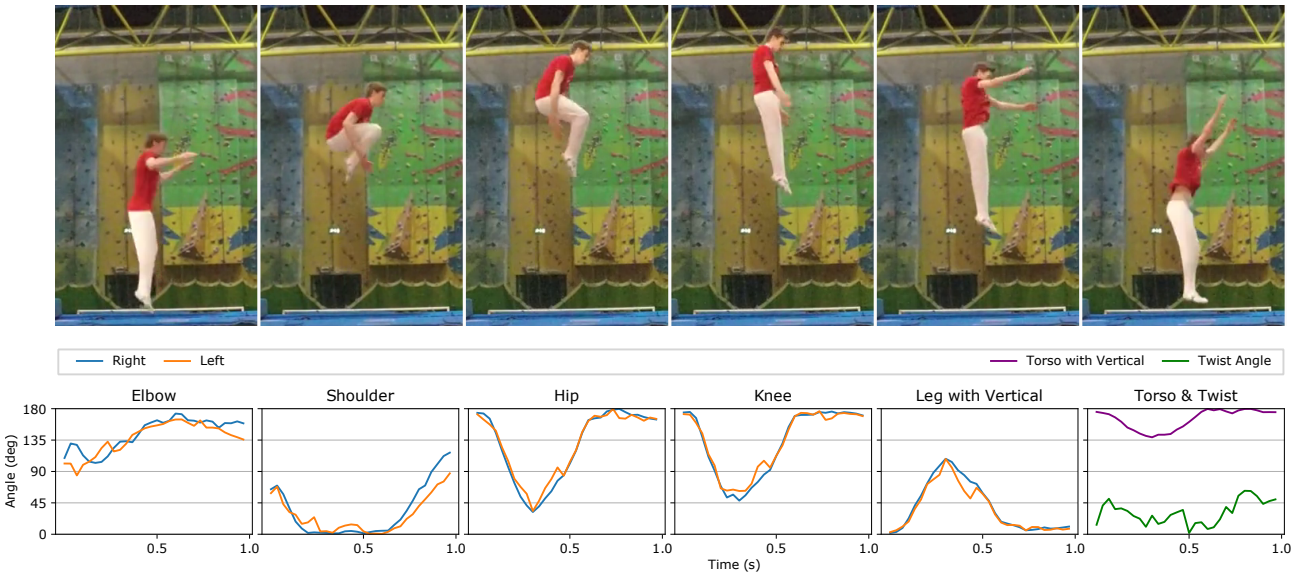


Figure 4: The motion sequence of a tuck jump with the estimated angles shown beneath.

4.3 Classification

The M feature angle trajectories are compared to those in a labelled reference set by calculation of the mean squared error (MSE). The observed skill is identified as equivalent to the reference giving the minimum MSE. The feature angle trajectories of the references $\boldsymbol{\theta}_i^R(t)$ are aligned through re-sampling by means of interpolation so as to have the same number of data points T as the observed angle trajectory $\boldsymbol{\theta}_i^O(t)$.

$$\text{MSE} = \frac{1}{TM} \sum_{t=1}^{T} \sum_{i=1}^{M} \left(\boldsymbol{\theta}_i^O(t) - \boldsymbol{\theta}_i^R(t) \right)^2 \tag{1}$$

 ISBN 978-0-9934207-2-6

5 Experimental Procedure

5.1 Data Acquisition

The procedure for data collection was submitted to and approved by the UCD Office of Research Ethics. Videos of routines were recorded at training sessions and competitions of the UCD Trampoline Club. Consent was sought from members of the UCD Trampoline Club prior to recording video for the purposes of the project. Routines were collected in UCD Sports Centre under normal sports hall lighting conditions. The background was not modified, typically consisting of a brick wall or nets.

The routines were recorded at a resolution of 1920×1080 at 30 frames per second (fps) using a consumer grade camera with a shutter speed of 1/120 to reduce motion blur. The camera was positioned at the typical location and viewing angle of the judging panel. All bounces were within the field of view of the camera. The video was subsequently downsampled to 896×504, maintaining a 16:9 aspect ratio, and audio was removed. These steps significantly reduced data file size and processing time while maintaining usable resolution. The videos were manually annotated with ground-truth labels by means of a custom built web interface.

5.2 Datasets

The resulting dataset consists of 49 routines by 30 adult athletes, 18 male and 31 female, totalling 23 minutes of video. This contained 28 distinct skills and 771 skill examples. The names and distribution of these skills are summarised in Table 2.

The accuracy of the identification algorithm was tested using 10-fold cross validation. The skills with fewer than 10 examples were not included in the test, leaving $N = 20$ distinct skills.

In each iteration of the evaluation, a subset of 10 examples of each skill were randomly selected from the database. Each subset was split evenly to give the number of reference examples $S_r = 5$ and the number of test examples $S_t = 5$. The total size of the reference set was $N \times S_r = 100$ skill examples. The test set was of the same size. The average accuracy over 20 iterations of the evaluation is reported herein.

6 Results and Discussion

The average accuracy of the system was 80.7% for the 20 distinct skills listed as included in classification in Table 2. The confusion matrix for the experiment is shown in Figure 5.

It was noted that subject identification can sometimes incorrectly focus on people in the background, particularly during seat, front and back landings, when the gymnast becomes obscured by the trampoline bed. This causes errors in trampoline contact detection resulting in frames without an obvious subject being presented to the pose estimator. The resulting angles are not representative of the skill performed. This can also cause errors in jump segmentation due to incorrect centroid extraction. Jump segmentation failed in 2 cases.

Significant confusion in skill identification occurs between FPF (pike jumps shown in Figure 1f) and FSF (straddle jumps shown in Figure 1g). From a side-on view, it is difficult to distinguish these movements. Another area of confusion is between the tuck and pike shape of the Barani skill (BRI). The features which distinguish these shapes are the angles of the hip and knees. The tuck shape in this skill is often performed loosely. This results in the angle of the hip being similar to that of the pike shape. For identification, the angle of the knees becomes the deciding feature and may be overwhelmed by noise from other features.

Use of a support vector machine might improve classification accuracy. For example, the difficulty in estimating the wrist and ankle joints for the 2D pose estimator can lead to noise in the angles for the elbows and knees. Weighting these features as less important might improve overall accuracy.

 ISBN 978-0-9934207-2-6

Skill Name	Code	Tariff	Occurrences
Straight Bounce	F0F	0.0	286
Tuck Jump	FTF	0.0	58
Pike Jump	FPF	0.0	40
Straddle Jump	FSF	0.0	42
Half Twist Jump	F1F	0.1	18
Full Twist Jump	F2F	0.2	19
Seat Drop	F0S	0.0	13
Half Twist to Seat Drop	F1S	0.1	10
Seat Half Twist To Seat	S1S	0.1	24
To Feet from Seat	S0F	0.0	11
Half Twist to Feet from Seat	S1F	0.1	24
Front Drop	F0R	0.1	4†
To Feet from Front	R0F	0.1	5†
Back Drop	F0B	0.1	10
To Feet from Back	B0F	0.1	8†
Half Twist to Feet from Back	B1F	0.2	12
Front Somersault (Tuck)	FSSt	0.5	4†
Front Somersault (Pike)	FSSp	0.6	7†
Barani (Tuck)	BRIt	0.6	24
Barani (Pike)	BRIp	0.6	19
Barani (Straight)	BRIs	0.6	9†
Crash Dive	CDI	0.3	18
Back Somersault (Tuck)	BSSt	0.5	28
Back Somersault (Pike)	BSSp	0.6	18
Back Somersault (Straight)	BSSs	0.6	30
Back Somersault to Seat (Tuck)	BSTt	0.5	10
Lazy Back	LBK	0.3	3†
Cody (Tuck)	CDYt	0.6	3†
Back Half	BHA	0.6	1†
Barani Ball Out (Tuck)	BBOt	0.7	7†
Rudolph / Rudi	RUI	0.8	3†
Full Front	FFR	0.7	1†
Full Back	FUB	0.7	2†

Table 2: Skill dataset. (†excluded from classification)

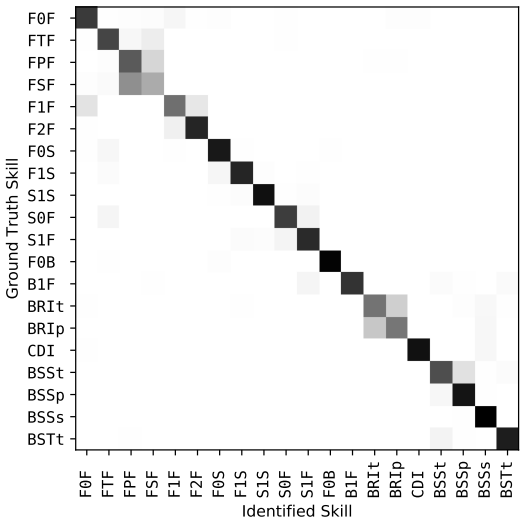


Figure 5: Confusion matrix showing the relative errors for each skill. This is the average of 20 iterations of 10-fold cross validation.

It is likely that accuracy could be improved by increasing the amount of data. Current pose estimation algorithms take a single image as input. It seems likely that performance could be improved by tracking pose over a video sequence. Adding a second video camera pointed towards the front of the gymnast would likely improve accuracy by allowing greater discrimination of motion parallel to the axis between the subject and the first camera. However, there are issues regarding the extra user effort in setting up the second camera and in synchronisation of the two devices. Modern trampoline judging systems incorporate force plates for detection of the centrality of landing on the trampoline bed. Fusing such information with the video data could possibly also result in improved accuracy.

Body extraction was performed at 15 fps on a 2 core Intel i7–3517U 2.4 GHz CPU. Estimation of pose using the Stacked Hourglass Network ran at 20 fps on an Ubuntu 16.04 with an Nvidia Titan X (Pascal) GPU and a 4 core Intel i7–920 2.67 GHz CPU with default parameter settings. Execution of the MonoCap algorithm ran at 0.3 fps on the same machine also with default parameters.

7 Conclusion

A system for identifying trampolining skills using a single monocular video camera was developed. The system incorporated algorithms for background subtraction, erosion and dilation, pose estimation, pose filtering and

 ISBN 978-0-9934207-2-6

classification. The system was found to provide 80.7% accuracy in identifying the 20 distinct skills present in a dataset contain 712 skill examples.

In future work, we plan to extend the classification algorithms to perform automated execution judging.

References

[Andriluka et al., 2014] Andriluka, M., Pishchulin, L., Gehler, P., and Schiele, B. (2014). 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3686–3693.

[Díaz-Pereira et al., 2014] Díaz-Pereira, M. P., Gómez-Conde, I., Escalona, M., and Olivieri, D. N. (2014). Automatic recognition and scoring of olympic rhythmic gymnastic movements. *Human Movement Science*, 34:63–80.

[FIG, 2017] FIG (2017). Trampoline Code Of Points. [Accessed 2017-03-30].

[Helten et al., 2011] Helten, T., Brock, H., Müller, M., and Seidel, H.-P. (2011). Classification of trampoline jumps using inertial sensors. *Sports Engineering*, 14(2):155–164.

[Johnson and Everingham, 2011] Johnson, S. and Everingham, M. (2011). Learning effective human pose estimation from inaccurate annotation. In *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1465–1472.

[Newell et al., 2016] Newell, A., Yang, K., and Deng, J. (2016). Stacked Hourglass Networks for Human Pose Estimation. *CoRR*, abs/1603.06937.

[Pishchulin et al., 2013] Pishchulin, L., Andriluka, M., Gehler, P., and Schiele, B. (2013). Poselet conditioned pictorial structures. In *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 588–595.

[Poppe, 2007] Poppe, R. (2007). Vision-based human motion analysis: An overview. *Computer Vision and Image Understanding*, 108(1–2):4–18. Special Issue on Vision for Human-Computer Interaction.

[Sapp and Taskar, 2013] Sapp, B. and Taskar, B. (2013). MODEC: Multimodal Decomposable Models for Human Pose Estimation. In *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3674–3681.

[Sigal, 2011] Sigal, L. (2011). Human pose estimation. [Accessed 2017-03-30].

[Weinland et al., 2011] Weinland, D., Ronfard, R., and Boyer, E. (2011). A survey of vision-based methods for action representation, segmentation and recognition. *Computer Vision and Image Understanding*, 115(2):224–241.

[Zhou et al., 2016] Zhou, X., Zhu, M., Leonardos, S., Derpanis, K. G., and Daniilidis, K. (2016). Sparseness Meets Deepness: 3D Human Pose Estimation from Monocular Video. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4966–4975.

Visual Lecture Summary Using Intensity Correlation Coefficient

Solomon E. Garber, Luka Milekic, Nick Moran, Aaditya Prakash, Antonella Di Lillo and James A. Storer

Department of Computer Science, Brandeis University, Waltham, Massachusetts

Abstract

We present an automatic technique for creating a video summary of chalkboard and whiteboard lectures with the speaker removed, and for generating a set of slides containing all of the written content but without the lecturer present. Our system works by continuously subtracting the lecturer from the video feed using a region based correlation feature. The final presentation slides are extracted from this new version of the video that no longer contains the speaker.

Keywords: Lecture Recording, Video Summarization, Video Indexing, Foreground Subtraction

1 Introduction

Online resources are increasingly used by students and educators. Pre-recorded videos of lectures containing handwritten mathematical content are a popular educational tool, and traditional whiteboards are still the preferred format for displaying this content in the classroom [Vemulapalli and Hayes, 2014]. However, these videos can be cumbersome to navigate. Thumbnails, video titles, and high speed video scrubbing fail to offer a useful summary, so locating specific content can be time consuming. There is a need for tools which aid in the organization of this content. Specialized hardware such as electronic whiteboards are expensive and require custom software, and do not address the navigability of the many videos already available online. We present an automatic technique for creating a video summary of a chalkboard (or whiteboard) lecture with the foreground removed (e.g., remove the professor who is lecturing using the chalkboard), and for generating candidate key frames for a slideshow lecture summary.

Some attempts have been made to make educational videos easier to navigate. In [Kannan and Andres, 2010], a system creates a database of tags to search screen captured videos, but does not provide for a more easily searchable visual representation. [Yang et al., 2012a] presents a method for extracting slides from such videos, and use OCR to allow text based search on the slides. [Yang et al., 2011], [Yang et al., 2012b] and [Tuna et al., 2015] use OCR to make video lectures searchable by content, while [Vemulapalli and Hayes, 2014] and [Yang et al., 2014] combine OCR with speech recognition for the same purpose. [Yadid and Yahav, 2016] uses OCR to extract code from programming instruction videos. [Pratusevich, 2015] and [Liao et al., 2015] present methods of localizing written content and lecturer within lecture videos. [Wang et al., 2003] and [Eberts et al., 2015] align powerpoint slides with lecture videos. [Shin et al., 2015] and [Monserrat et al., 2013] generate lecture notes from computer screen captures; they refer to such screen captures as "blackboard style videos". In contrast here we address full length videos of a speaker lecturing in front of an actual blackboard, where there is noise in the video capture process, changes in ambient lighting, and occlusion of the blackboard by the lecturer. [Lin et al., 2004] presents a method for segmenting lectures by topic, given a transcript. In [Prabhu et al., 2008] and [He and Zhang, 2007], a method for removing a lecturer from a whiteboard lecture video is presented, but there is no attempt to generate a set of slides from the output. In [He et al., 2003], slides from whiteboard videos are generated, but without quantitative evaluation. [Choudary and Liu, 2007] produces bitmask slides from a chalkboard lecture video; these bitmasks do not contain color information, and evaluation is done by hand.

 ISBN 978-0-9934207-2-6

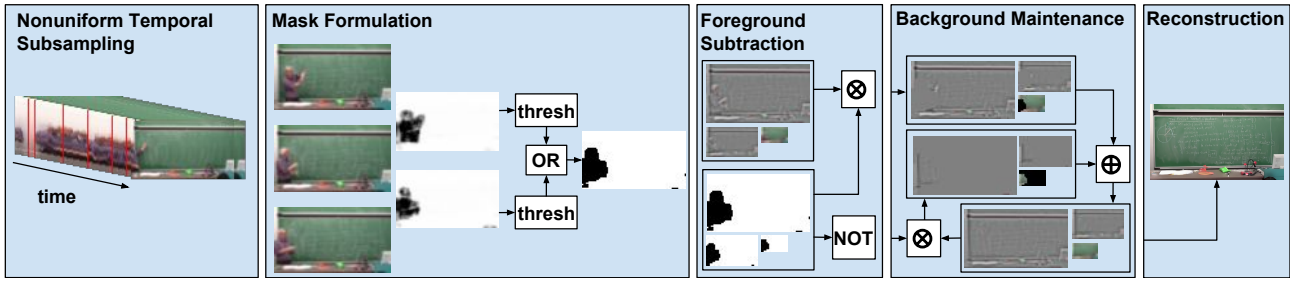


Figure 1: Our system.

Some attempts have been made to create continuous video summaries. In [Bennett and McMillan, 2007] a method is proposed of generating time-lapse videos from continuous feeds which summarize the changes between frames, as well as a method using non-uniform sampling in time to capture all important changes. Projects like Microsoft hyper-lapse [Kopf et al., 2014] warp videos in time to create a smoothly accelerated version of the video. Some attempts such as [Ejaz et al., 2012] and [Mundur et al., 2006] select key frames from the video to create a static video summary. In [Gianluigi and Raimondo, 2006] frame differences in a video feed are processed to select key frames for video summary. In [Mehmood et al., 2014] and [Ngo et al., 2003] attention models are used to predict the salient parts. These various summarization methods seek to capture the changes that occur in a video. Background is generally assumed to be uninteresting in these techniques, as it accounts for a very small portion of the changes in the videos. The majority of changes (e.g. speaker movement) from frame to frame in our videos have nothing to do with the content of the blackboard. We identify key frames in a video that we have modified to remove the speaker.

Much work has also been done on the problem of detecting and deleting the background from footage for applications such as video surveillance and object tracking. However, little has been written on the subject of removing the foreground and summarizing the background. In [Rubinstein et al., 2011] time-lapse videos are denoised by removing high speed changes and small jittery motions, under the assumption that the information in such videos is concentrated in low frequency changes. Much like time-lapse videos, most of the interesting content of chalkboard lecture videos is contained in gradual changes to the background image. Here we present an automatic technique for creating a video summary of the chalkboard lecture with the foreground (the speaker) removed, and for generating key frames for a slide show lecture summary.

2 Proposed Method

We process the input video in multiple stages, shown in Fig. 1. In the sampling stage described in Sec. 3, we take a nonuniform sampling from the input feed based on absolute frame difference. Next, in the foreground subtraction stage described in Sec. 4 and 5, we calculate the localized correlation coefficient computed over a sliding window, and threshold the results to obtain a foreground mask. We use the mask to update coefficients of a background image pyramid as explained in [Burt and Adelson, 1983] and described in Sec. 6. This pyramid is then reconstructed and added to the output stream. We then use a combination of edge detection and localized correlation to detect when a portion of the board has been erased or written over in the key frame detection stage described in Sec. 7. We describe our evaluation metrics and results in Sec. 8.

3 Nonuniform Temporal Subsampling

The changes that we wish to preserve are persistent, and failure to detect all foreground regions can lead to artifacts in the reconstructed background image. For this reason, a temporally subsampled version of the original feed is used. This sampling is done non-uniformly in time as a pre-processing step to prevent correlations created by temporarily static foreground regions. In order to avoid costly comparisons between every pair of

 ISBN 978-0-9934207-2-6

frames, we use a greedy heuristic approach to sampling in time. We perform two passes over the video. In the first pass we find the mean absolute difference between consecutive frames in a subsampled feed (for experiments reported here, approximately 1 frame for every 3 seconds of video). In the second pass, we sample from the original video by taking the first frame, and then only sample subsequent frames if the sum of the absolute differences between the current frame and the previously sampled frame is greater than the mean difference from the first pass. Because the majority of intensity and color changes are due to foreground motion, this non-uniform feed ensures that adjacent samples are uncorrelated in regions of motion.

4 Foreground Subtraction

Background subtraction is a common first step in many object tracking and security applications. Typically foreground masks are generated based on some pixelwise distance from a background model, or simply the pixelwise distance between consecutive frames in the input video.

Background subtraction approaches assume that the foreground is the interesting part of the video. However, as in [Rubinstein et al., 2011], we consider the foreground as noise and the background as signal, and seek a method which can preserve medium term changes in the background while deleting all foreground objects, such as the lecturer or students temporarily occluding the blackboard. Unlike [Rubinstein et al., 2011], however, we assume that the relevant background regions will not move between frames because both the camera and blackboard are assumed to be stationary. This allows us to avoid the costly message passing scheme used in [Rubinstein et al., 2011] to compute spatiotemporal displacement fields in favor of temporal displacements which can be computed using foreground masks. We therefore model the first stage of our process as foreground subtraction. Similar to background subtraction applications, the output of the foreground subtraction stage is a compact and easily scanned representation of the input video.

5 Mask Formulation

We model the background as regions where the shape of the intensity surface doesn't change from frame to frame in the subsampled feed. We detect such changes in shape using the regional cross correlation between input frames. We process the input video sequentially, obtaining a mask for every pair of adjacent frames. We treat each x, y, t pixel in the input video I as a sample from a population. At each time t, at each pixel $p = I(x, y, t)$, we compute the correlation coefficient for the intensity values over the region

$$\mathbf{N}(x, y) = \{(x_0, y_0) | x-\delta \leq x_0 \leq x+\delta, y-\delta \leq y_0 \leq y+\delta\} \quad (1)$$

between the frame at t and $t-1$, and t, $t+1$, and threshold those correlation coefficients. Let $\mu_{x,y,t}$ denote the average intensity of the frame at time t over the region $\mathbf{N}(x, y)$, and $Var(x, y, t)$ denote the variance of the intensity over the same region. We compute the local intensity correlation at each point $I_{x,y,t}$ as:

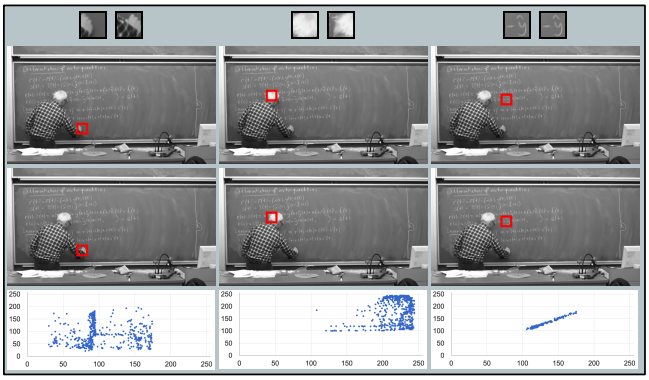


Figure 2: Three square sections taken from consecutive images in the temporally subsampled lecture. The first two blocks contain motion and therefore have weak and even negative correlations. The third does not move and thus the two blocks are strongly correlated.

$$\frac{\frac{1}{(2\delta)^2}\sum_{\mathbf{N}(x,y)}(I_{x_0,y_0,t} * I_{x_0,y_0,t+1}) - \mu_{x,y,t} * \mu_{x,y,t+1}}{\sqrt{Var(x, y, t)} * \sqrt{Var(x, y, t+1)}} \quad (2)$$

$$Var(x, y, t) = \frac{1}{(2\delta)^2} \sum_{\mathbf{N}(x,y,t)} I^2_{x_0,y_0,t} - \mu^2_{x,y,t} \quad (3)$$

 ISBN 978-0-9934207-2-6

This blockwise comparison is shown in Fig. 2 and an example of a surface obtained in this manner is depicted in Fig. 1. This metric (similar to the spatially modulated normalized vector distance used in [Matsuyama, 2000]) is sensitive to the size of the sampling window. However, as can be seen from equations 2 and 3, the computations can be done with a uniform blur kernel, which is separable and can therefore be computed in time proportional to the width of the window size δ rather than the square of the width as would be required in the naive implementation. This coupled with the fact that computations are entirely localized and can thus be massively parallelized, allows us to do the background maintenance quickly even with large spatial support for the correlation computation (we used a window size of $\delta = 12$ pixels). As a post-processing step, we apply a morphological erosion of the mask to close up holes in flat foreground regions and remove shadows.

6 Background Reconstruction

In most background subtraction applications, a statistical model of local features is kept for each pixel to allow for change detection. This model must be constantly maintained to accommodate dynamic backgrounds. In this sense our problem can be seen as continuous background maintenance and reconstruction. Simple low pass filters such as FIR (finite impulse response) and IIR (infinite impulse response) filters, commonly used for the task of background maintenance, tend to create ghosts in the background given any reasonable filter support. If the support is too large, important background information can take too long to appear in the smooth video. Also, since the speaker is almost always somewhere in the shot and often stays in a similar place for long amounts of time, even a median filter will output entire regions of misclassified pixels. For this reason we classify each frame into still regions and motion regions and use masks to block out any the locations in the frame where motion is detected. We update the background only in regions without motion, under the assumption that the foreground is never stationary. If the foreground does not move then large parts of the foreground will be incorporated into the estimated background image. To address this issue we process a subsampled version of the input video as described in Sec. 3. To initialize the background, we run this process without writing to the output stream until the entire frame has been updated and then start over from the beginning. In order to prevent edge artifacts from appearing at mask boundaries we store the background as a Laplacian pyramid, a tensor containing edges at different resolutions which can be used to reconstruct a full resolution image. We create a Gaussian pyramid from each mask, and use the mask pyramid to determine which coefficients to update in the background pyramid. This suppresses edge artifacts at the boundaries of the mask, especially due to aliasing introduced in the temporal subsampling process caused by light and shadow changes. In the last phase of processing, we use the edges in each frame to locate the key frames, so mask induced edge artifacts would be both distracting for the end user and detrimental to our process.

7 Slide Selection and Erasure Detection

Once the background video has been obtained we can create a useful and compact summary of the lecture by finding the frames in the background video that contain a blackboard full of writing, prior to some major erasure event. Our goal is to save all the slides directly before part of the board is erased. We use the intensity correlation described in Sec. 5 between consecutive frames in the background video to detect when the board is erased. When corresponding regions in consecutive frames fail to correlate the possible causes are information added, erased, or modified. We apply a Sobel filter to the first image in the pair and find the sum of edges in the changed regions. When the unmatched edges surpass a low threshold, we determine that something was erased. We save the first image in the pair, and sum the edges in the unmatched regions of the second image in the pair to start a running tally. Detected erasures will not trigger another slide to be saved unless the sum of the edges added since the last saved slide exceeds a lower limit, indicating that the lecturer has resumed writing on the board. Although this method successfully saves the slides we want, it can include extra slides containing no unique information. An example of such a slide is shown in Fig. 4 which was chosen by our algorithm because of the way we initialize the background. We do a final pass over the set of slides, automatically removing slides

 ISBN 978-0-9934207-2-6

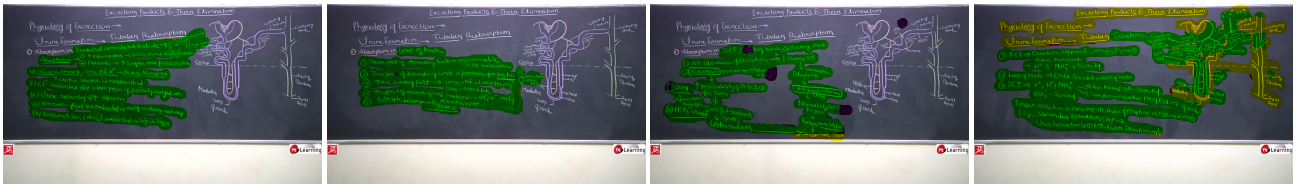


Figure 3: Ground truth slides from Fig. 4 colored based on the redundancy in our output. Best viewed in color.

whose edges can all be matched elsewhere in the set. The extra slide in Fig. 4 is not currently removed by this final pass because part of the diagram on the right hand side was written over and failed to match the similar regions in subsequent slides, a problem which we are addressing in current research.

8 Evaluation of Results

In this section we present a ground truth against which results may measured, together with a baseline method to which we compare our improved results. In order to evaluate our results, we selected recorded lecture videos posted to YouTube from a variety of instructors and institutions where handwritten material was presented on a whiteboard or blackboard, and where both the camera and the board remained in a fixed position over the course of the lecture. Ground truth slides were generated by hand (where a human watched each video in the test set), and every time the lecturer erased something in order to make room on the board, several frames from the video prior to the erasure were sampled and the regions of these frames where the background is visible were blended together (using the pyramid blending described in Sec. 6). The left column of Fig. 4 is an example of a ground truth slide set. Ground truth masks were constructed by marking the important regions of each ground truth slide (anything written on the board is behind just one of the masks). Fig. 3 shows how these masks can be used to evaluate a proposed slide set, where each slide in the ground truth slide set has been colored according to the number of times it appeared in our proposed slideshow in comparison with the ground truth, where purple regions were not matched to any slide, green regions occur at least once but no more than the ground truth, yellow regions one or two more times than the ground truth, and red regions more than two times more than the ground truth. The baseline takes a median filter over 50 frames sampled at 2 frames every 3 seconds, where a slide is taken when the sum of the edges from a Canny edge detector is a local maximum. Generated slideshows by both the baseline and our improved method are judged based on two metrics, $\mathscr{R}$ = the ratio of the size of a proposed slide set to the ground truth, and $\mathscr{P}$ = the percent of masked pixels with no match in the proposed

Table 1: Our results compared with the baseline described in Section 8. GT: Ground Truth. $\mathscr{R}$ indicates the number of slides generated by each method divided by the number of slides in our ground truth minimal slide set. $\mathscr{P}$ indicates the percent of masked pixels from the ground truth not matched to any slide in a generated slide set. # Slides indicates the number of slides chosen in our ground truth slide set. The baseline method against which we compare our results is described in Section 8, as well as the choice of videos.

	Lecture	$\mathscr{R}$		$\mathscr{P}$		**# Slides**
	id	ours	baseline	ours	baseline	GT
Whiteboard	1	1.15	0.42	4.6	34.6	26
	2	1.09	1.00	2.7	5.6	11
	10	3.50	2.70	2.5	4.6	10
	12	2.89	2.00	3.6	6.7	9
	15	1.46	0.62	2.4	25.4	13
Blackboard	3	1.25	1.50	1.0	4.4	4
	4	1.60	1.40	0.0	1.6	5
	5	2.25	1.50	1.2	5.6	4
	6	1.00	0.77	7.9	24.4	9
	7	1.33	0.72	5.1	27.4	18
	8	1.60	1.20	0.2	9.4	15
	9	1.22	0.77	2.5	27.3	18
	11	1.33	1.67	1.0	4.2	3
	13	1.4	1.00	3.4	25.3	5
	14	1.33	1.00	0.5	12.9	3
	16	1.17	0.83	1.0	24.4	6

 ISBN 978-0-9934207-2-6

Figure 4: A sample input frame (top left), ground truth slide set (left) and corresponding slides produced by our system (right). Our system produced an extraneous slide for this video, which did not correspond to any slide from the ground truth (top right). Best viewed in color.

 ISBN 978-0-9934207-2-6

slide set. Our results are reported in Table 1. In all videos tested, our method selected a set of slides containing over 90%, and in almost all cases over 95% of the pixels flagged in the ground truth as important. Our algorithm prioritizes completeness of the selected slide set over brevity, and yet the generated slide sets tended to stay within 50% of the minimal number of slides, rarely as much as double and in only one case did our slide set exceed triple the minimal number of slides from the human generated ground truth.

References

[Bennett and McMillan, 2007] Bennett, E. P. and McMillan, L. (2007). Computational time-lapse video. In *ACM Transactions on Graphics (TOG)*, volume 26, page 102.

[Burt and Adelson, 1983] Burt, P. J. and Adelson, E. H. (1983). A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics (TOG)*, 2(4):217–236.

[Choudary and Liu, 2007] Choudary, C. and Liu, T. (2007). Summarization of visual content in instructional videos. *IEEE Transactions on Multimedia*, 9(7):1443–1455.

[Eberts et al., 2015] Eberts, M., Ulges, A., and Schwanecke, U. (2015). Amigo-automatic indexing of lecture footage. In *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, pages 1206–1210. IEEE.

[Ejaz et al., 2012] Ejaz, N., Tariq, T. B., and Baik, S. W. (2012). Adaptive key frame extraction for video summarization using an aggregation mechanism. *Journal of Visual Communication and Image Representation*, 23(7):1031–1040.

[Gianluigi and Raimondo, 2006] Gianluigi, C. and Raimondo, S. (2006). An innovative algorithm for key frame extraction in video summarization. *Journal of Real-Time Image Processing*, 1(1):69–88.

[He et al., 2003] He, L.-w., Liu, Z., and Zhang, Z. (2003). Why take notes? Use the whiteboard capture system. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 5, pages V–776. IEEE.

[He and Zhang, 2007] He, L.-W. and Zhang, Z. (2007). Real-time whiteboard capture and processing using a video camera for remote collaboration. *IEEE Transactions on Multimedia*, 9(1):198–206.

[Kannan and Andres, 2010] Kannan, R. and Andres, F. (2010). Towards automated lecture capture, navigation and delivery system for web lecture on demand. *International Journal of Innovation in Education*, 1(2):204–212.

[Kopf et al., 2014] Kopf, J., Cohen, M. F., and Szeliski, R. (2014). First-person hyper-lapse videos. *ACM Transactions on Graphics (TOG)*, 33(4):78.

[Liao et al., 2015] Liao, H.-C., Pan, M.-H., Chang, M.-C., Lin, K.-W., et al. (2015). An automatic lecture recording system using pan-tilt-zoom camera to track lecturer and handwritten data. *International Journal of Applied Science and Engineering (IJASE) 13 (1)*, pages 1–18.

[Lin et al., 2004] Lin, M., Nunamaker, J. F., Chau, M., and Chen, H. (2004). Segmentation of lecture videos based on text: a method combining multiple linguistic features. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, pages 9–pp. IEEE.

[Matsuyama, 2000] Matsuyama, T. (2000). Background subtraction for non-stationary scenes. In *Proc. 4th Asian Conference on Computer Vision, 2000*, pages 662–667.

[Mehmood et al., 2014] Mehmood, I., Sajjad, M., and Baik, S. W. (2014). Visual attention based extraction of semantic keyframes. *Advances in Information Science and Applications*, 1.

[Monserrat et al., 2013] Monserrat, T.-J. K. P., Zhao, S., McGee, K., and Pandey, A. V. (2013). Notevideo: facilitating navigation of blackboard-style lecture videos. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1139–1148. ACM.

[Mundur et al., 2006] Mundur, P., Rao, Y., and Yesha, Y. (2006). Keyframe-based video summarization using delaunay clustering. *International Journal on Digital Libraries*, 6(2):219–232.

[Ngo et al., 2003] Ngo, C., Ma, Y., and Zhang, H. (2003). Automatic video summarization by graph modeling. In *Computer Vision, 2003. Proc. 9th IEEE International Conference on*, pages 104–109.

[Prabhu et al., 2008] Prabhu, N., Kumar, R. P., Punitha, T., and Srinivasan, R. (2008). Whiteboard documentation through foreground object detection and stroke classification. In *Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on*, pages 336–340. IEEE.

[Pratusevich, 2015] Pratusevich, M. (2015). *Edvidparse: Detecting people and content in educational videos*. PhD thesis, Massachusetts Institute of Technology.

[Rubinstein et al., 2011] Rubinstein, M., Liu, C., Sand, P., Durand, F., and Freeman, W. T. (2011). Motion denoising with application to time-lapse photography. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 313–320. IEEE.

[Shin et al., 2015] Shin, H. V., Berthouzoz, F., Li, W., and Durand, F. (2015). Visual transcripts: lecture notes from blackboard-style lecture videos. *ACM Transactions on Graphics (TOG)*, 34(6):240.

[Tuna et al., 2015] Tuna, T., Joshi, M., Varghese, V., Deshpande, R., Subhlok, J., and Verma, R. (2015). Topic based segmentation of classroom videos. In *Frontiers in Education Conference (FIE), 2015. 32614 2015. IEEE*, pages 1–9. IEEE.

[Vemulapalli and Hayes, 2014] Vemulapalli, S. and Hayes, M. (2014). Audio-video based character recognition for handwritten mathematical content in classroom videos. *Integrated Computer-Aided Engineering*, 21(3):219–234.

[Wang et al., 2003] Wang, F., Ngo, C.-W., and Pong, T.-C. (2003). Synchronization of lecture videos and electronic slides by video text analysis. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 315–318. ACM.

[Yadid and Yahav, 2016] Yadid, S. and Yahav, E. (2016). Extracting code from programming tutorial videos. In *Proceedings of the 2016 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software*, Onward! 2016, pages 98–111.

[Yang et al., 2012a] Yang, H., Gruenewald, F., and Meinel, C. (2012a). Automated extraction of lecture outlines from lecture videos-a hybrid solution for lecture video indexing. In *CSEDU (1)*.

[Yang et al., 2012b] Yang, H., Oehlke, C., and Meinel, C. (2012b). An automated analysis and indexing framework for lecture video portal. In *International Conference on Web-Based Learning*, pages 285–294. Springer.

[Yang et al., 2014] Yang, H., Quehl, B., and Sack, H. (2014). A framework for improved video text detection and recognition. *Multimedia Tools and Applications*, 69(1):217–245.

[Yang et al., 2011] Yang, H., Siebert, M., Luhne, P., Sack, H., and Meinel, C. (2011). Lecture video indexing and analysis using video ocr technology. In *Signal-Image Technology and Internet-Based Systems (SITIS), 2011 Seventh International Conference on*, pages 54–61. IEEE.

Automatic Tracking System for Event Detection and Classification in Tennis

Pushyami Rachapudi[1], Abhishek Sharma[2], and Navjyoti Singh[1]

[1]*Centre for Exact Humanities, IIIT Hyderabad, India*
[2]*International Institute of Information Technology(IIIT) Hyderabad, India*

Abstract

Sport video analysis has caught a lot of attention in recent times, especially for tennis. Most of the current systems for player detection, ball tracking and automatic annotation use broadcast tennis videos. This paper is an attempt to make technology an integral part of learning a sport. In this paper, domain knowledge is used to formalise the definition of shot and also to help select features to be extracted. An ergonomic, easy to replicate documentation and event detection system for the purpose of coaching is put forward which describes an innovative approach of generalised feature based motion tracking algorithm towards automatic player detection and an improvised object-and-trajectory based algorithm for ball tracking in real world situations. These tracking algorithms are used to extract selected shot features which are then classified and validated using SVM and KNN techniques. The features extracted along with the accuracies are: spin of the ball (83.85%), trajectory height (95.56%), position (94.65%), number of bounces (98.93%) and shot prediction (88.75%).

Keywords: Event Detection, Video Tracking, Machine Learning, Player and Ball Detection, Sport Video Analysis

1 Introduction

In recent times, rapid growth of video databases lead to a tremendous increase in sport video analysis and automatic annotation systems. Most of these videos are broadcast videos of tennis tournaments telecasted for audience's entertainment. Analysing this video data will enable better understanding of the game and can also act as a learning tool in which a player can introspect their game. This paper is an attempt to introduce ergonomic and easy to implement methods of tennis video analysis into the field of coaching. This not only helps the coach to teach better but also helps a player learn better with the help of quantitative analysis of the game. This paper is divided into three major parts:

- Domain Knowledge Formalisation: This part discusses about a formalised notation of a tennis shot and its attributes. This notation, not only helps in defining and identifying shots but also decomposes huge event detection problem into smaller components. Based on this notation, a few features are selected in order to automatically detect events.
- Computer Vision Modules: This paper talks about three modules which are implemented in order to detect the above selected events. *Player detection Module* which discusses about our implementation of a generalised feature based motion tracking algorithm to track and detect the player in videos taken offline.*Ball detection Module* which proposes and implements an improvised object-and-trajectory based algorithm and *2D to 3D conversion Module* which converts the above detected player and ball 2D image coordinates into 3D real world coordinates using calibration and transformation techniques with the help of a single camera.
- Classification Model: The attributes selected in the first part determine the types of features to be extracted from the videos. Computer vision modules detect the fundamental blocks of an event namely: the player and the ball. The features are nothing but varied interpretations of the detected events. These features are then used to classify events based on two classification models:SVM and KNN.

The goal of the paper is to put forward an efficient event detection system based on domain knowledge for feature selection and classification in tennis. This paper also plans to extend these applications and algorithms of tennis analysis systems to coaching industry as it will have a huge impact on raising the standards of learning a sport.

2 Literature Review

Player detection has become a key element in tennis video analysis algorithms. Most player detection algorithms in the field of tennis are focussed on broadcast tennis videos owing to their abundant availability [Zhong and Chang, 2001,

 ISBN 978-0-9934207-2-6

Han et al., 2006, Bertini et al., 2006, Jiang et al., 2009, Archana and Geetha, 2015]. These algorithms are based on adaptive template matching [Miyamori and Iisaku, 2000], spatio temporal information of frame difference , background subtraction [Zhong and Chang, 2001, Bertini et al., 2006, Zivkovic et al., 2003], dominant colour extraction [Han et al., 2006], edge detection [Jiang et al., 2009] etc. Most of the algorithms depend on player specific details such as colour, approximate position of the player, court anatomy etc [Jiang et al., 2009, Archana and Geetha, 2015]. These approaches are not generalised and are also very specific to camera view point. Therefore we have implemented a feature based algorithm for motion detection in order to detect and track a player on court. This paper only deals with singles match i.e., there must be one player on either side of the net, anything else would be considered as noise. Ball detection is very crucial and difficult in the field of tennis analysis. A feature based algorithm was used to detect the player with minimal or no input from the user. In broadcast tennis videos, the size of a tennis ball is very small and has very fast motion making it difficult to detect and track. In videos taken offline depending on the camera parameters, the shape of the ball may appear elongated or deformed due to motion blur. The tennis ball is prone to occlusions and background noises too. Characteristics such as shape, size and colour will be necessary but not sufficient to discriminate the target from other objects. So, the feature based algorithm is insufficient for target detection. Therefore, other additional aspects such as domain knowledge and interpretation of motion consistencies across frames are needed. This problem is thus an object indistinguishable problem [Yu et al., 2003].

We propose an innovative approach towards tennis analysis through improvised detection and tracking algorithms based on formal definition of a shot and its attributes which are described in the section below.

3 Formal Definition and Attribute Selection

Shot is the basic stroke in tennis which can be defined and categorised in many ways. In this section, we formally and extensively define a tennis shot based on its attributes.

A shot 'x' can be represented as a nested tuple:

x = (X,($s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}$), ($c_1, c_2, c_3$))

Where, X is the name of the shot and $s_i \in S_i, where S_i$ is the set of all possible values s_i can have.

- s_1 denotes the type of the shot *(serve or a rally stroke(shot))*
- s_2 denotes area in which the shot was hit by the player *(near the net, mid court or deep)*
- s_3 denotes where the shot was hit with respect to the player *(right, left, shoulder height, overhead or between the legs)*
- s_4 denotes the racket parameters *(side of contact and racket motion)*
- s_5 denotes the number and type of bounce the incoming ball has
- s_6 denotes the trajectory of the ball hit *(low, high or vertical)*
- s_7 denotes where the ball was hit to on the other side *(at the opponent, cross court or down the line)*
- s_8 denotes the pace of the ball hit *(slow or fast)*
- s_9 denotes the spin of the ball hit *(flat, backspin or topspin)*
- s_{10} denotes the grip with which the ball was hit
- s_{11} denotes the roll of the ball
- s_{12} denotes the action with which the ball was hit *(swing and follow through)*
- s_{13} denotes the type of the shot *(attack or defence)*
- s_{14} denotes the position of the opponent on court *(relative)*
- s_{15} denotes the result of the shot
- c_1 denotes the position (coordinates) of player A *(The one who hit the shot)*
- c_2 denotes the position (coordinates) of player B *(The receiver)*.
- c_3 denotes the position (coordinates) of the ball bounce after the shot was hit.

All the above eighteen features completely define a shot. A shot need not require all of these to be well defined i.e., every shot has a certain subset of these attributes which are sufficient to uniquely define it.
For example: a shot 'x' is called a drive volley(X), when:
s_1 = *shot;* s_2 = *mid court;* s_3 = *shoulder height;* s_5 = *zero;* s_8 = *little pace; and* s_{13} = *attack;*
and all the other parameters can hold any value within their domain and would not contribute in defining a drive volley. Most of the shots in tennis have been manually defined with the above parameters and can be found at: `https://goo.gl/uo9Iws`

 ISBN 978-0-9934207-2-6

3.1 Attribute Selection

In this paper, we try to extract these properties of a shot computationally with the help of a video taken from a single camera. Attributes such as s_4, s_8, s_{10}, s_{11} and s_{13} would require some extra input apart from the video to be computed. This can be in the form of manual input or sensors(such as speed sensors or biometric sensors). So, we select the following attributes which can be computed from a tennis video :

- *Spin of the ball (s_9)*: Ball spin is a very innate property of the ball and very crucial in the game of tennis. Spin of the ball determines the trajectory of the ball and affects the bounce of the ball on court. There are 4 kinds of spins that are popularly known: Flat (No spin), Topspin, Backspin and Sidespin, but we mainly focus on the first three.
- *Height of the incoming trajectory (s_6)* This attribute identifies the height of the incoming trajectory of the ball from the net enabling us to differentiate between lobs (and hailmary) and shots hit closer to the net.
- *Position of court bounce*(Part of trajectory of the ball s_9): This attribute differentiates the types of shots based on the position of ball bounce on court. There are 3 major zones parallel to the net on court where the ball can be hit to: Near the net, mid court and baseline. In this paper, we focus on baseline (deep shots) and mid court shots.
- *Number of bounces (s_5)*: This attribute is fundamentally used to classify volley shots (0 bounces) and ground strokes (1 bounce). This is a straightforward classification which just requires the trajectories of the ball.
- *Predicting the next shot(s_6)*: This does not fall into any class of attributes of the shot but uses two main attributes namely, ball bounce c_2 and player position during ball bounce c_3 as inputs in order to predict the next shot likely to played by the player.

3.2 Uses of the attributes

The attributes selected in the above section not only give us the quantitative information such as coordinates of ball bounce or exact trajectory equations of the ball but they open opportunities to a plethora of applications. Focussing on the big picture, which is improving coaching standards with the help of technology, these kinds of attributes help in the analysis of one's game. For instance, shots that fall on the left side of a right handed player often lead to backhand stroke and those on right to forehand. With the help of 'Prediction of the next shot', we can generate a player specific analysis such as the player converting a lot of left side shots into forehand which points at his inability to execute backhand. It can also be used to analyse an opponent's game and generate statistics of his play. Whereas the other four attributes described above can play a very crucial role in the learning process and improving one's own game. In order to extract these attributes from the videos we have designed and implemented three computer vision modules which are explained in the section below.

4 Computer Vision Modules

4.1 Player Detection and tracking

Player detection is a fundamental aspect of any tennis video analysis. The player detection algorithm implemented in this paper starts with the disintegration of the raw input video into a series of frames. Each frame is preprocessed with grayscale conversion and gaussian blur smoothing. The regions of interest are objects with motion in the foreground. These are extracted by eliminating all background noises with background subtraction techniques[Zivkovic, 2004, Zivkovic and Van Der Heijden, 2006] . The resultant mask has all the foreground objects which have movement in them. Next, the image is dilated and mined for all the contours present. The player is the region of interest, which is one of many contours detected. Other contours include moving leaves, stray balls or anything with slight movement. To acquire the target region of interest, we have laid constraints on contour parameters such as size (area of the contour) and position in the frame (closer to ground/court).

Assuming no other person (or equivalently large object) except the player is present(or moving) on the court (as singles matches are considered), the player has the largest contour. A bounding rectangle is drawn around and the position of the player (player coordinates) is defined as the midpoint of the bounding rectangle's base. Taking into consideration, the size of the contour of the player and the angle of the camera(discussed in section 4.1), there will not be any occlusions for the desired region of interest. As the target can be distinguished from other objects based on a feature and not just position in a frame, it is called an object-distinguishable problem [Yu et al., 2003]. If the camera position and parameters are unchanged, then parameters such as contour area range will remain constant, hence there is minimal or no user input required for this method. A drawback in this approach is major shift in pose of the player. Rare positions such as sitting on the court or doing a split after hitting a shot have major impact in the size and shape of the contour. Any normal variations such as bending, ducking, stretching do not affect the player detection.

 ISBN 978-0-9934207-2-6

(a) with stationary ball on court

(b) with similar looking background

Figure 1: Player (green box) and moving ball (blue box) detected.

4.2 Ball Detection and tracking

As ball detection is an object indistinguishable problem (as explained in section 2), additional motion information is required in order to discriminate the target from similar looking objects and backgrounds such as moving leaves of trees, stray balls etc. Initially, the moving object was segmented using motion data, but only limited information could be mined in this process as the video is rich in motion information. A lot of preprocessing was needed in order to stabilise the motion information. Yet false detections such as parts of the player and the background were identified as the ball and in some others, the ball itself was not detected due to its non-ball like appearance because of deformation, object extension, occlusions and non-differentiability from the background. Thus, just object-based methods were not enough to detect the ball accurately. Hence, we propose an improvised object-and-trajectory based algorithm with three components, to overcome the above mentioned challenges.

The first component of our algorithm deals with preprocessing each frame by HSV (Hue, Saturation and Value) filtering to detect the green shade of ball. It is a variable parameter, depending on the video requirements it has to be set interactively. The preprocessing step helps remove a lot of noise, like moving tree branches and slight camera movements at the same time making the ball more prominent. The second component uses optical flow algorithm [Horn and Schunck, 1981] to acquire the trajectories of all moving objects based on motion in successive frames. The third component is contour detection whose parameters are the ball size. These three components together detect the ball by marking a bounding rectangle in every frame. Further in the paper, the usage of the term "Ball coordinates" is generally understood to mean the centre of the above mentioned bounding rectangle.

The accuracy of detecting a ball increases with increasing the shutter speed of the camera. Drawback in this approach is the rare situation of presence of multiple moving balls on the court. The algorithm detects all the balls until they come to rest.

4.3 Conversion of 2D points to 3D

After detecting the points of interest (the ball and the player), one of the major issue that arised is that these points are on the image plane i.e. 2D. For accurate trajectory equations and classifications, we would require 3D points in the real world, as boundaries(planes) between classes are more well-defined and accurately represented in 3D space.

This section focusses on the conversion of 2D image coordinates into 3D real world coordinates. This process generally is performed with stereo vision or multiple camera

We have used a single calibrated camera and geometric transformations [Siswantoro et al., 2013] to obtain the 3D coordinates. The algorithm we used given the image point p(x_i,y_i) to determine the corresponding world point P(x_w,y_w) is[Siswantoro et al., 2013]:

1. Calibration of the camera to extract Extrinsic and Intrinsic properties of the camera [Tsai, 1987]. Extrinsic properties specify the World to camera coordinate transformation and consist of Rotation matrix(R) and Translation Matrix(T) where as Intrinsic properties specify camera to Image coordinate transformation and consist of focal lengths(f_x,f_y) where $f_x = fs_x$ and $f_y = fs_y$, aspect ratio(a) and the point where the optical axis intersects the image plane(c_x,c_y).
2. The point p in the image plane is obtained from sections 3.1 and 3.2, which is given as input to the following set of mathematical transformations.
3. Image to camera coordinate conversionp_c (x_c,y_c,z_c): [Siswantoro et al., 2013]

$$x_c = \frac{x_i - c_x}{s_x}$$

 ISBN 978-0-9934207-2-6

$$y_c = \frac{y_i - c_y}{s_y}$$

$$y_c = f$$

4. Camera to world coordinate conversion: [Siswantoro et al., 2013]

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = R^{-1} \left(\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} - T \right) \tag{1}$$

Although, this approach is proved and validated in [Siswantoro et al., 2013], we have re-validated it by measuring the distances between known points on court and crosschecked it with the real values. This algorithm is iterated over all the points detected and now we have a list of 3D world coordinates of the player and ball (and its trajectory).

5 Data Set

5.1 Videos

The aim of the paper is to introduce technology into everyday coaching, hence we have to work on videos taken specifically for this purpose. We have collected 12 hours of video footage of tennis being played at an academy which contains a lot of background noise along with trees and other people moving in the background mimic the real world scenario. These videos are taken with the camera being placed at an angle of 45 degrees to the net facing one side of the court. To create the data set we have used Canon EOS 60D camera to record videos. A frame rate of 50 fps and a shutter speed of 1/1600 s was set in order to capture the fast moving tennis ball intact. The ISO measure was set to 2000. Due to high shutter speed, increasing ISO will help make the image less darker but also reduce the image quality. As for the techniques implemented in this paper, the resolution is not of high priority and thus can be compromised.

5.2 Feature Extraction

5.2.1 Features

In order to extract selected shot attributes, we have defined main features that are used repeatedly across the attributes however the interpretation and evaluation of these features is unique to an attribute. Among all the features, Trajectory Estimation, Ball bounce coordinates and Player coordinates are the main features to be extracted from the video dataset. Player coordinates can be extracted straightforward as discussed above whereas trajectory and ball bounce are interlinked and require a little more computation.

Every video is preprocessed to generate “events”. Event is a single shot trajectory captured in a sequence of frames. Every event is comprised of multiple sub events. Sub events are the uninterrupted trajectories of the ball. Ball bounce, striking the racket or making contact with any other object are considered as interruptions. Computationally, the abrupt change in the trajectory of the ball in either directions triggers an interruption. After the ball has been tracked, as described in Section 3.2, the ball coordinates from each frame are stored in a list. Although, after searching through a window of frames, in the unlikely event of the ball not being detected, the list is flushed and a new sub event starts.

The above collected data is now fed into a curve fitting algorithm. Figures 2(a) and 2(b) illustrate the trajectories of such subevents where the X axis denotes the X coordinates of the image and Y axis denotes the Y coordinates of the image. The output of the curve fitting algorithm is a curve of the order 2, hence the curve is considered quadratic. It is clearly depicted by the parabolic trajectories shown in Figures 2(a) and 2(b). Hence we have a quadratic equation for every subevent, which depicts the trajectory of the ball.

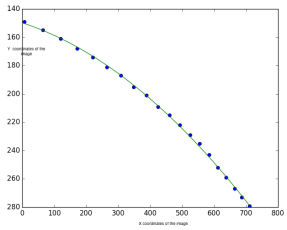
Thus the trajectory of the ball (parabolic equation along with all detected points) is computed. Ball bounce is defined as the first interruption of an event of an incoming ball.

5.2.2 Annotation

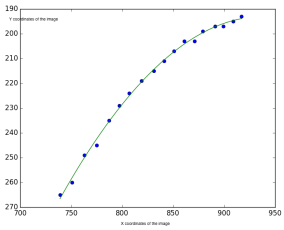
The initial raw data set is a collection of 12 hour long videos taken in accordance with the standards explained in Section 4.1. We have taken two sets of videos: Training and Testing. These videos are to train and test the classification model explained in section 5. In the training video set, every attribute has a unique video which contains shots portraying that particular attribute. For instance, in the case of spin determination, the training set has a video which contains only topspin shot and another video which contains only backspin shots. Whereas the testing videos contain a complete match which has all kinds of shots. These videos were manually annotated.

The preprocessing step for annotation has been programmed to pause at the end of every event in order to take input from the user. This input includes the decision of considering that particular event to annotate or not along with the

 ISBN 978-0-9934207-2-6



(a) Trajectory of ball before bounce



(b) Trajectory of ball after bounce

Figure 2: Points of ball trajectory tracked. X axis and Y axis show the pixel (X and Y) coordinates of the ball detected in the video respectively.

suitable label. Thus computed data along with the annotated label is written into a text file. Every video after annotation generates five different text files which contain mined information. They are:

1. Coefficients of trajectories before and after bounce
2. Ball bounce coordinates
3. Player coordinates
4. Coordinates of point of contact of ball and racket
5. Coordinates of ball during the complete trajectory

These generated text files are then fed into the classification module as training data set which is discussed in detail in the section below.

6 Methodology and Results

The initial raw input for this classification model are videos. Searching through videos and processing them is a very tedious and difficult process. Hence to reduce the search space, the videos are preprocessed and annotated into text files which are used as the direct input to the classification models. We have used supervised learning technique for all the attributes. Owing to the variance in number of classes and dimensionality in features, a single classifier was not able to accommodate for all the attributes, hence two models SVM and KNN have been tried and they yielded results based on parameters such as the number of classes and separability. For SVM, we have considered the parameter C to be 1 and the kernel choice depended on linear separability of the data. We have considered Linear and polynomial kernels as per requirement in this process and for KNN we have taken k = 5 as these parameters yielded better results for same data. We have used a 10-fold cross validation method for both the models.

The results obtained from the classification model described in section 5 for all the attributes is illustrated in Fig 3. Based on the linear separability criteria and number of classes, both methods SVM and KNN yielded different results. A detailed discussion on attribute specific observations are below.

6.1 Spin

Spin of the ball generally affects the trajectory and the bounce. In this paper, we reverse map the trajectories to the spin associated. In a flat shot, the ball is played with no spin and the angle with which the ball strikes the ground(angle of incidence) is equal to the angle with which it leaves the ground(angle of reflection). However after striking the court, the ball acquires a topspin. In a topspin shot, the angle of incidence is greater than the angle of reflection and has a higher bounce as it is hit high over the net [Brody et al., 2002] Whereas in a backspin shot, the angle of incidence is lesser than angle of reflection.

This information is used compare the trajectories of all three spins. We compare the before bounce and after

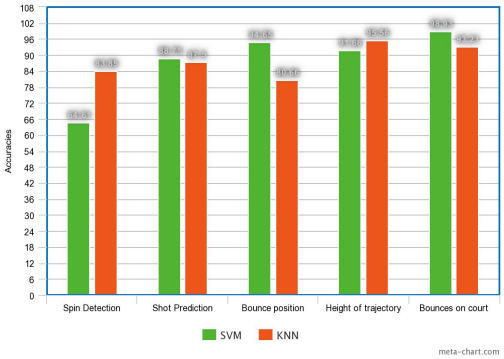


Figure 3: Average Accuracy Results for attributes

 ISBN 978-0-9934207-2-6

bounce coefficients of one spin to another. As this is not a linearly separable problem, ,we have used a polynomial kernel for SVM. As this is a three class problem KNN performed better than SVM. Accuracy for KNN is 83.85%and accuracy for SVM is 64.61%

6.2 Trajectory height

This attribute is mainly to distinguish a lob shot from other non-lob shots. A lob shot has very high trajectory before hitting the ground. Thus it is relatively easily distinguishable from normal close to net shots. SVM with RBF kernel and KNN have given similar average accuracies of 91.66% and 95.56% respectively where as when used with SVM linear kernel an average accuracy of 88% was shown.

6.3 Position of ball bounce

Here, we classify shots based on where they are hit to, on the court. As discussed in section 2.1, we consider deep shots and mid court shots in this paper. Deep shots are those which are close to the baseline whereas mid court shots are those which are hit neat the T. This classification is linearly separable and thus an SVM with linear kernel was used along with a KNN with k as 5. With linear separability and binary classification, SVM yields better results than a KNN. Average accuracy of SVM is 94.65% and average accuracy of KNN is 80.66%.

6.4 Number of bounces

This is by far the most easy and highly accurate classification done by the model. In this section, we classify shots which are hit directly without a bounce(volley) and the shots that are hit after one bounce(ground stroke). We consider the first interruption of the incoming ball which is on the racket for volleys and the bounce on court for ground strokes. These two happen on different planes which are considerably far from each other. Hence a clear boundary can be established. Both SVM and KNN yield high accuracies of 98.93% and 93.21% respectively.

6.5 Shot prediction

n the game of tennis, estimating the strengths and weaknesses of the opponent plays a key role. This classification helps estimate the game of a player by predicting the next shot a player is likely to hit or miss. This classification helps in quantifying the frequency of each shot played and a player's success rate of attempting it. Shot prediction has two class labels: Forehand(+1) and Backhand (-1). SVM and KNN gave similar average accuracies for this classification: 88.75% and 87.5% respectively.

These results clearly who the competency of the methods and techniques presented in this paper.

7 Conclusion and Future work

This paper is a preliminary attempt to make the learning process of tennis more effective with the help of technology. We use domain knowledge in order to select attributes and their features of a shot. Computationally two main features were extracted from tennis videos: A. Player detection by a generalised feature based motion tracking algorithm, and B. Ball tracking by an improvised object-and-trajectory based algorithm. Every shot played is defined as a 18-feature tuple. Selected features from the definition were computed by extracting ball trajectory and player position in real world situations. The features thus extracted are then classified and validated using SVM and KNN techniques. The features extracted along with their accuracies are: spin of the ball (83.85%), trajectory height (95.56%), position (94.65%) and number of bounces (98.93%). Additional application of shot prediction (88.75%), which predicts the next likely shot to be hit based on the player and ball bounce positions, was also implemented.

This idea can be extended to implement plethora of applications which can be used to draw analyses and statistics of a player's game. This research can also be extended to use other modes of input like biometric sensors in order to extract the remaining attributes. An immediate extension would be to replicate the system on the other side of the court and by aligning both the cameras in order to capture the entire match and also to extend this kind of domain knowledge based methods to analyse other sports.

References

[Archana and Geetha, 2015] Archana, M. and Geetha, M. K. (2015). An efficient ball and player detection in broadcast tennis video. *Intelligent Systems Technologies and Applications*, 1:427.

[Bertini et al., 2006] Bertini, M., Cucchiara, R., Bimbo, A., and Prati, A. (2006). Semantic adaptation of sport videos with user-centred performance analysis. *IEEE Transactions on Multimedia*, 8(3):433–443.

[Brody et al., 2002] Brody, H., Cross, R., and Lindsey, C. (2002). *The physics and technology of tennis*. Usrsa.

[Chang et al., 2012] Chang, C.-K., Fang, M.-Y., Kuo, C.-M., and Yang, N.-C. (2012). Event detection for broadcast tennis videos based on trajectory analysis. In *Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on*, pages 1800–1803. IEEE.

[Han et al., 2006] Han, J., Farin, D., et al. (2006). Multilevel analysis of sports video sequences. In *Electronic Imaging 2006*, pages 607303–607303. International Society for Optics and Photonics.

[Horn and Schunck, 1981] Horn, B. K. and Schunck, B. G. (1981). Determining optical flow. *Artificial intelligence*, 17(1-3):185–203.

[Jiang et al., 2009] Jiang, Y.-C., Lai, K.-T., Hsieh, C.-H., and Lai, M.-F. (2009). Player detection and tracking in broadcast tennis video. In *Pacific-Rim Symposium on Image and Video Technology*, pages 759–770. Springer.

[M. Archana, 2016] M. Archana, M. K. G. (2016). Automatic event detection and classification based on ball trajectory in broadcast tennis video using svm and hmm. *International Journal of Innovation and Scientific Research*, 23(2):233–242.

[Miyamori and Iisaku, 2000] Miyamori, H. and Iisaku, S.-I. (2000). Video annotation for content-based retrieval using human behavior analysis and domain knowledge. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 320–325. IEEE.

[Pingali et al., 1998] Pingali, G. S., Jean, Y., and Carlbom, I. (1998). Real time tracking for enhanced tennis broadcasts. In *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*, pages 260–265. IEEE.

[Siswantoro et al., 2013] Siswantoro, J., Prabuwono, A. S., and Abdullah, A. (2013). Real world coordinate from image coordinate using single calibrated camera based on analytic geometry. In *Soft ComputIng Applications and Intelligent Systems*, pages 1–11. Springer.

[Tsai, 1987] Tsai, R. (1987). A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344.

[Xu et al., 2003] Xu, G., Ma, Y.-F., Zhang, H.-J., and Yang, S. (2003). A hmm based semantic analysis framework for sports game event detection. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 1, pages I–25. IEEE.

[Yan et al., 2014] Yan, F., Christmas, W., and Kittler, J. (2014). Ball tracking for tennis video annotation. In *Computer Vision in Sports*, pages 25–45. Springer.

[Yu et al., 2007] Yu, X., Jiang, N., and Ang, E. L. (2007). Trajectory-based ball detection and tracking with aid of homography in broadcast tennis video. In *Electronic Imaging 2007*, pages 650809–650809. International Society for Optics and Photonics.

[Yu et al., 2004] Yu, X., Sim, C.-H., Wang, J. R., and Cheong, L. F. (2004). A trajectory-based ball detection and tracking algorithm in broadcast tennis video. In *Image Processing, 2004. ICIP'04. 2004 International Conference on*, volume 2, pages 1049–1052. IEEE.

[Yu et al., 2003] Yu, X., Xu, C., Leong, H. W., Tian, Q., Tang, Q., and Wan, K. W. (2003). Trajectory-based ball detection and tracking with applications to semantic analysis of broadcast soccer video. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 11–20. ACM.

[Zhong and Chang, 2001] Zhong, D. and Chang, S.-F. (2001). Long-term moving object segmentation and tracking using spatio-temporal consistency. In *Image Processing, 2001. Proceedings. 2001 International Conference on*, volume 2, pages 57–60. IEEE.

[Zivkovic, 2004] Zivkovic, Z. (2004). Improved adaptive gaussian mixture model for background subtraction. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 28–31. IEEE.

[Zivkovic et al., 2003] Zivkovic, Z., Petkovic, M., Van Mierlo, R., van Keulen, M., van der Heijden, F., Jonker, W., and Rijnierse, E. (2003). Two video analysis applications using foreground/background segmentation. *IEEE*.

[Zivkovic and Van Der Heijden, 2006] Zivkovic, Z. and Van Der Heijden, F. (2006). Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern recognition letters*, 27(7):773–780.

Saliency Detection and Object Classification

Christopher Cooley; Sonya Coleman; Bryan Gardiner; Bryan Scotney

Ulster University

Abstract

Humans have a distinct ability to process only the information that is of interest within a scene, however, this is not an easy task for computers. Trying to replicate this behaviour, many methods have been proposed to generate saliency maps that segment the object of interest within an image. In this paper, we investigate the problem of object classification, and whether saliency detection can be used. We generate saliency maps produced by two different currently published saliency detection methods, and train separate linear SVMs using the feature vectors obtained from these methods. We evaluate these methods against the traditional approach of extracting features from an image for object classification, namely HoG features. Our results show that saliency detection can be used for object classification, and improves accuracy by 5%.

Keywords: Image Processing, Saliency Detection, Classification

1 Introduction

Humans have the ability to look at a visual environment and focus only on the most interesting regions. This is despite the amount of visual stimuli that meets the eye every second (approximately 10^8 – 10^9 bits of visual information) [Borji and Itti, 2013], which is too much to process in real-time. Therefore, the human visual system picks out interesting areas by means of optimisation. Motivated by this selective processing mechanism salient object detection, aiming to detect and segment the most attractive objects from the background of an image, has attracted a lot of interest from research communities including neuroscience, psychology, robotics and computer vision. Over the past decade a number of methods have been proposed, incorporating different feature cues such as colour [Ma and Zhang, 2003], edges [Yang et. al., 2017], contrast [Niu et. al., 2016], texture [Zhang et. al., 2017] and a combination of features [Cheng et. al, 2014].

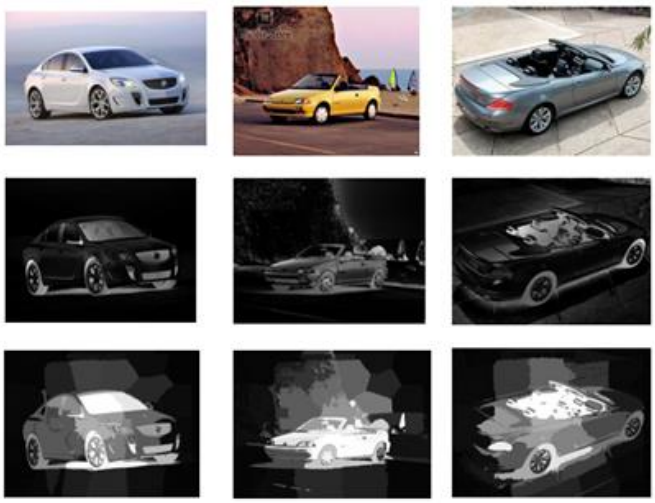


Figure 1: Positive observations. Top row: input images. Middle row: SRDAS method saliency maps. Bottom row: DES method saliency maps.

Salient object detection is often described as an important pre-processing step for other tasks such as classification, retrieval and object co-segmentation. Robotics and computer vision communities want to replicate this behaviour due to it being more computationally efficient, decreasing the regions within an image to be processed [Chen et. al., 2002]. Salient object detection remains a challenging task due to the acquisition of prior knowledge and the difficulty of understanding and replicating the complex visual attention mechanism in humans [Ren et. al., 2014]. This paper studies one area of visual attention – salient object detection. The specific topic being investigated is whether salient detection can truly improve object recognition. This involves completing a comparative study consisting of the classification of objects fusing saliency maps generated from RGB images

 ISBN 978-0-9934207-2-6

using two different existing approaches [Cheng et. al., 2014, Achanta et. al., 2008].

Histogram of Oriented Gradients, introduced by [Dalal and Triggs, 2005] was originally applied to the problem of detecting humans with the help of an SVM classifier, but has expanded into other areas such as face recognition [Zhu and Ramanan, 2012], scene classification [Xiao et. al., 2010] and object detection [Zhang et. al., 2007]. The essential thought behind HoG descriptors is that local object appearance and shape can be described by the distribution of intensity gradients or edge directions.

The contributions of this paper are twofold: 1) we investigate whether saliency detection can be used for the classification of objects, 2) if so, can it improve the object classification accuracy when compared with other common feature extraction approaches. The remainder of the paper is organised as follows: in Section 2, we present a brief overview of current prominent saliency detection methods. Section 3 outlines the structure of work undertaken and the methods used. We provide details of the machine learning algorithm used in Section 4 with Sections 5 and 6 containing the performance evaluation and conclusion respectively.

2 State of the Art in Saliency Detection Methods

Saliency detection methods can be divided into two categories, top-down and bottom-up [Itti et. al., 1998]. Top-down approaches are task-dependent and based on preceding information of a scene or object, whereas bottom-up methods are focused on detecting regions or points that attract people's attention, driven by low-level stimuli within a scene, such as colour, orientation and contrast. Originally, computing saliency meant predicting the areas of an image that people would look at, but has expanded to object level saliency detection. The three main stages of saliency detection include the extraction of meaningful features, activation, which shows the conspicuous areas within an image and lastly, normalisation/combination in which each feature channel is amalgamated into a master saliency map. In the past number of decades, many techniques have been proposed for detecting salient objects within an image. One of the earliest models was proposed by [Itti et. al., 1998], in which they generate a saliency map based on the variance between fine and coarse scales in intensity, orientation and colour feature maps. Their final saliency map is the result of a winner-takes-all (WTA) competition employed to select the most conspicuous image locations. This method has become influential in saliency detection and has inspired numerous methods. [Ma and Zhang, 2003] argue that, although successful traditional techniques considered three image properties, colour, texture and shape, these are not based on human understanding of images and therefore proposed a saliency model constructed on local contrast analysis. Salient areas within the saliency map were enhanced using a fuzzy growing method. [Frintrop et. al., 2007] based their method upon [Itti et. al, 1998], in which their method uses a combination of top-down and bottom-up cues. This method differs in that they use integral image for computational efficiency, when creating their real-time visual attention system.

The use of edges and edge strengths for detecting salient regions are explored by [Yang et. al., 2016]. [Liu et. al., 2016] exploit background priors to produce saliency maps. The algorithm is based on the combination of background seeds, a centrosymmetric Gaussian function and the smoothness prior constraints. An approach for estimating saliency is proposed by [Hu et. al., 2004] which applies heuristic measures on initial saliency measures obtained by histogram thresholding of feature maps consisting of colour, intensity and orientation. [Achanta et. al., 2009] propose using Difference of Gaussian (DoG) filters to eliminate redundant information, and to output full resolution saliency maps with the boundaries of salient object being well defined.

Within this paper we use two saliency detection methods, Depth Enhanced Saliency Method (DES) [Cheng et. al., 2014] which incorporates the use of colour, depth and spatial bias cues, and Salient Region Detection and Segmentation (SRDAS) [Achanta et. al., 2008] in which saliency is computed using image sub-regions and compared with each neighbouring area.

 ISBN 978-0-9934207-2-6

3 Feature Extraction Methods

The analysis involves the use of three different feature extraction approaches, two of which compute saliency maps using different approaches. The saliency feature vectors gained from each method is used for the classification task, and compared with the use of HoG features for classification. First, from the RGB input image, HoG features [Dalal and Triggs, 2005] are computed. HoG features are commonly used for object recognition and classification. After this, the same image is put through two different salient object detection algorithms to calculate saliency maps using colour and contrast cues. The three approaches used are described below. We then train a two-class linear support vector machine with the HoG features and saliency maps respectively to answer the research questions addressed in section one. To the best of the authors' knowledge, there are no other approaches using saliency maps for classifying objects, via the use of a support vector machine.

3.1 HoG Features

Histogram of oriented gradients (HoG) is a dense feature descriptor [Dalal and Triggs, 2005], which extracts features from all locations (or a region of interest) in an image. This approach counts the occurrences of directions of gradients in a dense grid of uniformly spaced cells. It does so by dividing an image into small (typically 8x8 pixels) cells and blocks of 4x4 cells. A brief outline of this process can be seen in Fig. 2. This technique is alike that of edge detectors, SIFT descriptors and shape contexts, nevertheless, HoG is computed on a dense grid of uniformly spaced cells. For improved accuracy, HoG uses overlapping local contrast normalisation. We implement the HoG feature extraction based on the findings of [Dalal and Triggs, 2005], which yielded the best performance with linear gradient voting into 9 orientation bins in 0° - 180°. Our approach differs in that a cell is made up of 8x8 pixels whereas theirs was set at 16x16, as some of the finer feature detail is lost in larger cellblocks. We decided that cells of 8x8 pixels would provide enough large-scale spatial information without compromising on the small-scale details.

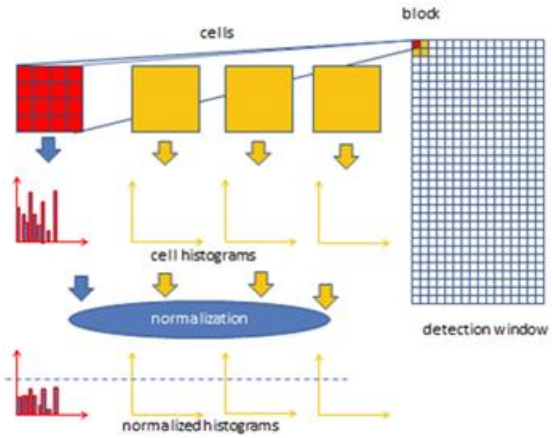


Figure 2: Overview of Histogram of Oriented Gradients [Intel Developer Zone, 2017]

3.2 DES Method

Originally, the Depth Enhanced Saliency Method (DES) [Cheng et. al., 2014] incorporated three feature cues for computing saliency namely, colour contrast, depth contrast and spatial bias as it was designed for use with RGB-D images. Humans have a tendency to look at the centre of an image, this is referred to as 'centre-bias' in 2D image data; spatial bias is the extension of this when using 3D information. However, as we currently only investigate object classification using RGB images; we have simplified this method to use the colour contrast cue only, due to spatial bias's dependence on depth data.

This method starts by grouping similar pixels into K clusters based on colour using the K-means algorithm. Each region is defined: $R = (r_k)_{k=1}^{K}$, with each region being represented as a 5-dimensional vector consisting of RGB colour and image coordinates. The final saliency map extracts the object within the scene whose colour contrasts with the background within the image. The colour contrast cue (CC) of each region r_k is calculated as:

$$F_{cc}(r_k, r_i) = ||c_k - c_i||_2 \quad (1)$$

 ISBN 978-0-9934207-2-6

where $F_{cc}(r_k, r_i)$ refers to the Euclidean distance in colour space and c_i is the colour centre of region r_i. Colour contrast is a useful feature cue when the object of interest is clearly distinguishable from the background; this however, degrades somewhat when the background colour is close to that of the object [Cheng et. al., 2014]. An example of this can be seen from the generated saliency maps. Some examples of the final saliency maps from this method can be seen in the bottom row of Fig. 1.

3.3 Saliency Region Detection and Segmentation Method

Within the Saliency Region Detection and Segmentation Method (SRDAS), saliency is defined as the local contrast of an image region with respect to its neighbourhood at various scales [Achanta et. al., 2008]. An average feature vector comprised of pixels from image sub regions is evaluated against the average feature vector of pixels from its neighbourhood. Saliency maps are computed at different scales, combined pixel wise and normalized to obtain the final saliency map. At a particular scale, the saliency value $c_{i,j}$ for pixel (i, j) within an image is determined by the distance D between the average vectors of pixel features of the inner region R_i and the outer region R_2 as:

$$c_{i,j} = D\left[\frac{1}{N_1}\left(\sum_{p=1}^{N_1} v_p\right), \left(\frac{1}{N_2}\sum_{q=1}^{N_2} v_1\right)\right] \quad (2)$$

N_1 and N_2 are the number of pixels in the respective regions of R_1 and R_2, and v is pixel feature vector. If the feature elements in v are uncorrelated then distance D is Euclidean, otherwise D is a Mahalanobis distance. To calculate the Euclidean distance RGB image values are translated into CIELab colour space. The final saliency map is obtained by adding the saliency maps pixel-wise. The average saliency value of each image segment is then compared against a threshold, with the values exceeding this chosen as salient. This method produces a saliency map of the same resolution as the input image and achieves better segmentation of the salient object as can be seen in the middle row of Fig. 1.

4 Saliency and Machine Learning

4.1 Support Vector Machine

The research question we are aiming to answer is composed of a classification problem. Classification refers to the allocating of an observation x_i to a qualitative class $y_i \in \{1, 2, \ldots N\}$ Support vector machine (SVM) is a state-of-the-art supervised machine learning technique introduced by Vladimir Vapnik [Cortes and Vapnik, 1995], based on statistical learning theory. The idea behind the SVM is to map input vectors into high dimensional feature space.

We have a dataset $D = \{(x_i, y_i)\}_{i=1}^{e}$ of labelled examples, where each observation $y_i \in \{-1, 1\}$, therefore we use a linear kernel to discriminate between the data. We trained each support vector machine (RGB-HoG and the two respective saliency methods) using 1,552 samples (broken into 776 positive and 776 negative samples) and tested the classifier using a sample of 1,360 images (consisting of 680 positive and 680 negative). Fig. 3 gives an overview of the training and testing process using saliency maps.

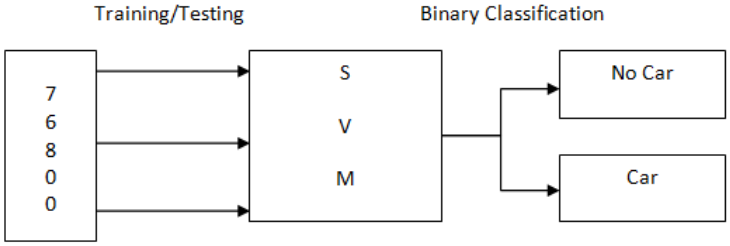


Figure 3: Overview of Saliency Maps and SVM

 ISBN 978-0-9934207-2-6

5 Performance Evaluation

To answer the research questions posed in this paper, we carried out two different experiments: Classifying objects by means of extracting HoG features from RGB images. We also put the same images through two different saliency detection methods to produce saliency maps. After this, we trained and tested three separate SVM classifiers using the respective HoG features and saliency maps. Experiments were run in Matlab and all images were resized (to 240 x 320 pixels), scaled using bicubic interpolation, due to the varying sizes of images within the selected datasets.

There are two datasets used in this work RGB-D Object Dataset [Lai et. al., 2011] of which we only use the colour images and Cars dataset [Krause et. al., 2013]. The positive class of the SVM refers to an image with a car, whereas the negative class is no car. For the negative sample images, we use a number of object categories from the RGB-D dataset, including apple, ball, banana, calculator and cap. The positive samples are taken from the cars dataset, which includes 196 different types of cars. The orientation, colour of the car and the background scene varies throughout the dataset. This increases the difficulty of detecting the salient object within a scene. Our total dataset is made of 2,912 images; we manually labelled all samples as ground truth before carrying out training or testing on the SVM. Each method was executed with the identical images and sample sizes to draw a fair comparison.

At first we ran each method on a subsample of the dataset, training the SVM on 18 negative classes and 18 positive classes. We then tested the classifier on the same sample size, which consisted of different image data. The results of the preliminary experiment were positive, as can be seen in Table 1, in that it showed that saliency maps can be used for object classification,

5.1 RGB-HoG

From an RGB input image, we computed the HoG features which produced a feature vector to be used for training the SVM. We trained the linear SVM using a sample of data that included 1552 observations. The total number of observations was split into 50% positive and 50% negative. The trained SVM was tested on 1360 observations, again made-up of 50% positive and negative samples that were manually labelled for ground truth. The trained SVM accurately classified 89.9% of the dataset. This was the benchmark against which, the saliency detection methods would be judged.

5.2 Saliency Detection

Firstly, we computed the saliency map from the DES method. This resulted in 1552 saliency maps being generated, which would then be used for training the support vector machine. This method was not able to segment the complete conspicuous object in some of the images within the car dataset, due to the orientation and colour of the car. The DES trained SVM, in terms of accuracy performed worse than RGB-HoG achieving a percentage of 88.6%.

After this we performed the same process, using the SRDAS method. Training consisted of 776 images with cars and 776 without cars. Testing was carried out using 680 positive and 680 negative samples. The saliency maps produced by this method were better quality, keeping the same resolution as the input image. Orientation did not cause this method the same trouble in fully segmenting the salient object. SRDAS trained support vector machine accurately classified 94.9% of the test data.

5.3 Accuracy

Accuracy is the evaluation metric used to compare the success of each method used for classification within this paper. Accuracy A can be defined as:

$$A = \left(\frac{cc_n}{tc_n}\right) * 100 \quad (3)$$

 ISBN 978-0-9934207-2-6

Where cc_n and tc_n are the number of correctly predicted classes and total number of testing samples respectively. Each method was tested with a subsample of data before being run on the full dataset. From the results in Table 1, it can be observed that the DES method had a misclassification rate of 3% on the preliminary experiment, whereas the HoG features and SRDAS methods accurately classified all the data. This experiment answered one of the original questions posed in that it proved saliency maps are suitable for classification tasks.

Next, we implemented these methods with the full dataset, evenly splitting the train and test data into 50% positive and 50% negative classes. Firstly, we trained the SVM with HoG features, which in turn correctly classified 89.9% of the dataset. Using the DES method did not improve object classification, however, trained with the saliency maps computed using the SRDAS method, accuracy recorded a 5% improvement.

	HoG Features			DES Saliency Map			SRDAS Saliency Map		
	Train	Test	Accuracy	Train	Test	Accuracy	Train	Test	Accuracy
Preliminary	36	36	100%	36	36	97%	36	36	100%
Full dataset	1552	1360	89.90%	1552	1360	88.60%	1552	1360	94.90%

Table 1: Accuracy results from experiments

6 Conclusion

In this paper, we have shown that it is possible to use saliency maps as a means for classifying objects. We carried out experiments involving creating saliency maps and using these for classification instead of HoG features. Results show that saliency detection using the SRDAS method improves object recognition by 5% in comparison to the traditional approach of extracting HoG features from the input image and using these for training the SVM, whereas accuracy decreases by 1.3% when the DES saliency map is used for training. The reason being that SRDAS produces a higher quality saliency map for our dataset. The DES method only partially segments some of the salient objects within images due to their orientation, and colour not being very distinctive from that of the background.

7 Future Work

The next step in this research is to consider saliency detection in videos, incorporating the use of depth data which opens up a new problem domain, with multiple frames to consider alongside objects coming into and leaving the field of view. We will also be extending the number of classes considered for classification within this work. Features from a deep network will also be considered. Ways of improving upon current performance will be an issue we will be addressing moving forward.

8 References

[Achanta et. al., 2008] Achanta, R., Estrada, F., Wils, P., & Sabine, S. (2008). Salient Region Detection and Segmentation. *International Conference on Computer Vision Systems, ICVS, 2008, 5008*, 66–75.

[Borji and Itti, 2013] Borji, A., & Itti, L. (2013). State-of-the-Art in Visual Attention Modeling, *35*(1), 185–207.

[Chen et. al., 2002] Chen, L., Fan, X., & Zhang, H. (2002). A visual attention model for adapting images on small displays A visual attention model for adapting images on small displays. In *MSR-TR-2002-125 Microsoft Research, Redmond, WA (2002)* (pp. 1–21).

[Cheng et. al., 2014] Cheng, Y., Fu, H., Wei, X., Xiao, J., & Cao, X. (2014). Depth Enhanced Saliency Detection Method. In *Proceedings of International Conference on Internet Multimedia Computing and Service*.

[Cortes and Vapnik, 1995] Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, *20*(3), 273–297.

[Dalal and Triggs, 2005] Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005* (Vol. I, pp. 886–893). https://doi.org/10.1109/CVPR.2005.177

[Frintrop et. al., 2007] Frintrop, S., Klodt, M., & Rome, E. (2007). A Real-time Visual Attention System Using Integral Images. In *International Conference on Computer Vision Systems*.

[Hu et. al., 2004] Hu, Y., Xie, X., Ma, W., Chia, L., & Rajan, D. (2004). Salient Region Detection using Weighted Feature Maps based on the Human Visual Attention Model. In *Pacific Rim Conference on Multimedia*.

[Intel Developer Zone, 2017] Intel Developer Zone. (2017). Histogram of Oriented Gradients (HOG) Descriptor. In https://software.intel.com/en-us/node/529070. Accessed 22/05/2017.

[Itti et. al., 1998] Itti, L., Koch, C., & Niebur, E. (1998). A Model of Saliency-Based Visual Attention for Rapid Scene Analysis. In *IEEE Trans. on Pattern Analysis and Machine Intelligence* (Vol. 20, pp. 1254–1259).

[Krause et. al., 2013] Krause, J., Stark, M., & Li Fei-Fei, J. D. (2013). 3D Object Representations for Fine-Grained Categorization. In *4th IEEE Workshop on 3D Representation and Recognition, at ICCV 2013*.

[Lai et. al., 2011] Lai, K., Bo, L., Ren, X., & Fox, D. (2011). A Large-Scale Hierarchical Multi-View RGB-D Object Dataset. In *EEE International Conference on Robotics and Automation, 2011* (pp. 1817–1824).

[Liu et. al., 2016] Liu, Z., Gu, G., Chen, C., Cui, D., & Lin, C. (2016). Background Priors based Saliency Object Detection. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA)*.

[Ma and Zhang, 2003] Ma, Y., & Zhang, H. (2003). Contrast-based Image Attention Analysis by Using Fuzzy Growing. In *ACM International conference on Multimedia* (pp. 374–381).

[Niu et. al., 2016] Niu, J., Bu, X., & Qian, K. (2016). Exploiting contrast cues for salient region detection. In *Multimedia Tools and Applications (2016)* (pp. 1–15). Multimedia Tools and Applications. https://doi.org/10.1007/s11042-016-3430-2

[Ren et. al., 2014] Ren, Z., Gao, S., Chia, L., & Tsang, I. W. (2014). Region-Based Saliency Detection and Its Application in Object Recognition, *24*(5), 769–779.

[Xiao et. al., 2010] Xiao, J., Hays, J., Ehinger, K. A., & Torralba, A. (2010). SUN Database : Large-scale Scene Recognition from Abbey to Zoo. In *Computer Vision and Pattern Recognition (CVPR)* (pp. 3485–3492).

[Yang et. al., 2017] Yang, B., Zhang, X., Chen, L., Yang, H., & Gao, Z. (2017). Edge guided salient object detection. *Neurocomputing*, *221*(August 2016), 60–71. https://doi.org/10.1016/j.neucom.2016.09.062

[Zhang et. al., 2017] Zhang, Q., Lin, J., Tao, Y., Li, W., & Shi, Y. (2017). Neurocomputing Salient object detection via color and texture cues. *Neurocomputing*, *243*, 35–48. https://doi.org/10.1016/j.neucom.2017.02.064

[Zhang et. al., 2007] Zhang, W., Zelinsky, G., & Samaras, D. (2007). Real-time Accurate Object Detection using Multiple Resolutions. In *Proceedings of the IEEE International Conference on Computer Vision*.

[Zhu and Ramanan, 2012] Zhu, X., & Ramanan, D. (2012). Face Detection, Pose Estimation, and Landmark Localization in the Wild. In *Computer Vision and Pattern Recognition (CVPR)* (pp. 2879– 2886).

IDT Vs L2 Distance for Point Set Registration

H. Alghamdi, M. Grogan and R. Dahyot

School of Computer Science and Statistics
Trinity College Dublin
Ireland

Abstract

Registration techniques have many applications such as 3D scans alignment, panoramic image mosaic creation or shape matching. This paper focuses on (2D) point cloud registration using novel iterative algorithms that are inspired by the Iterative Distribution Transfer (IDT) algorithm originally proposed to solve colour transfer [Pitié et al., 2005, Pitié et al., 2007]. We propose three variants to IDT algorithm that we compare with the standard L2 shape registration technique [Jian and Vemuri, 2011]. We show that our IDT algorithms perform well against L2 for finding correspondences between model and target shapes.

Keywords: Registration, IDT, L2 distance.

1 Introduction

We use the following notations: the datasets $\{\mathbf{u}_i\}_{i=1,\cdots,m}$ (model *moving* point cloud) and $\{\mathbf{v}_j\}_{j=1,\cdots,n}$ (target point cloud), are point clouds in $\mathbb{R}^d$. $\mathbf{e}$ is a unit vector in $\mathbb{R}^d$ to project samples on a 1D space e.g. $u_i^e = \mathbf{e}^T\mathbf{u}_i$ and $v_j^e = \mathbf{e}^T\mathbf{v}_j$ are 1D scalar values. We investigate the registration of point clouds in $\mathbb{R}^d$ using an iterative approach where at each step the problem solved is the registration of 1D datasets $\{u_i^e\}_{i=1,\cdots,m}$ on $\{v_j^e\}_{j=1,\cdots,n}$. The intuitive idea to this strategy is that if the two point clouds are aligned in all possible 1D projective spaces, then registration is also achieved in $\mathbb{R}^d$. This strategy was first proposed as part of the IDT algorithm for colour transfer [Pitié et al., 2005, Pitié et al., 2007]. They used an optimal transport solution for registration in 1D spaces, and proposed a choice for the selection of unit vector direction $\mathbf{e}$ for solving colour transfer in 3D colour spaces ($d = 3$). In this paper, we propose several IDT solutions in 2D space to solve shape registration (c.f. section 3). We then compare our algorithms with the L2 shape registration technique proposed by Jian et al [Jian and Vemuri, 2011] (section 4), and evaluate their performance in finding correspondences between model and target shapes (c.f. Figure 1 for illustration).

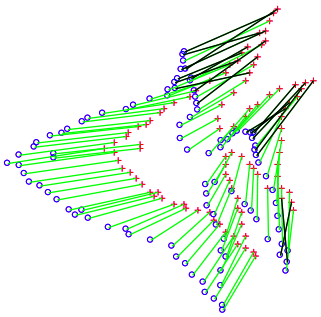


Figure 1: Shape registration (IDT-AvgD): model point set (red) and target point set (blue) with correspondences found (green for correct correspondences and black for incorrect ones).

2 State of the Art

In shape registration, many techniques have been proposed for shape registration that involve minimizing a divergence between two pdfs capturing the model point cloud $\{\mathbf{u}_i\}_{i=1,\cdots,m}$ and target point cloud $\{\mathbf{v}_j\}_{j=1,\cdots,n}$

 ISBN 978-0-9934207-2-6

[Sharp et al., 2008, Myronenko and Song, 2010, Jian and Vemuri, 2011, Ma et al., 2013]. Jian et al. propose to capture the structure of the shapes using Gaussian Mixture Models (GMMs), and estimate a parametric transformation T which registers the shapes by minimizing the L2 distance between the GMMs [Jian and Vemuri, 2011]. This technique performs very well against the state of the art and has since been extended to take into account additional information such as point correspondences or normal vectors to improve the registration result [Arellano and Dahyot, 2016, Ma et al., 2013]. Registration techniques are also important in the area of colour transfer, where the goal is to register the colour distribution of the target image to match that of the palette image. L2 registration of GMMs capturing the colour distribution of images has also been shown to give competitive results in this area [Grogan et al., 2015, Grogan and Dahyot, 2017]. Because of its performance and versatility, L2 registration [Jian and Vemuri, 2011] is used for comparison against our approach in our experimental results section.

Pitié et al. propose the Iterative Distribution Transfer (IDT) algorithm to register the 3D colour distributions of two images by iteratively projecting the 3D distributions onto several 1D subspaces before computing the registration in 1D space [Pitié et al., 2005, Pitié et al., 2007]. This dimension reduction technique reduces the computational complexity of the 3D registration problem and speeds up the process while giving good colour transfer results. However, their non-parametric transformation has been shown to cause problems when applied directly to the 3D shape registration problem [Grogan, 2017].

Algorithm 1 Iterative Distribution Transfer algorithm [Pitié et al., 2005, Pitié et al., 2007]

Require: Datasets $\{\mathbf{u}_i\}_{i=1,\cdots,m}$ and $\{\mathbf{v}_j\}_{j=1,\cdots,n}$

Require: Initialization $\mathrm{R} = [\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_4, \mathbf{e}_5, \mathbf{e}_6]$ (Eq. 1)

repeat

Compute 1D transfer functions T_1 to T_6 such that T_k is the optimal transport solution mapping the projections $\{u_i^{e_k} = \mathbf{e_k}^T \mathbf{u}_i\}$ onto the projections $\{v_j^{e_k} = \mathbf{e_k}^T \mathbf{v}_j\}$.

Compute for each point $\mathbf{u}$

$$T(\mathbf{u}) = \mathbf{u} + \mathrm{R}\begin{pmatrix} T_1(\mathbf{e_1}^T\mathbf{u}) - \mathbf{e_1}^T\mathbf{u} \\ T_2(\mathbf{e_2}^T\mathbf{u}) - \mathbf{e_2}^T\mathbf{u} \\ \vdots \\ T_6(\mathbf{e_6}^T\mathbf{u}) - \mathbf{e_6}^T\mathbf{u} \end{pmatrix} = \mathbf{u} + \mathrm{R}\begin{pmatrix} T_1(u^{e_1}) - u^{e_1} \\ T_2(u^{e_2}) - u^{e_2} \\ \vdots \\ T_6(u^{e_6}) - u^{e_6} \end{pmatrix} = \mathbf{u} + \underbrace{\sum_{k=1}^{6} (T_k(u^{e_k}) - u^{e_k})\ \mathbf{e_k}}_{\text{shift to particle } \mathbf{u}}$$

Update (i.e. move) model dataset $\mathbf{u} \leftarrow T(\mathbf{u})$

$\mathrm{R} \leftarrow$ Random rotation of R

until Convergence

Algorithm 1 presents the IDT code as shared by the authors[1]. The matrix R sets 6 directions (unit vectors) for projection, and the model dataset is moved by the sum of the 6 displacements along these 6 axes. In the IDT algorithm at initialization stage, the matrix R is chosen as:

$$\mathrm{R} = \begin{pmatrix} 1 & 0 & 0 & 2/3 & 2/3 & -1/3 \\ 0 & 1 & 0 & 2/3 & -1/3 & 2/3 \\ 0 & 0 & 1 & -1/3 & 2/3 & 2/3 \end{pmatrix} \tag{1}$$

and unit vectors $\mathbf{e}_1$ to $\mathbf{e}_6$ in $\mathbb{R}^3$ correspond to the columns of R (from left to right). The transformation T_k is the Optimal Transport solution registering $\{u_i^{e_k}\}$ to $\{v_j^{e_k}\}$ computed by:

$$T_k(u) = P_v^{-1} \circ P_u(u) \tag{2}$$

where P_v and P_u are the Cumulative distribution Functions (CDFs) of v and u respectively. These CDFs are, in practice, approximated by the Empirical distribution functions using observations $\{u_i^{e_k}\}$ and $\{v_j^{e_k}\}$ [Pitié et al., 2005, Pitié et al., 2007].

[1] https://github.com/frcs/colour-transfer

 ISBN 978-0-9934207-2-6

The convergence of the algorithm is observed experimentally after several iterations when the overall transfer function become the identity function: $T(\mathbf{u}) \simeq Id(\mathbf{u}) = \mathbf{u}$. Convergence can also be measured experimentally by computing the Kullbach-Leibler divergence between pdfs associated with each datasets at each iteration [Pitié et al., 2007]. Following from IDT algorithm, several research contributions have also used 1D projection strategy to register distributions [Bonneel et al., 2015].

3 Alternative IDTs

We are proposing next to adapt the IDT algorithm to a 2D space for solving shape registration (see Algorithm 2). Three variants are tested using $n_{\mathbf{e}} = 6$ (noted IDT-AvgD), $n_{\mathbf{e}} = 2$ (noted IDT-orthD) and $n_{\mathbf{e}} = 1$ (noted IDT-seqD) projections per iteration of the algorithm. For all three variants, the first unit vector $\mathbf{e_1}$ is randomly generated in $\mathbb{R}^2$ as follows [Muller, 1959]:

$$\mathbf{e_1} \sim \mathcal{N}(\mathbf{0}, \mathrm{I}_2) \tag{3}$$

where $\mathcal{N}(0, \mathrm{I}_2)$ is a normal distribution in $\mathbb{R}^2$ centered on the origin in $\mathbb{R}^2$ with covariance matrix equal to the identity matrix in $\mathbb{R}^2$. Then $\mathbf{e_1}$ is normalized as follow so that $\mathbf{e_1}$ is a unit vector:

$$\mathbf{e_1} \leftarrow \frac{\mathbf{e_1}}{\|\mathbf{e_1}\|} \tag{4}$$

IDT-SeqD: sequential displacements ($n_{\mathbf{e}} = 1$). Unit vector $\mathbf{e_1}$ is the only one needed for IDT-seqD, and it is renewed at each iteration of our algorithm 2. This implies that the model dataset is moved or updated using one projected shift.

IDT-OrthD: using an orthogonal basis of $\mathbb{R}^2$ ($n_{\mathbf{e}} = 2$). In this variant of IDT, vector $\mathbf{e_1}$ is randomly generated as before and $\mathbf{e_2}$ is computed to be orthogonal to $\mathbf{e_1}$ so that $(\mathbf{e_1}, \mathbf{e_2})$ defines an orthonormal basis of $\mathbb{R}^2$. In this variant of IDT, the model dataset is updated using two cumulated projected shifts. The assumption is that the projections along the two orthogonal axis are independent, and the transfer functions T_1 and T_2 model two independent transfer functions for the marginal pdfs computed in the directions of $\mathbf{e_1}$ and $\mathbf{e_2}$. The overall shift is then computed by summing their displacements.

IDT-AvgD: average displacements $n_{\mathbf{e}} = 6$. Following Pitié et al., this variant of IDT in $\mathbb{R}^2$ considers 6 unit vectors at each iteration. While $\mathbf{e_1}$, $\mathbf{e_3}$ and $\mathbf{e_5}$ are randomly generated in $\mathbb{R}^2$ (c.f. equations 3 and 4), unit vectors $\mathbf{e_2}$, $\mathbf{e_4}$ and $\mathbf{e_6}$ are orthogonal respectively to $\mathbf{e_1}$, $\mathbf{e_3}$ and $\mathbf{e_5}$. The mapping can then be applied by taking the average of the independent 1D displacements on all axes to transform the 2D sample point, we will refer to this approach by average displacements (IDT-AvgD).

4 Experimental Results

We present experimental results on the application of our IDT variants to register 2D synthetic shapes differing by a non-rigid deformation. The data[2] consists of shapes with 5 different levels of deformation, with each level including 100 instances. All experiments are performed using MATLAB R2016b on a PC with 16 GB of RAM and an Intel Xeon E5-1620 (3.7 GHz) CPU. The goal of the point set registration is to align the model point set onto the target point set. The model point set is presented using red pluses and the target point set by blue circles in the figures below. To provide a quantitative comparison, we also report the results of the state-of-the-art algorithm GMM-TPS [Jian and Vemuri, 2011], which is implemented using publicly available code[3]. The GMM-TPS algorithm estimates a parametric non-rigid transformation (Thin Plate Spline) by minimizing the L2 distance between two GMMs capturing the shape of the point clouds. Since parametric TPS transformation is designed to be smooth, it typically maintains good point correspondences after registration and we investigate

[2]obtained at `http://www.cise.ufl.edu/anand/students/chui/research.html`

[3]obtained at `https://github.com/bing-jian/gmmreg`

 ISBN 978-0-9934207-2-6

Algorithm 2 Alternative IDT.

Require: Datasets $\{\mathbf{u}_i\}_{i=1,\cdots,m}$ and $\{\mathbf{v}_j\}_{j=1,\cdots,n}$

Require: $n_\mathbf{e}$ number of projections at each iteration

repeat

Generate random unit vectors in $\mathbb{R}^2$ to create matrix R of size $2 \times n_\mathbf{e}$

Compute 1D transfer functions T_1 to $T_{n_\mathbf{e}}$ such that T_k is the optimal transport solution mapping the projections $\{u_i^{e_k}\}$ onto the projections $\{v_j^{e_k}\}$.

Compute for each point $\mathbf{u}$

$$T(\mathbf{u}) = \mathbf{u} + \mathrm{R}\begin{pmatrix} T_1(\mathbf{e_1}^T\mathbf{u}) - \mathbf{e_1}^T\mathbf{u} \\ T_2(\mathbf{e_2}^T\mathbf{u}) - \mathbf{e_2}^T\mathbf{u} \\ \vdots \\ T_{n_\mathbf{e}}(\mathbf{e_{n_e}}^T\mathbf{u}) - \mathbf{e_{n_e}}^T\mathbf{u} \end{pmatrix} = \mathbf{u} + \underbrace{\sum_{k=1}^{n_\mathbf{e}} (T_k(u^{e_k}) - u^{e_k})\ \mathbf{e_k}}_{\text{shift to particle } \mathbf{u}}$$

Update $\mathbf{u} \leftarrow T(\mathbf{u})$ (Move)

until Convergence

how the correspondences estimated using our non-parametric IDT based algorithms compare against it. Note that to improve the convergence of GMM-TPS, we use a coarse-to-fine strategy by applying deterministic annealing on the bandwidth h.

Evaluation Criterion. For quantitative comparison, we compute the recall and precision metrics for each technique. These metrics are used to quantify the accuracy of the estimated point correspondences. The estimated correspondences are computed as the points in the target and transformed model points sets that are the closest, and that fall within a given accuracy threshold in terms of pairwise distance. Recall and precision are defined as follows:

$$Recall = \frac{TP}{TP+FN} \qquad Precision = \frac{TP}{TP+FP} \tag{5}$$

where TP denotes the number of true-positives, defined as the number of ground truth corresponding point pairs that fall within a given accuracy threshold in terms of pairwise distance after registration. FN denotes the number of false-negatives, defined as the number of ground truth corresponding point pairs that fall outside the given accuracy threshold. FP denotes the number of false-positives, defined as the number of estimated correspondences that are not ground truth.

We also use the recall-accuracy curve as in [Jian and Vemuri, 2011], under different accuracy thresholds (from 0 to 0.03) to evaluate the algorithms' ability to estimate true-positive correspondences with low errors in accuracy. To explore the convergence of the algorithms and quantify the similarity between the target point cloud and transformed model point cloud at every iteration, we choose to compute the robust L2 distance measure between the point clouds. When computing the L2 distance we use the same formulation as Jian et al and set the bandwidth to $h = 0.1$ for all methods.

Results on synthetic data. Figure 2 shows a sample of the registration results obtained by the three variants of the IDT algorithm: AvgD, OrthD and SeqD, as well as GMM-TPS, when registering shapes with different levels of deformation: the level of shape deformation in the target and model point clouds increases (0.02, 0.035, 0.05, 0.065, 0.08) from left to right. The registration result, estimated correspondences, percentage recall and percentage precision are shown from top to bottom for each algorithm. From the results, we observe that all IDT methods are able to produce almost perfect alignment for all levels of deformation. On the other hand, the ability to estimate true-positive correspondences varies. This can also be seen in Figure 3, which displays plots of the average recall curves over 100 shapes for each algorithm. We see that with moderate deformation,

 ISBN 978-0-9934207-2-6

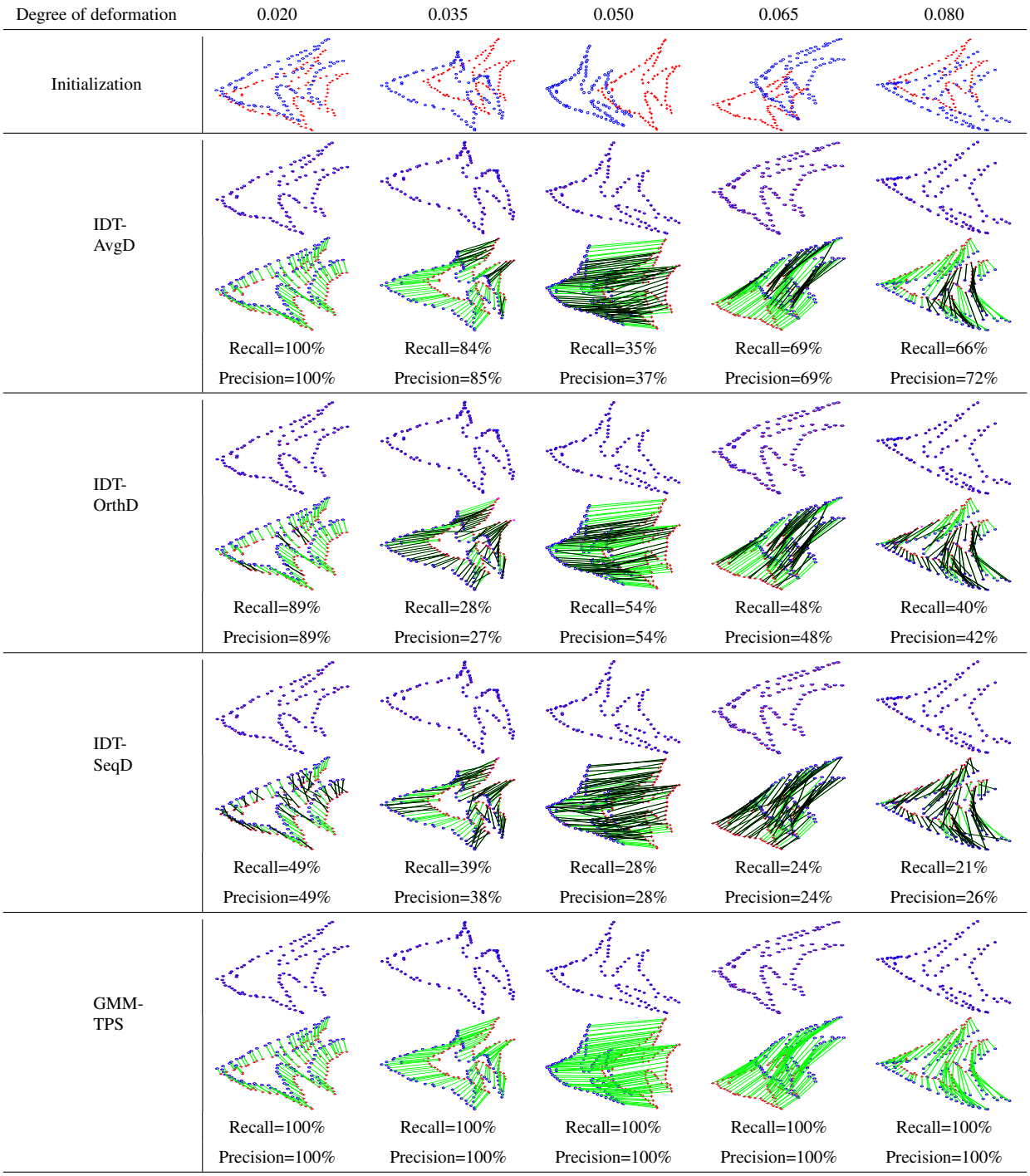


Figure 2: Comparisons of the registration results of IDT-AvgD, IDT-OrthD, IDT-SeqD and GMM-TPS with the registration result, estimated correspondences, recall and precision presented in every four rows. The first row indicates the degree of deformation, which is increasing from left to right. The second row displays the model and the target points set. The goal is to register the model point set (red pluses) onto the target point set (blue circles). In the correspondence figures, the coloured lines indicate the estimated correspondences (green = true positive, black = false positive).

IDT methods are able to achieve satisfactory results, in particular IDT-AvgD. Moreover, IDT-AvgD maintains better performance compared with the other IDT approaches as the deformation level increases. However,

 ISBN 978-0-9934207-2-6

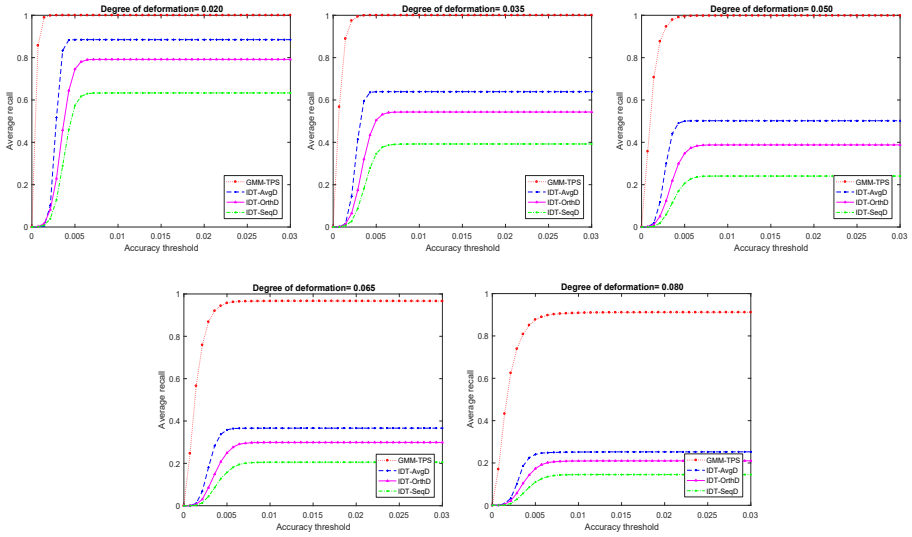


Figure 3: Average recall-accuracy curves over 100 shapes are used to evaluate IDT-AvgD, IDT-Orth, IDT-Seq and GMM-TPS under the accuracy threshold from 0 to 0.03.

the overall performance of the IDT algorithms degrades gradually as the degree of deformation in the data increases. On the other hand, the parametric transformation used by GMM-TPS generates good results, and maintains accurate point correspondences even as the level of deformation increases.

Figure 4 shows plots obtained by averaging the L2 distance, computed between the transformed model and target point sets, over 100 different shapes at each iteration in 2D. Note that we generate a sequence of projection axes and fixed these axes for all IDT algorithms to be the same. From left to right, the deformation level increases, and from top to bottom, the results for IDT-AvgD, IDT-OrthD and IDT-SeqD are presented. While all algorithms converge, we can see that the convergence behaviour of IDT-SeqD is less smooth than that of IDT-OrthD and IDT-AvgD, indicating that combining information from orthogonal bases improves the convergence of the algorithm. We can also see that although IDT-AvgD and IDT-OrthD converge to similar values by the final iteration, at earlier iterations IDT-OrthD obtains lower L2 values than IDT-AvgD. Therefore IDT-OrthD may be the preferred algorithm for convergence when considering fewer number of projection axes.

In Figure 5, for each algorithm we plot a box plot of the registration results for all 100 shapes at each level of deformation. The purpose is to show how close the IDT methods are to the state-of-the-art algorithm GMM-TPS in minimizing L2 distance between the transformed model and target point sets. We compare using the median instead of the mean to avoid the effect of outliers on the mean value. We see from the figure that the parametric GMM-TPS algorithm creates results with the lowest L2 distance, while the IDT-SeqD has the largest median L2 distance value. The median for GMM-TPS increases slightly when the degree of deformation increases, while the IDT methods maintain similar medians across all levels of deformation. The overall variability of the performance for all algorithms increases when the degree of the deformation increases. IDT-AvgD is the best of the IDT variations in term of minimizing L2 distance.

5 Conclusion & Future Work

This paper has investigated the registration of point clouds in $\mathbb{R}^2$ using a non-parametric iterative approach where at each step the problem solved is shape registration of 1D datasets. Overall our IDT based algorithms have a good performance while L2 remains the best. Note that IDTs solve iteratively the problem in 1D projective spaces with an unconstrained non parametric transformation while L2 solves it directly in 2D considering a smooth parametric transformation (TPS). TPS does not scale well in high dimensional spaces, but IDT ap-

 ISBN 978-0-9934207-2-6

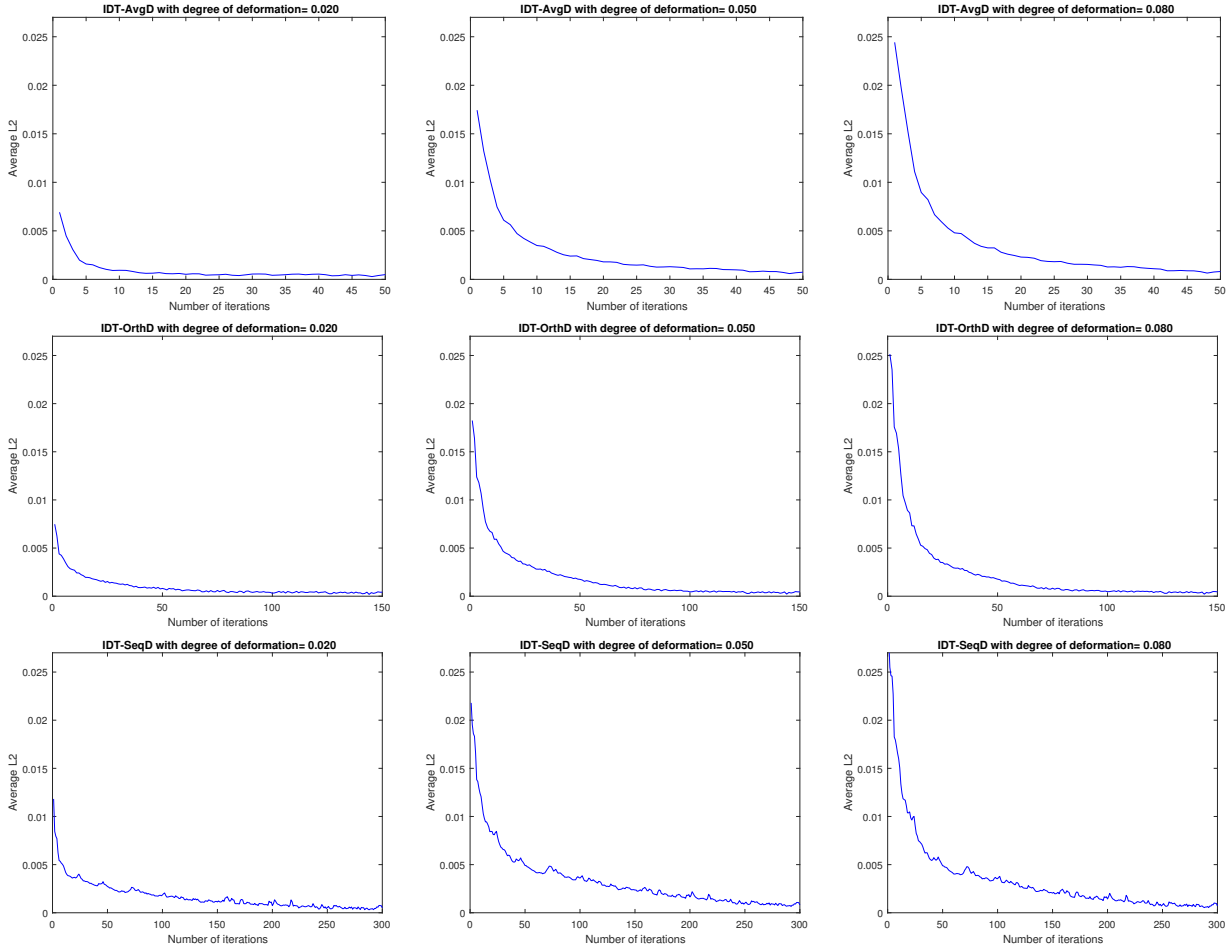


Figure 4: Evolution of the average L2 distance in 2D for 100 simulations. The abscissa indicates the number of iterations of our IDT algorithms: IDT-AvgD (top) at 20 iterations has used the same successive 20×6 projections as IDT-OrthD (middle) at 60 iterations and IDT-SeqD (bottom) at 120 iterations.

proach that considered 1D projective space has the potential to adapt well in higher dimensions, and it is also suitable for parallel optimization. Future work will then aim at assessing if a projective approach to registration can ease efficiently the computational load for registration of datasets in these cases.

Acknowledgments: The first author would like to thank Umm Al-Qura University, Saudi Arabia for funding this work as part of her PhD scholarship Programme. This work is also partly supported by the ADAPT Centre for Digital Content Technology (`www.adaptcentre.ie`) that is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

References

[Arellano and Dahyot, 2016] Arellano, C. and Dahyot, R. (2016). Robust ellipse detection with gaussian mixture models. *Pattern Recognition*, 58.

[Bonneel et al., 2015] Bonneel, N., Rabin, J., Peyré, G., and Pfister, H. (2015). Sliced and radon wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45.

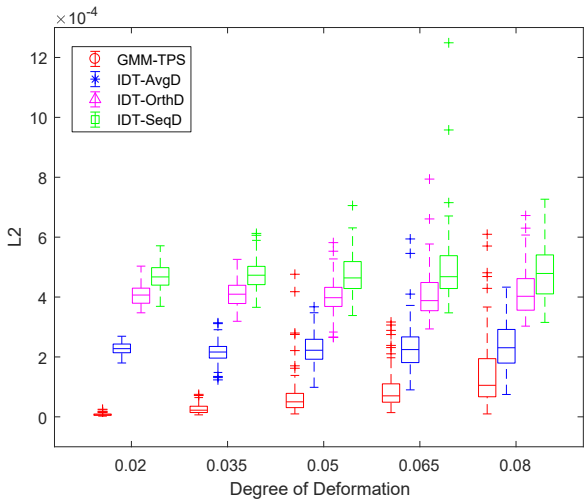


Figure 5: Box plots are used to compare the performance of IDT-AvgD, IDT-OrthD, IDT-SeqD and GMM-TPS in minimizing L2 distance. The median and the dispersion of 100 shapes are compared as the deformation degree increases.

[Grogan, 2017] Grogan, M. (2017). *Colour Transfer and Shape Registration using Functional Data Representations*. PhD thesis, Trinity College Dublin.

[Grogan and Dahyot, 2017] Grogan, M. and Dahyot, R. (2017). Robust registration of gaussian mixtures for colour transfer. Technical report, `https://arxiv.org/abs/1705.06091`.

[Grogan et al., 2015] Grogan, M., Prasad, M., and Dahyot, R. (2015). L2 registration for colour transfer. In *European Signal Processing Conference (Eusipco)*, Nice France.

[Jian and Vemuri, 2011] Jian, B. and Vemuri, B. (2011). Robust point set registration using gaussian mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1633 – 1645.

[Ma et al., 2013] Ma, J., Zhao, J., Tian, J., Tu, Z., and Yuille, A. L. (2013). Robust estimation of nonrigid transformation for point set registration. In *IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, USA USA.

[Muller, 1959] Muller, M. E. (1959). A note on a method for generating points uniformly on n-dimensional spheres. *Communications of the ACM*, 2(4):19–20.

[Myronenko and Song, 2010] Myronenko, A. and Song, X. (2010). Point set registration: Coherent point drift. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(12):2262–2275.

[Pitié et al., 2005] Pitié, F., Kokaram, A. C., and Dahyot, R. (2005). N-dimensional probability density function transfer and its application to color transfer. In *Tenth IEEE International Conference on Computer Vision (ICCV 2005)*, volume 2, pages 1434 – 1439.

[Pitié et al., 2007] Pitié, F., Kokaram, A. C., and Dahyot, R. (2007). Automated colour grading using colour distribution transfer. *Computer Vision and Image Understanding journal (Special Issue on Color Image Processing)*.

[Sharp et al., 2008] Sharp, G. C., Lee, S. W., and Wehe, D. K. (2008). Maximum-likelihood registration of range images with missing data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(1):120–130.

 ISBN 978-0-9934207-2-6

Detecting and Tracking Meeting Participants using Motion Heat Maps

Nahlah Algethami and Sam Redfern

n.algethami3@nuiglway.ie,sam.redfern@nuigalway.ie

Abstract

This paper presents an implementation of a tracking algorithm using a motion heat maps technique, for tracking the movements of participants in meeting environments. The tracking algorithm consists of two processing stages: firstly, for each video frame, a motion heat map is generated based on accumulated frame differencing; secondly, participant detection and tracking is performed based on the heat map images. We identify and segment blobs associated with high intensity motion, then the locations of people are identified using a contour detection method. The performance of the proposed technique is evaluated using the AMI public meeting corpus. Our intention is that the output of the tracking algorithm will be used to direct the information obtained from other cameras and thereby improve meeting analysis modules such as human activity recognition.

Keywords: Image Processing, Motion Heat Maps, People Tracking, Meeting Analysis.

1 Introduction

Despite the importance of meetings for such tasks as knowledge sharing, problem solving and decision-making, it is not always possible for people to attend. The uses of smart meeting systems include the mitigation of information loss due to non-attendance, and therefore smart systems aim to automatically capture and analyze meetings, and to provide meeting browsers for efficient review of captured data which help people to understand (browse) the meeting topic quickly

A number of smart meeting systems have been published in the literature, including systems developed by Microsoft (Cutler et al., 2002) and Carnegie Mellon University (Gross et al., 2000). Features include broadcast and future review of meeting records. The most recent smart meeting systems aim to use audio and visual monitoring to automatically identify the main events taking place - for example the time when a presentation begins or the time when a new participant enters the room (Ronzhin and Karpov, 2015). Recent research projects in this space have sought to further develop the automatic analysis of meetings. Notable examples include (i) the Multi Modal Meeting Manager (M4), which is concerned with structuring and browsing a meeting automatically (McCowan et al., 2003); (ii) the Augmented Multiparty Interaction (AMI), which aims to annotate a meeting and assist remote users (McCowan et al., 2005a); and (iii) the ICSI Meeting Project, which aims to automatically generate meeting transcripts and automatic analysis of meetings such as meeting summaries (Janin et al., 2003).Public repositories of visual and audio data have been made available by these projects, for future work in the area.

Smart meeting systems use various modalities including speech, vision, and others. Since speech is the predominant communicative modality in meeting analysis, many meeting applications focus on speech processing methods. For Example, in (Brdiczka et al., 2005), speech activity detection is used to detect group interaction. Earlier work prototyped a meeting recorder and browser which automatically produced meeting transcrips and meeting summaries of audio recording (Kubala et al., 1999). Another early meeting application also sought to

 ISBN 978-0-9934207-2-6

address the problem of speech recognition and dialogue summarization (Waibel et al., 1998), by presenting a meeting browser to allow the user to search and browse a meeting.

Visual modality also plays an important role in meeting analysis, and visual processing aims to track participants' movements, automatically recognise and classify their actions, and track their focus of attention. For instance, a real time multiple head tracker was proposed by (Hradis and Jurnek, 2006). Here, the head is detected based on a skin color model, then background subtraction and components analysis as performed, and finally a Kanade-Lucas-Tomasi (KLT) feature tracker is combined with a color model to track detected objects. Another system performed action recognition in order to classify individual actions in a meeting scenario, using global motion features (Zobl et al., 2003). A number of promising research activities have been concerned with multimodal processing, in order to achieve better accuracy in meeting analysis, e.g. (Al-Hames et al., 2005). The visual tracking of participants' movements can play a significant role in meeting analysis, as it can be used as a source of high level information to assist the classification of activities in smart meeting environments. In this paper, we present details of our ongoing research project whose goals include the creation of an annotated time line of participant movements. The contextual information obtained from overhead cameras will be used to direct and therefore improve the information obtained from personal cameras - thereby enhancing action recognition. The initial step to creating this meeting time line is to track people's movements. We use motion heat maps to achieve this, and the performance of our proposed approach is tested on the AMI public meeting corpus (McCowan et al., 2005a).

2 Related Work

2.1 Semantic analysis of meeting videos

Semantic analysis of meeting videos seeks to extract semantic information in order to classify group actions. For example, two-layered Hidden Markov Models have been used to classify individual and group actions (Moehrmann et al., 2010); (McCowan et al., 2005b); (Zhang et al., 2004). (Hakeem and Shah, 2004) proposed a framework for classification of meeting videos, in which the head and hands are tracked, then events such as raising or lowering of hands may be detected, and finally a rule based method may be used to classify actions such as voting. In these previous projects, high-level contextual information is extracted from audio and visual data in order to classify group actions. This high level context (or semantic) information is not adopted for low level processing. Our work differs in that the high level context information obtained from overhead cameras (i.e., number and location of participants) can be used to improve the information obtained from personal cameras and therefore to have an impact on low level visual processing, in order to enhance action recognition.

A small number of researchers have previously proposed the extraction of semantic information with the goal of using it as a guidance to low level visual processing, or to select relevant fixed cameras. (Dai et al., 2007) proposed an online action recognition system where a set of visual detection and tracking modules are applied to classify meeting contexts as: speech (presentation), discussion, and meeting break. Low level visual processing of individual events is directed by this high level context. For example, if the current meeting context is speech or discussion, further individual activity analysis such as head pose estimation and hand tracking is applied. Furthermore, a set of active and static cameras is used to capture visual information at multiple levels. At the lowest level, tasks such as participant detection are performed, and then a participant tracking algorithm is applied in order to extract higher level semantic information (i.e. people's trajectories), and finally activities are identified (e.g. person enters the room) from this semantic information. The most appropriate camera is dynamically selected and higher semantic information such as face recognition is derived (Trivedi et al., 2005).

The MeetingAssistant application (Yu et al., 2007) uses an overhead camera and four personal cameras. Face recognition is performed to automatically control the focus and orientation of personal cameras, in order to keep participants centrally in view. Finally, an audio-based algorithm to direct the camera to the active speaker in a smart meeting room is presented in (Ronzhin Al L, 2011).

In our research, high level visual information obtained from an overhead camera is used to build prior knowledge of the environment. This knowledge can then be used to perform some low-level analysis such as

 ISBN 978-0-9934207-2-6

discarding frames from a personal camera when there is someone else moving behind the owner of that camera. Therefore, errors may be reduced and the accuracy of action recognition methods increased. This high level information can also be used to build a robust background image as the background extraction is one of the most important methods in object detection and tracking.

2.2 Vision based participant tracking

Another area of related work is concerned with the tracking of people; many researchers have worked on vision based participant tracking in meeting environments using single or multiple cameras, standard video cameras or omnidirectional cameras. Examples of tracking approaches from a single video sequence include that of (Hradis and Jurnek, 2006) and (Nait-Charif and McKenna, 2003). (Hradis and Jurnek, 2006) perform their tracking task in two stages: firstly, the head is detected based on a skin color model, background subtraction and components analysis; secondly, a Kanade-Lucas-Tomasi (KLT) feature tracker is combined with a color model to track detected objects.(Nait-Charif and McKenna, 2003) proposed a reliable head tracking algorithm which was evaluated for the PETS-ICVS 2003 video data sets. Firstly, the head is modelled using a fixed ellipse that represents the head boundary, and the ellipse's interior region is represented using a color histogram. This head likelihood model is used by a particle filter based on Iterated Likelihood Weighting (ILW) in order to track participants in the meeting room.

(Potucek and Sumec, 2004) propose methods for tracking people in meeting rooms with three cameras. A skin color algorithm is used to detect head and hands from a video frame, and a face detector is used to detect the likely presence of a face based on a skin color blob. Each object is classified as either head or hands, and is assigned to a meeting participant. To keep information about the position of the hands and face or the direction of the head, a tracking method is needed to assign an object in frame i with the corresponding object in frame i+1. For these purposes, Potucek and Sumec propose the use of Kalman filtering in order to predict the position of the object in the next frame based on the previous position. Therefore,the boundary of the search area of an object is defined.

As omnidirectional images produce a large view with low cost, many tracking algorithms have been developed using omnidirectional views. In (Potucek et al., 2007), two different tracking algorithms are tested and compared using panoramic video sequences and two perspective cameras. The first tracking method is based on skin color segmentation and face detection, and the tracking is applied based on the position of the previous frames. The second tracking algorithm uses skin color segmentation and background subtraction to detect the head, and then a KLT feature based tracker is used. (Patil et al., 2004) proposed a fast detection and tracking algorithm from panoramic images where a set of cameras is used to produce a high resolution panoramic image. CAMEO (The Camera Assisted Meeting Event Observer) is a physical system to recognize people's actions in a meeting and then generate a summary of events occurring in the meeting. CAMEO's tracking system consists of many components including a region of interest (ROI) extractor, face detector, shape detector, mean shift color tracker, and Bayesian network-based action recognizer.

(Huang and Trivedi, 2003) present and compare two 3D real time trackers; namely a rectilinear video array tracker (R-VAT) and an omnidirectional video array tracker (O-VAT). (Focken and Stiefelhagen, 2002) present a 3D vision based tracking algorithm where the tacking is done using three cameras. Foreground objects (i.e., people) are extracted from the three images, and their 3D location estimated. A probabilistic tracker is implemented to track the person based on their 3D location.

Most of these tracking algorithms used skin color models, background subtraction and face detectors to detect people in smart meeting rooms, followed by their tracking using a technique such as Kalman filtering. In this paper, we implement a simple and effective tracking algorithm using a motion heat maps approach.

3 A Participant Tracking Algorithm Using Motion Heat Maps

To achieve our goal towards building an annotated time line for the movement of people in a meeting room, a heat map-based approach is proposed. A motion heat map is a 2D visual representation of frame difference

 ISBN 978-0-9934207-2-6

data values, whereby intensity (or 'heat') is presented using colors ranging from blue to red. It is a 2D color histogram representing the region with high motion as well as region with very low motion. This approach has often been used for crowd density estimation in surveillance applications, as well as in the analysis of people's behavior. For example, (Parzych et al., 2013) generated heat maps for people's movements in sales room, for use in customer behavior analysis. The use of heat maps to recognize group activity was first introduced in (Lin et al., 2015).

To the best of our knowledge, our work represents the first time where the motion heat maps are used to track the movement of people in meeting rooms. Tracking based on motion cues provides good reliability for tracking people in meeting spaces where there is no constraints on participant movements and their features are not clear from overhead cameras.

3.1 Algorithm

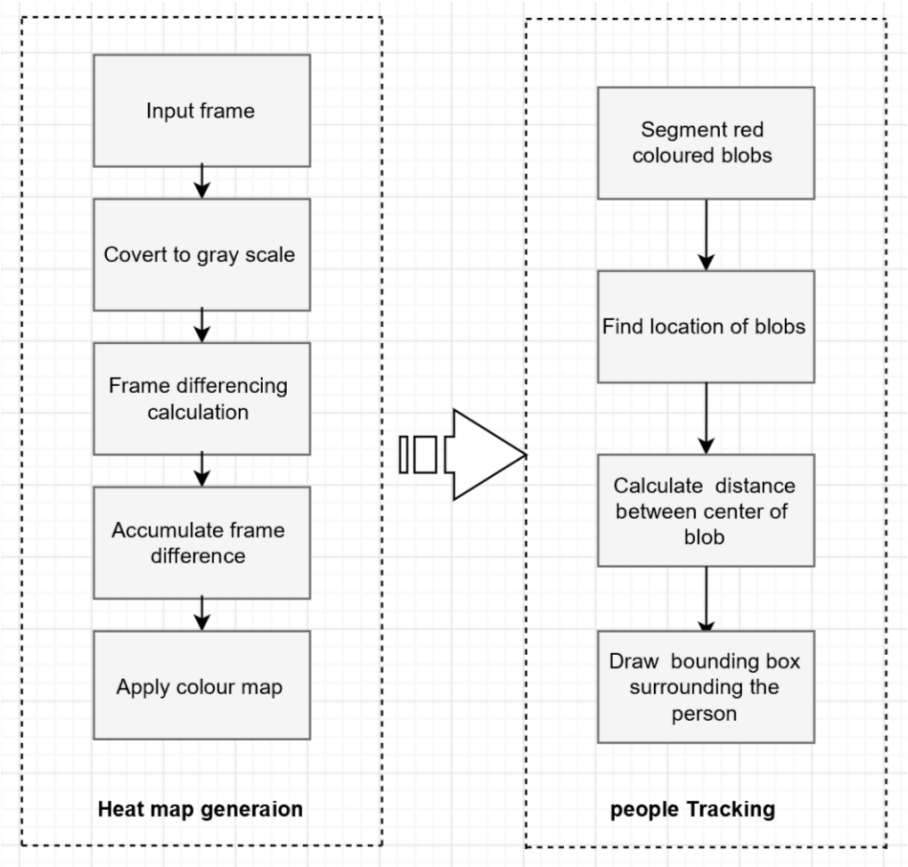


Figure 1: People tracking algorithm overview

Our proposed tracking algorithm is currently implemented using the Open CV library version 3.0. As shown in figure 1, our detection and tracking algorithm consists of two stages: firstly, the heat map image is generated from each frame in video sequences; which is accumulated frame differencing for the next 100 frames. Secondly, the tracking algorithm is applied. The algorithm operates on greyscale images. There are many techniques for motion detection such as background subtraction, Gaussian mixture and frame consecutives differences. Our current implementation uses frame differencing since it is simple and provides a satisfactory result. A window size of 100 frames is used to calculate the heat map for each frame in video sequences; i.e. accumulated frame differencing from frames 0 to 99 is accumulated to generate the heat map for the first output frame. The next step is to detect motion. This is done by, firstly, segmention of regions of high movement intensity (depicted as red blobs in most heat map visualisations). Therefore, red color segmentation is applied which

returns a binary mask where red pixels are set to 1 and other pixels set to 0 (OpenCV-documentation, 2017a). Morphological dilation is then applied to fill in white holes. Contour detection is applied to find the location of these regions(OpenCV-documentation, 2017b). Only contours with large areas are chosen, and bounding rectangles and the centre of each blob are calculated. In some cases, the head and hands for a person may be detected separately (as shown in figure 2). Since the goal is to detect and track individual people, distance is calculated between each two rectangles, and when the distance is too small, two rectangles are merged.

(a) head and hands of person 4 is detected separately

(b) merging two blobs of person 4

Figure 2: Tracking Result

3.2 Experimental data

The Augmented Multiparty Interaction (AMI) (McCowan et al., 2005a) public meeting corpus is used to evaluate the performance of our proposed approach. This is a multimodal data set in which meeting data was collected in three differently equopped meeting rooms at the University of Edinburgh (U.K.), Idiap (Switzerland), and the TNO Human Factors Research Institute (The Netherlands). The work discussed in the current paper was tested using videos captured at the Edinburgh meeting room. This meeting room was equipped with static cameras to capture individual details, as well as room overview. Four close up cameras are used to capture individual details for each participant. For capturing room overview, a corner camera and ceiling camera was used. Figure 3 shows different view cameras in the Edinburgh meeting room.

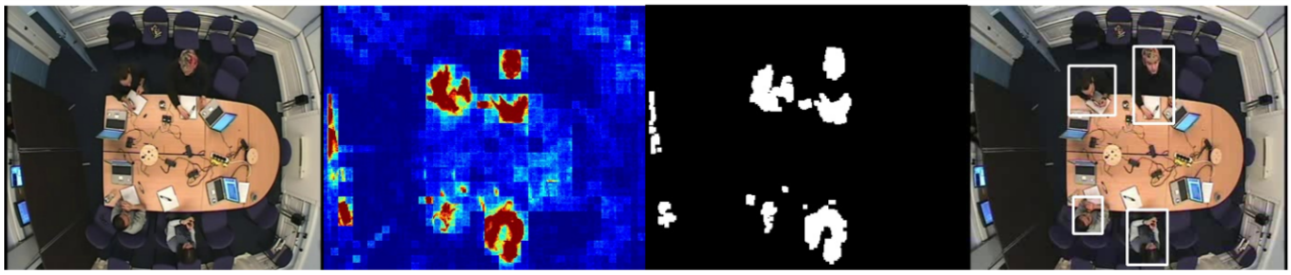
Figure 3: different view cameras in Edinburgh meeting room.

3.3 Experiments and Discussion

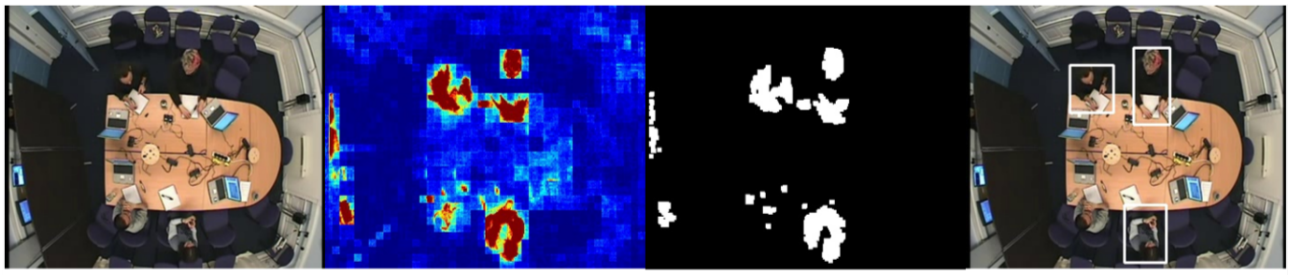
Our proposed detection and tracking method is applied in meeting video sequences captured using the overhead camera. Visual video processing experiments of our tracking algorithm are shown in Figures 4a, 4b and 4c. Our proposed method gives promising results (a) where four meeting participants are detected correctly. However, this is an early version of our algorithm, and a person has not been detected correctly when he has made a small motion during the 100 window size motion calculation: see figure 4b. One possible improvement we are considering is that when a person is detected in many frames and missed in the next few frames, the tracking

 ISBN 978-0-9934207-2-6

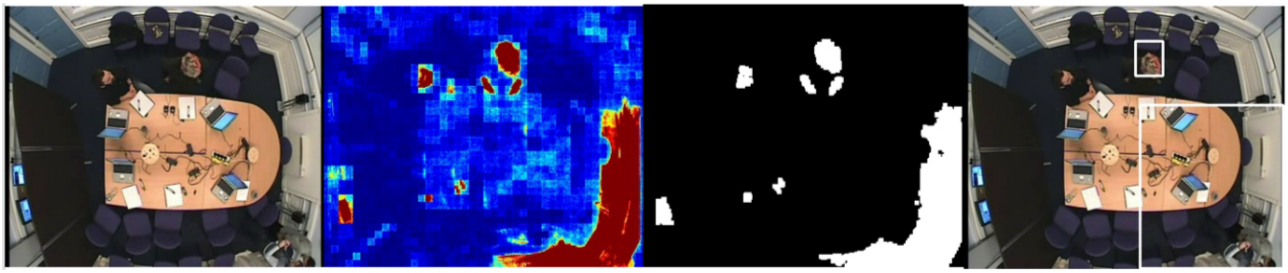
algorithm assumes the person is still there. Moreover, when two people walk close to each other, their motion blobs will overlap; therefore the algorithm detects both people as one person: see figure 4c. A smaller window size can be used in these circumstances to avoid blob overlap. Furthermore, assuming small distances between the head and hands of a person may, in some situations, lead to errors when two participants are close to each other. Considering the fixed room layout (i.e. table location), the accuracy of the tracking method and head-hand merging stage can be improved significantly. We are currently working on all possible improvements to build a more robust participant detection and tracking algorithm.



(a) Tracking result



(b) tracking result when the motion is small



(c) tracking result when two people walk close to each other

Figure 4: visual processing of the tracking algorithm where the first image is the input frame, next is the motion heat map, third is a binary mask of red blob segmentation and the fourth is final result

4 Conclusion

Participant tracking is very important as the tracking information can be used as a source of data in subsequent meeting analysis modules, for example in action recognition or in selecting appropriate personal cameras for detailed analysis. In this paper, a visual tracking algorithm using motion heat maps is implemented. Our initial experimental results are encouraging. However, in some cases the current algorithm fails to detect or separate people, i.e. when there is small motion or when there are overlaps between blobs. We are currently investigating a number of improvements to cope with these problems.

In the future work, we plan to use the high level information obtained from our tracking algorithm to generate an indexed time line indicating people's movement in meeting space. Moreover, we plan to use this information as a cue for extracting useful features from other cameras, in order to enhance our individual

 ISBN 978-0-9934207-2-6

action recognition system. Finally, we plan to use these heatmaps to perform other low level visual processing, for example frames with motion may be discarded, and therefore a more robust background model may be generated using the median filtering approach.

References

Al-Hames, M., Dielmann, A., Gatica-Perez, D., Reiter, S., Renals, S., Rigoll, G., and Zhang, D. (2005). Multimodal integration for meeting group action segmentation and recognition. In *International Workshop on Machine Learning for Multimodal Interaction*, pages 52–63. Springer.

Brdiczka, O., Maisonnasse, J., and Reignier, P. (2005). Automatic detection of interaction groups. In *Proceedings of the 7th international conference on Multimodal interfaces*, pages 32–36. ACM.

Cutler, R., Rui, Y., Gupta, A., Cadiz, J. J., Tashev, I., He, L.-w., Colburn, A., Zhang, Z., Liu, Z., and Silverberg, S. (2002). Distributed meetings: A meeting capture and broadcasting system. In *Proceedings of the tenth ACM international conference on Multimedia*, pages 503–512. ACM.

Dai, P., Tao, L., and Xu, G. (2007). Dynamic context driven human detection and tracking in meeting scenarios. In *VISAPP (Special Sessions)*, pages 31–40.

Focken, D. and Stiefelhagen, R. (2002). Towards vision-based 3-d people tracking in a smart room. In *Multimodal Interfaces, 2002. Proceedings. Fourth IEEE International Conference on*, pages 400–405. IEEE.

Gross, R., Bett, M., Yu, H., Zhu, X., Pan, Y., Yang, J., and Waibel, A. (2000). Towards a multimodal meeting record. In *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, volume 3, pages 1593–1596. IEEE.

Hakeem, A. and Shah, M. (2004). Ontology and taxonomy collaborated framework for meeting classification. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 4, pages 219–222. IEEE.

Hradis, M. and Jurnek, R. (2006). Real-time tracking of participants in meeting video. In *Proceedings of CESCG*.

Huang, K. S. and Trivedi, M. M. (2003). Video arrays for real-time tracking of person, head, and face in an intelligent room. *Machine vision and applications*, 14(2):103–111.

Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., et al. (2003). The icsi meeting corpus. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 1, pages I–I. IEEE.

Kubala, F., Colbath, S., Liu, D., and Makhoul, J. (1999). Rough'n'ready: a meeting recorder and browser. *ACM Computing Surveys (CSUR)*, 31(2es):7.

Lin, W., Chu, H., Wu, J., Sheng, B., and Chen, Z. (2015). A heat-map-based algorithm for recognizing group activities in videos. *arXiv preprint arXiv:1502.06076*.

McCowan, I., Bengio, S., Gatica-Perez, D., Lathoud, G., Monay, F., Moore, D., Wellner, P., and Bourlard, H. (2003). Modeling human interaction in meetings. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 4, pages IV–748. IEEE.

McCowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., et al. (2005a). The ami meeting corpus. In *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, volume 88.

McCowan, L., Gatica-Perez, D., Bengio, S., Lathoud, G., Barnard, M., and Zhang, D. (2005b). Automatic analysis of multimodal group actions in meetings. *IEEE transactions on pattern analysis and machine intelligence*, 27(3):305–317.

Moehrmann, J., Wang, X., and Heidemann, G. (2010). Motion based situation recognition in group meetings. In *IS&T/SPIE Electronic Imaging*, pages 75380N–75380N. International Society for Optics and Photonics.

Nait-Charif, H. and McKenna, S. J. (2003). Head tracking and action recognition in a smart meeting room. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*. Citeseer.

OpenCV-documentation ((accessed May 20, 2017)a). InRange open CV method at: Opencv 3.0.0, documentation, org.opencv.core package, Core Class. http://docs.opencv.org/java/3.0.0/.

OpenCV-documentation ((accessed May 20, 2017)b). Find Contour open CV method at: Opencv 3.0.0, documentation, org.opencv.imgproc package, imgproc Class.http://docs.opencv.org/java/3.0.0/.

Parzych, M., Chmielewska, A., Marciniak, T., Dabrowski, A., Chrostowska, A., and Klincewicz, M. (2013). Automatic people density maps generation with use of movement detection analysis. In *Human System Interaction (HSI), 2013 The 6th International Conference on*, pages 26–31. IEEE.

Patil, R., Rybski, P. E., Kanade, T., and Veloso, M. M. (2004). People detection and tracking in high resolution panoramic video mosaic. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 2, pages 1323–1328. IEEE.

Potucek, I., Beran, V., Sumec, S., and Zemcik, P. (2007). Evaluation and comparison of tracking methods using meeting omnidirectional images. In *Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MLMI)*, volume 12.

Potucek, I. and Sumec, S. (2004). Participant activity detection by hands and face movement tracking in the meeting room. In *Computer Graphics International, 2004. Proceedings*, pages 632–635. IEEE.

Ronzhin, A. and Karpov, A. (2015). A software system for the audiovisual monitoring of an intelligent meeting room in support of scientific and education activities. *Pattern Recognition and Image Analysis*, 25(2):237–254.

Ronzhin Al L, K. A. (2011). System of audio-visual streams recording and synchronization for the smart meeting room.

Trivedi, M. M., Huang, K. S., and Mikic, I. (2005). Dynamic context capture and distributed video arrays for intelligent spaces. *IEEE Transactions on Systems, Man, and Cybernetics-PART A: Systems and Humans*, 35(1):145–163.

Waibel, A., Bett, M., Finke, M., and Stiefelhagen, R. (1998). Meeting browser: Tracking and summarizing meetings. In *Proceedings of the DARPA broadcast news workshop*, pages 281–286. Citeseer.

Yu, Z., Ozeki, M., Fujii, Y., and Nakamura, Y. (2007). Towards smart meeting: enabling technologies and a real-world application. In *Proceedings of the 9th international conference on Multimodal interfaces*, pages 86–93. ACM.

Zhang, D., Gatica-Perez, D., Bengio, S., McCowan, I., and Lathoud, G. (2004). Modeling individual and group actions in meetings: a two-layer hmm framework. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*, pages 117–117. IEEE.

Zobl, M., Wallhoff, F., and Rigoll, G. (2003). Action recognition in meeting scenarios using global motion features. In *Proc. PETS-ICVS*, pages 32–36.

Gabor and HOG approach to facial emotion recognition

Ryan Melaugh, Nazmul Siddique, Sonya Coleman and Pratheepan Yogarajah

Ulster University

Abstract

Automated facial emotion recognition is an essential step for proper Human-Machine Interaction (HMI) since much of human-human interaction occurs outside of our speech and tone of voice. While other papers have tried singular approaches to this problem, we explore a combination of Gabor and Histogram of Oriented Gradients (HOG) to accurately recognise emotion from still images. This novel method has outachieved many competing Gabor alone and HOG alone methods with future work aiming to explore larger databases and classifiers.

Keywords: Emotion Recognition, Gabor, HOG, Neural Network, SVM

1 Introduction

Most accurately described as an experience, our emotions play an essential role in how we interact with the world around us [Weseley and McEtarffer, 2007]. Myers defines emotion as a mix of psychological activation, expressive behaviours, and conscious experience [Myers, 2000]. This is built on Schachter's two-factor theory which describes emotional response as a combination of "both our physical responses and our cognitive labels (our mental interpretations)" [Weseley and McEtarffer, 2007]. Typically when working with emotions, researchers often reference Ekman's work, reducing someone's entire emotional experience to just six emotions: joy, sadness, anger, disgust, surprise, and fear [Ekman, 1970]. Facial expression is one of the many ways we pick up on the emotions of other people. Ekman created a coded system to determine an emotion called FACS (Facial Action Coding System) [Ekman and Friesen, 1978]. This is able to score muscle movements in the face to produce a unique FACS code. The code itself only relates to the positioning of the muscles but the relative positioning of muscles is precisely what would be used later by researchers to detect emotion. While the FACS codes are not typically used in present-day computer algorithms for emotion detection, it formed the psychological basis for the feature extraction stage - where the positions of the parts of the face are taken together to create a feature vector. In order for HMI to really take off, it is important that the machine be able to understand our expressions.

According to Fernandes and Bala, the latest state of the art face recognition techniques are Discrete Cosine Transform (DCT), Hierarchal Dimensionality Reduction, Local and Global combined Computational Features (LGFT), Combined Statistical Moments and score Level Fusion Techniques (LFT) [Fernandes and Bala, 2017]. After the face is detected, a feature selector can be applied to the facial area followed by a classifier. Some emotion detections from facial expression include Deng et al. who use Gabor, PCA, and Linear Discriminant Analysis [Deng et al., 2005], and [Li et al., 2013], use HOG and LBP to detect micro expressions, which are rapid involuntary expressions revealing true emotions [Li et al., 2016]. However, this paper uses the Viola-Jones (V-J) method for face detection because of its low computational costs, ease of use, and integration with the OpenCV software [Viola and Jones, 2001] [Castrillon-Santana et al., 2007]. A more recent application of the V-J method is seen in Shan's 2011 paper focusing on gender determination from the face [Shan, 2011]. The V-J method in this paper is followed by a tight facial cropping, Gabor, HOG, feature scaling via a Standard Scaller, and finally feature reduction via PCA. Classification occurs via an artificial neural network (NN) or support vector machine (SVM) to detect the emotion.

 ISBN 978-0-9934207-2-6

2 Methods

An overview of the steps can be seen in Figure 1, while Figure 2 shows images from the image, preprocessing, and feature extraction steps. It should be noted that Figure 2b-d are not in scale with Figure 2a, which has been reduced in size for inclusion in this report. The images were first read in from the popular JAFFE database, talked more about in Section 4. These images go through preprocessing, followed feature extraction, feature selection and finally classification before an output is reached. The preprocessing steps include detection of the face using the V-J method, and cropping tightly around the face. Gabor filters are applied to the cropped image and then HOG is preformed to create a feature vector. This feature vector is normalised by a standard scaler and reduced by PCA before being passed separately to a Support Vector Machine (SVM) and an artificial neural network (NN). HOG, SVM and NN used the Scikit Image [scikit-image developers, 2017] and Scikit Learn [scikit-learn developers, 2017] implementation while V-J method, and Gabor Filters used the OpenCV library [OpenCV-team, 2017]. The computer used for testing had an i7 processor and ran on Windows 10. Anaconda 2.7 ran the program.

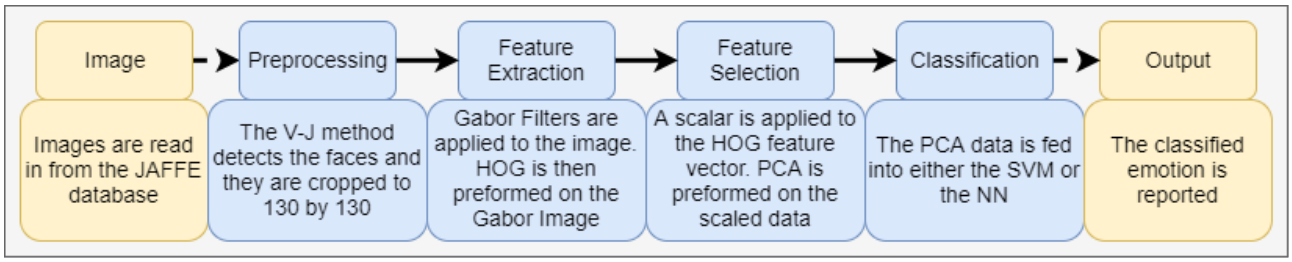


Figure 1: Overview of the methods.

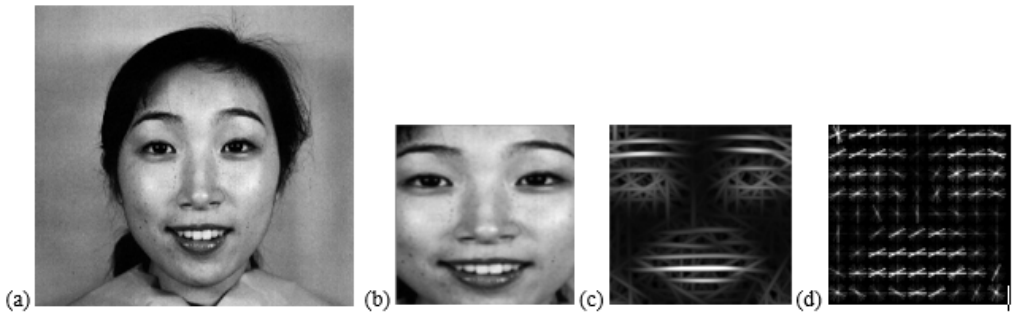


Figure 2: (a) Original JAFFE image. (b) Cropped image from the V-J method and cropping method. (c) Resulting Gabor image. (d) Resulting HOG image.

2.1 Prepocessing

Information within the image background, such as hair etc. can negatively affect emotion recognition methods and need to be trimmed. Thus leaving the desirable features, which the method requires, the facial features and their relative muscle position. Therefore, the images are cropped to a tight frame around the face using the Viola-Jones (V-J) method [Viola and Jones, 2001]. The V-J method is a Haar cascading classifier, which uses the cascade function trained using 'positive' images and 'negative' images. Positive images are images that match the target object while negative images are irrelevant images that are used to distinguish between what is desired and what is not. We used open source code created by various authors to detect the face and eye region under the Intel Licence Agreement, also used by OpenCV to perform the V-J method [OpenCV-team, 2017] [Castrillon-Santana et al., 2007]. If the eyes could not be detected - such as in the case where the eyes were

 ISBN 978-0-9934207-2-6

mostly closed or in a shape unfamiliar to the cascade - the image would be cropped based on the face area as a whole rather than the finer cropping created by detecting the eyes. There was no variation in head pose, negating any need for a method in head pose estimation. The cascading classifier selects the region of face without consideration for the size of the area or image. This means that each image is effectively cropped to its own size, all within a small margin of each other, but nonetheless any difference will result in a feature vector of different lengths, which is not allowed by the chosen learning techniques. Therefore, each image has to be resized to an identical size; 130x130 pixels. This was the maximum size achievable after cropping for all images, thus minimising data lost by resizing and avoiding entirely adding padded data by resizing larger. The cropping is seen in the change from Figure 2a to Figure 2b. Images of equal size are essential for proper classification by both the SVM and NN.

2.2 Gabor Filter

Following the preprocessing stage, the novelty in our approach lies in the use of Gabor filtering prior to processing by HOG. Some papers have used an edge detection method prior to processing by HOG [Li et al., 2013]. These edge techniques are often the likes of canny edge detection [Canny, 1986], Sobel [Sobel, 2015], or even just thresholding [Li and Lee, 1993]. Their purpose is solely in their name, they only reveal the edges in the image. This eliminates unwanted detail, similar to the preprocessing stage (provided the details are not strong enough to be highlighted in themselves by the edge detector) leaving an outside 'wireframe' look to the image. This results in the details we actually desire such as the contours of the lips, eyes, eyebrows and their position within a clean environment. The Gabor filter is comparable to the method humans use for image recognition [Marĉelja, 1980]. That is, like the human eye, the Gabor Filter analyses changes in lighting and texture in order to analyse the image. In particular, the Gabor filter targets edge and texture changes in an image highlighting the prominent features. We use the Gabor Filter to exaggerate the orientation of the facial images, for example Gabor turns smiles into triangular shapes as seen in Figure 2c The exaggerated and sharper edges of the facial features become useful and simpler features - compared to the original image - by creating a more distinguished orientation of the features for the HOG feature descriptor. It should be noted that like other techniques, lighting can affect the results of the filters, generating shapes where there are none. This can be mitigated by preprocessing techniques to reduce the effect of lighting, but is beyond the scope of this paper. The Gabor filters are described in equation 1:

$$f(x,y,\lambda,\theta,\psi,\sigma,\gamma) = exp\left(\frac{(xcos\theta+\gamma sin\theta)^2+\gamma^2(-xsin\theta+\gamma cos\theta)^2)}{(2\sigma^2)}\right) cos\left(2\pi\frac{xcos\theta+\gamma sin\theta}{\lambda}+\psi\right) \quad (1)$$

Where x and y in the function are the co-ordinates of the pixel being analysed. λ is the desired wavelength for the Gabor filter, it is a sine factor. θ is the desired angle and to obtain a Gabor image such as the one in Figure 2c, multiple filters need to be created over a range of θ from 0 to 180 degrees. ψ is the phase offset, which shifts the sine function. The standard deviation is represented by σ and γ controls the elliptical nature of the filter. This value ranges between 0, nearly a straight line, and 1, a full circle. After the filters are defined, seen in Figure 3, they are applied one by one to the image layering on top of each other until a final image is created.

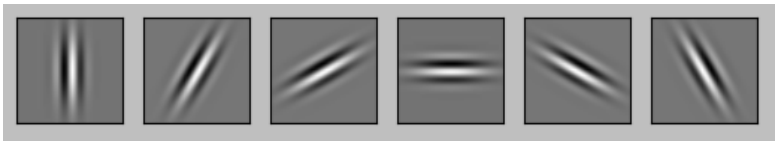


Figure 3: The Gabor Filter Bank (images not to scale). The real size of each filter is 51x51.

The parameter values used are σ = 4.5, λ = 8.25, γ = 0.4, ψ = 0.9, with six σ orientations, equally spaced between 0-180 degrees. This is then applied to an area of 51x51 pixels. Figure 3 contains the filters created by

 ISBN 978-0-9934207-2-6

these values. Figure 2c illustrates the resulting feature map after the Gabor Filter Bank is applied to the 'happy' image in Figure 2b. The filters were applied one by one using OpenCV's Filter2D [OpenCV-team, 2017]. We pass the pre-processed image into the Gabor Filter in order to detect, and importantly for the HOG, clearly define the prominent features of the face such as the eyes, eyebrows, nose, mouth and any laugh lines. These make up the set of features whose muscle movements amount to tell-tale signs of an emotional response, such as smiling, frowning, furrowing of the brow, etc. Notice the sharp diagonal lines of the mouth, and elevated, rounded eyebrows, which expose a clear happy expression from the face which corresponds to the emotional tag given. The filters produce an image reduced to the essential elements, lines of the mouth, nose, eyes, and eyebrows, without extra information such as slight changes in skin-tone or grain from the reduced photo quality. presenting the HOG with a simplified but clearer image to process.

2.3 Histogram of Oriented Gradients

HOG takes the transformed image from the Gabor Filter and finds the most prominent orientation for each group of pixels, called a cell. In terms of its usefulness to Gabor, HOG calculates the gradient orientation and intensity of the Gabor image in a Histogram block. This provides a clear mathematical description of the Gabor image for classification, transforming it into a series of descriptor blocks. Figure 4 shows two outputs of HOG resulting from Figure 2b and Figure 2c It should be noted that the Histogram block generated by this process is scale invariant. The HOG process calculates the luminance gradient of each pixel before creating a histogram for each cell. The luminance gradient looks at each pixel in a cell (designated group of pixels) and calculates the direction and magnitude of the change in colour intensity using the four adjacent pixels (top, bottom, left, and right). The intensity of the pixel above is compared to the intensity of the pixel below, and the intensity of the pixel to the right is compared to the intensity of the pixel on the left. Intensity is measured from 0-255 as we are using grayscale images. Since the Gabor image is put through a minimum threshold, the extra noise created from slight abnormalities in the image is reduced, creating optimal conditions for the HOG to detect the magnitude and direction of the edges of the eyes, nose, mouth, eyebrows, and face. The process is completed by normalising the data and creating the descriptor blocks. These descriptor blocks are the combined magnitudes and directions of a group of cells and they are normalised to reduce illumination and contrast in localised areas. HOG potential is limited somewhat in unequal illumination environments. This would need to be mitigated by the preprocessing steps such as histogram adjustments, gamma correction etc. None of these were necessary here as the JAFFE database is robust in maintaining even light with only some minor noticeable issues. The HOG was applied to all images equally, with eight orientation bins, 14x14 pixels forming a single cell, and those cells organised in 8x8 formation to form a block. Output of the Gabor-HOG can be seen in Figure 4a, while HOG alone on the pre-processed image can be seen in Figure 4b. Along with the transformed image, a feature vector is also output defining the orientation bins. This feature vector containing the image descriptions, not the hog image seen in Figure 4, is the input into the feature selection and classification algorithm.

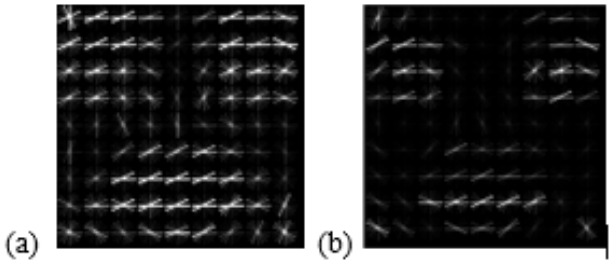


Figure 4: (a) HOG performed on a happy image that went throught preprocessing and Gabor steps, (b) HOG applied to the same happy image that just went through preprocessing steps.

 ISBN 978-0-9934207-2-6

2.4 Classification

The sklearn's Cross Validation Score (CVS) [scikit-learn developers, 2017] was used to run 50 tests across the JAFFE dataset. CVS worked in conjunction with sklearn's Pipeline [scikit-learn developers, 2017], which allowed the feature selection steps and classification processes to be ordered starting with sklearn's standard scalar to produce a properly scaled set for use by the classifier, then the result was fed into PCA. After the PCA was performed the transformed features would be classified either by SVM or NN. This allowed the standard scalar and PCA to only fit to the training data and transform the testing data separately, which is an important step when the future plans include moving to live data. This was all done using the CVS to ensure that the classifier was not over-fit to the training data [scikit-learn developers, 2017].

PCA transforms the scaled data, currently over 1000 features, to fit a new coordinate system and in doing so it reduces the features to the minimum between the following three options: the number of samples, the number of features per sample, and a fixed value if one is provided to the program [scikit-learn developers, 2017]. Since there are 213 images in total, the maximum number of components that the PCA will output is 213 if PCA was to transform the entire set at once. Our method uses 170 of the total 213 images for training, so the output from the PCA is 170 features in size. This transformed set of data is fed into the SVM and NN.

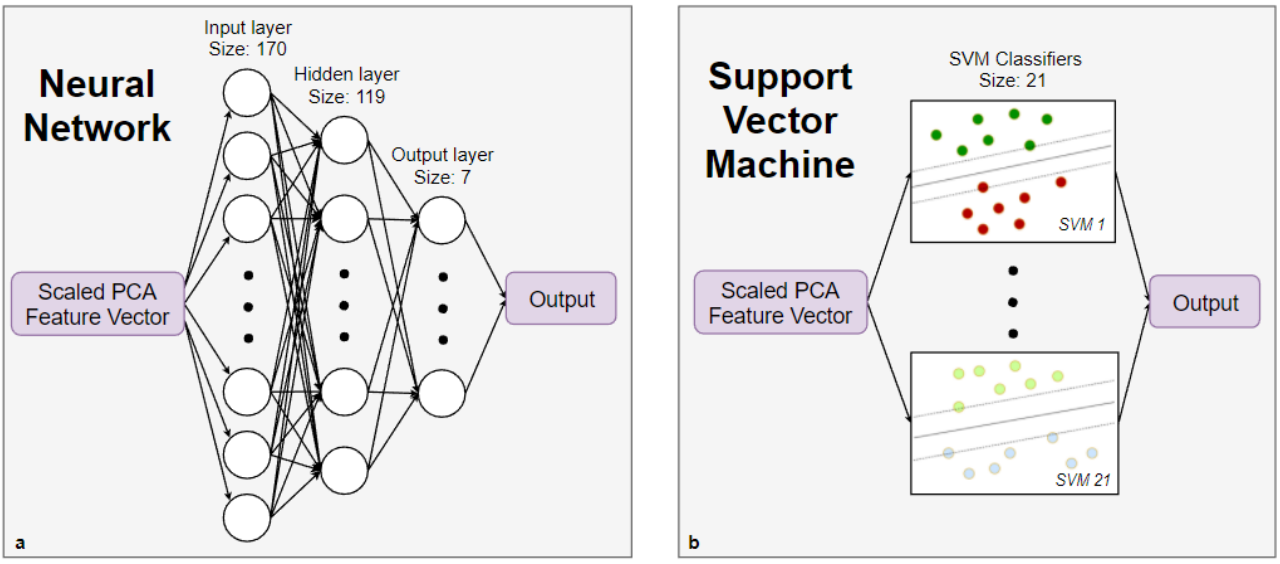


Figure 5: (a) The Hidden Layer includes 119 nodes, while the output layer has 7. The output produced by the final layer is the classified emotion. (b) The Support Vector Machine takes the Scaled PCA Feature Vector and compares the vector against each of 21 classifiers to determine which the most likely match is. The output is the classified emotion.

The NN uses three layers, including input and output, with a stochastic gradient descent activation [scikit-learn developers, 2017]. The input layer has 170 inputs, each corresponding to a single feature from the transposed PCA reduced HOG feature vector of size 170 (now a single column as opposed to the original single row). This feeds into the hidden layer of 119 neurons, with the hidden layer using a logistic activation function. The final layer is the output consisting of 7 outputs for the 7 emotions being classified. This layer uses a linear activation function, returning the final classified emotion as a binary output. The NN architecture can be seen in Figure 5a. These were found to be the optimal parameters after testing on the training data with no apparent overfitting. Training was done via the fit method as part of sklearn's NN [scikit-learn developers, 2017]. The SVM is a discriminative binary classifier using a separating hyperplane with a single input. This SVM used a polynomial kernel with a one-vs-one approach, checking the image against two classes at a time. The result is a single output determining which class the feature vector belongs to. As this is a binary classifier and 7 emotions/classes are to be classified, 21 classifiers are created to sweep through all possible combinations of emotions to generate a final outcome. For example, Happy/Sad, Happy/Angry, Happy/Neutral, Sad/Angry and

 ISBN 978-0-9934207-2-6

so on until each combination is exhausted. Again the PCA reduced (but not transposed) feature vector is fed into the SVM and checked against the 21 classifiers producing a single classified output. Figure 5b includes a diagram representative of the SVM classification. Training was done via the fit method as part of sklearn's SVM [scikit-learn developers, 2017].

3 Results

The results obtained by the combined Gabor and HOG approach over the tests achieved an accuracy 97.7% using the SVM and an accuracy of 97.7% for the NN. Compared to HOG alone, the proposed method performed 4.7% better using the SVM and 2.4% better using the NN. Preprocessing and processing time was slightly slower, as expected, however the SVM processing time was faster, while the NN processing time was about the same. When Compared with Gabor alone the proposed method performed 14.0% better using the SVM and 23.3% better using the NN. The preprocessing and processing time was also much slower than the proposed method. It was also slower in the classification time. The full results can be seen in Table 1. Training accuracies for all methods were between 98.2% and 99.4%, which is an error of 1 to 3 training image. 97.7% for testing results is representative of the misclassification of a single image, while 74.4% is representative of the misclassification of 11 images.

Table 1: Accuracies for Gabor, HOG and the Gabor/HOG method along with their processing times.

	Preprocessing & Processing Total Time	SVM Accuracy	SVM Processing Time	NN Accuracy	NN Processing Time
Gabor alone	89.31 s	83.7%	1.25 s	74.4%	1.38 s
HOG alone	54.43 s	93.0%	0.44 s	95.3%	0.81 s
Gabor/HOG	67.38 s	97.7%	0.27 s	97.7%	0.91 s

4 Discussion

For images, the Japanese Female Facial Expression (JAFFE) was used [Lyons et al., 1998]. The JAFFE database uses 10 Japanese female subjects, with approximately 3 images of the same emotion, covering the whole range of Ekman's [Ekman, 1970] universal emotion including Neutral. Each image of the JAFFE database comes with a corresponding emotion label. The database contains 31 happy images, 31 sad images, 30 angry images, 29 disgusted images, 30 surprised images, 32 fear images, and 30 neutral images, which amounts to 213 images total. Each image is a 256x256, 8-bit grayscale photograph of female Japanese faces from frontal view, all equidistance from the camera. Each maintains an even illumination with only minor differences and no notable variation in the head pose or positioning within the image. The JAFFE images are from 1992 and do contain a mild amount of film grain but the effect is mild enough that it can used without alteration to the image. Uneven shadows are present, both across a single face (shadow one side, none of the other) and across the images in total. However, we are solely looking at the effects of the Gabor-HOG method and so preprocessing is limited to cropping. The JAFFE database also includes two sets of Semantic Ratings Data. Using a scale of 1 (low) to 5 (high), each picture was rated for the six basic emotions described by Ekman [Lyons et al., 1998]. The first study used 60 female Japanese students to do the ratings, while a second Semantic rating was done using 30 female Japanese students excluding the descriptor "fear" as well as the photos labelled "fear" because the author felt that the actors did not pose fear well [Lyons et al., 1998]. While many of images were rated with a correct emotion score, some images were frequently misdiagnosed, showing that even humans find difficulty in diagnosing emotions from still images. Several other papers have used the JAFFE database as a standard alongside the Cohn-Kanade database [Chen et al., 2014], which has been excluded in this report due to the

 ISBN 978-0-9934207-2-6

labelling structure and timeframe constraints. Using HOG to detect emotion from facial expression resulted in a 94.3% average result using a linear SVM [Chen et al., 2014]. Pyramid HOG resulted in an accuracy of 86.4% on the JAFFE database [Dhall et al., 2012], and Weber Local Descriptor (WLD) plus HOG resulted in an accuracy of 94.0% [Dhall et al., 2012] These results are similar to our HOG alone method, however our Gabor-HOG method is able to outperform these methods. Similar results were seen with the use of Gabor filters, with reported results between 93.15% and 95.18% [Buciu et al., 2003] [Lajevardi and Lech, 2008]. These results outperform our Gabor alone method, however our Gabor-HOG method still achieved better accuracies. It should be noted that neither the Gabor nor the HOG variables were optimised for individual performance, but rather optimised together to create the best achievable accuracies without overfitting to the data. The small number of images in the database resulting in insufficient training of the neural network could be the cause of lower accuracy ratings. All tests were performed on the same SVM and NN with no change to their architecture. Human emotion-diagnosticians have had trouble accurately diagnosing all of the images, especially in the case of fear [Lyons et al., 1998] and the capturing of "artificial" expressions might diminish the authenticity of the emotion database in certain regards. This work only classifies base emotions and not the intensities of these emotions. Future work could investigate accurately detecting intensity of emotion. Other possibilities from this include testing on a larger still or video database.

References

[Buciu et al., 2003] Buciu, I., Kotropoulos, C., and Pitas, I. (2003). Ica and gabor representation for facial expression recognition. *IEEE*.

[Canny, 1986] Canny, J. (1986). A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8(6).

[Castrillon-Santana et al., 2007] Castrillon-Santana, M., Deniz-Suarez, O., Hernandez-Tejera, M., and Guerra-Artal, C. (2007). Encara2: Real-time detection of multiple faces at different resolutions in video streams. *Journal of Visual Communication and Image Representation*, 18(2):130–140.

[Chen et al., 2014] Chen, J., Chen, Z., Chi, Z., and Fu, H. (2014). Facial expression recognition based on facial components detection and hog features. Scientific Cooperations International Workshops on Electrical and Computer Engineering Subfields:64–69.

[Deng et al., 2005] Deng, H.-B., Jin, L.-W., Zhen, L.-X., and Huang, J.-C. (2005). New facial expression recognition method based on local gabor filter bank and pca plus lda. *International Journal of Information Technology*, 11(11):86–96.

[Dhall et al., 2012] Dhall, A., Goecke, R., Lucey, S., and Gedeon, T. (2012). Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. *IEEE*.

[Ekman, 1970] Ekman, P. (1970). Universal facial expressions of emotion. *California Mental Health Research*, 8:151–158.

[Ekman and Friesen, 1978] Ekman, P. and Friesen, W. (1978). *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press.

[Fernandes and Bala, 2017] Fernandes, S. and Bala, J. (2017). A comparative study on various state of the art face recognition techniwues under varying facial expressions. *The international Arab Journal of Information Technology*, 14(2):254–259.

[Lajevardi and Lech, 2008] Lajevardi, S. M. and Lech, M. (2008). Averaged gabor filter features for facial expression recognition. *IEEE*.

[Li and Lee, 1993] Li, C. H. and Lee, C. K. (1993). Minimum cross entropy thresholding. *Pattern Recognition*, 26(4):617–625.

[Li et al., 2013] Li, M., Bao, S., Dong, W., Wang, Y., and Su, Z. (2013). Head-shoulder based gender recognition. *IEEE*.

[Li et al., 2016] Li, X., HONG, X., Moilanen, A., Huang, X., Pfister, T., Zhao, G., and Pietikainen, M. (2016). Towards reading hidden emotions: A comparative study of spontaneous micro-expression spotting and recognition methods. *IEEE*.

[Lyons et al., 1998] Lyons, M. J., Akemastu, S., Kamachi, M., and Gyoba, J. (1998). Coding facial expressions with gabor wavelets. *IEEE International Conference on Automatic Face and Gesture Recognition*, 3:200–205.

[Marĉelja, 1980] Marĉelja, S. (1980). Mathematical description of the responses of simple cortical cells. *Journal of the Optical Society of America*, 70(11):1297–1300.

[Myers, 2000] Myers, D. G. (2000). *Psychology*. Worth Publishers, 6 edition.

[OpenCV-team, 2017] OpenCV-team (2017). Opencv.

[scikit-image developers, 2017] scikit-image developers (2017). scikit-image.

[scikit-learn developers, 2017] scikit-learn developers (2017). scikit-learn.

[Shan, 2011] Shan, C. (2011). Learning local binary patterns for gender classification on real-world face images. *Science Direct, Pattern Recognition Letters*, 33(4):431–437.

[Sobel, 2015] Sobel, I. (2015). History and definition of the so-called "sobel operator" more appropriately named the sobel-feldman operator. ResearchGate.

[Viola and Jones, 2001] Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Conference on computer vision and pattern recognition 2001*.

[Weseley and McEtarffer, 2007] Weseley, A. J. and McEtarffer, R. (2007). *AP Psychology*. Barron's Educational Series, Inc., 3 edition.

Similarity Measures and the Performance of Biometric Systems

Inas Al-Taie, Adrian Clark and Nassr Azeez

Computer Science and Electronic Engineering
University of Essex, UK

Abstract

Distance similarity measures are core components used by distance-based classification algorithms. This paper investigates whether the way in which similarity is measured can affect the performance of PCA- and LDA-based recognition systems using face, ear and palmprint biometric datasets. Four distance functions were considered: Euclidean, Manhattan, and Mahanobolis distances and a cosine similarity measure. The presence of statistically-significant performance differences was assessed using McNemar's test. It was found that all distance measures considered identified LDA as significantly outperforming PCA but that no individual similarity measure was more reliable than the others, leading to the conclusion that the content of the database used has an effect on the similarity measure.

Keywords: LDA, PCA, Biometric, Classification, Distance Similarity.

1 Introduction

Systems that require users to authenticate themselves are commonplace: credit cards have signatures and PINs, passports have photographs, and so on. Although all of these provide satisfactory proof of personal identification, they also have many drawbacks: in particular, passwords, PINs *etc* can be lost or forgotten [Shailaja and Gupta, 2006]. Hence, it has become imperative to find more effective ways to determine the identity of a person. Consequently, biometrics have become the focus of attention. The term 'biometric' refers to a measurement that can be distinguished automatically depending on the subject's physiological or behavioural characteristics and has become important in person identification in border control, where face recognition is used by some countries.

The work presented in this paper is focused on the recognition performance of three single biometric recognition systems: face, ears and palm print. Two well-known approaches are examined, Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). Researchers and developers use different ways of measuring the similarity of a probe image with images of subjects enrolled in the database. Ideally, the performance of a system should be independent of the choice of similarity measure and this work ascertains whether this is the case.

The remainder of the paper is organized as follows. Section 2 presents some background with respect to face, ear and palm print recognition and describes feature extraction based on PCA and LDA. This is followed by a discussion of similarity measures in section 3. A statistical test, McNemar's test, is used for identifying performance differences and this is described in section 4. Experimental results are reported in section 5 and the paper ends with our conclusions in section 6.

2 Background

2.1 Face, ear and hand biometrics

Face recognition: Over recent years, face recognition has become one of the most popular and successful biometrics [Sharifara et al., 2014], typically in security systems. Facial recognition can be used both to

 ISBN 978-0-9934207-2-6

identify a person (find them in a database) or verify them (confirm that they are who they claim to be); both cases involve analysing and matching patterns.

Recognition algorithms are classified into two broad approaches. The first is geometric (feature based), which tries to find distinctive features within an image and then uses these features to look for other images with similar features. The second approach is photometric (view based), a statistical approach that normalizes a gallery of face images to the same shape and saves only data that is useful for face recognition. After that, a probe image is compared with these data in order to eliminate variations [Ghimire and Lee, 2013]. In general, a face recognition system involves four steps: face detection, feature extraction feature matching, and finally recognition. Arguably, the most important step of is face detection (also known as face location) because when faces are located accurately, the recognition step becomes less complicated [Zhao et al., 2003].

Ear recognition: The ear shape is unique and permanent: the appearance of ear does not change during a human life, though its size increases with age. Additionally, it is not affected by changes of expression, unlike the face. Hence, interest in ear recognition has grown [Pflug and Busch, 2012]. In general, the idea of ear recognition is to extract from an input image a set of features and then compares this against feature sets from other images to define identity. The stages of an ear recognition system are essentially the same as those of a face recognition one [Al Mashagba, 2016] [Ziedan et al., 2016].

Palm print recognition: A palm print is an image of the palm region of the hand. It can be either an image taken by a digital scanner directly or with ink and paper and that scanned [Wu et al., 2005]. A palm print contains many line features: principal lines, wrinkles, and ridges. Because of the large surface and the rich line features, it is expected to be robust to have high individuality [Zhang et al., 2003] [Xu et al., 2015].

2.2 The recognition process

Principal component analysis Algorithm (PCA): PCA is a well-established statistical pattern recognition technique for data reduction and feature extraction [Eleyan and Demirel, 2007]. It works by identifying the principal types of variation of an input data set and then calculating an basis set that maximises variance along orthogonal directions in feature space. This decomposition is optimal in the linear least squares sense. An individual feature vector can be calculated as a weighted sum of the basis set [Pissarenko, 2002], so those weights describe a feature vector completely. In the context of face recognition, the PCA approach is usually termed *eigenfaces* [Naz et al., 2006], relying on a low-dimensional representation of face images [Sirovich and Kirby, 1987]. Faces images which are similar in overall formation will be clustered in feature space and therefore can be characterized by a low dimensional subspace.

An eigenface (basis vector) will not generally correspond to the variation of a common facial feature such as eyes, or mouth but rather to the variation present in the input to the eigen decomposition. The face recognition process will categorize a probe image containing a face as known or unknown, depending on how close the feature vector calculated from the probe image matches those of subject enrolled in the system [Singh and Kumar, 2012].

2.2.1 Linear Discriminant Analysis algorithm (LDA)

LDA, sometimes termed Fisher Discriminant Analysis (FDA), also makes use of projection into a lower dimensional feature space. The important distinction compared to PCA, which decomposes purely on the basis of variation in the input data, is that LDA also involves the class to which the data belong, meaning that more discriminating information is retained [Wagner, 2012]. Consider a situation where the significant variation is due to an external light source: the most significant eigenvector from PCA will encode this variation even though it provides no discriminating information, reducing the effectiveness of correct classification. LDA clusters the same classes tightly together, while different classes are placed as well away from each other [Martínez and Kak, 2001].

 ISBN 978-0-9934207-2-6

3 Similarity Measures

Similarity is the degree to which two features resemble each other. It is commonly calculated as a distance between the features vectors of two objects [Cha, 2007]. In this work, we have examined four common similarity measures.

Euclidean Distance: The Euclidean distance between two feature vectors **x** and **y** is simply the path length connecting them [Greenacre and Primicerio, 2008] as determined by Pythagoras's theorem:

$$\text{Eculidean Distance}(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{k}(\mathbf{x}_i - \mathbf{y}_i)^2} \tag{1}$$

For this to be meaningful, the individual components of the feature vector need to be of the same type. For example, if one component is a radius and the other an angle, the Euclidean distance is not a sensible similarity measure.

Manhattan Distance: The Manhattan distance between two points is the sum of the absolute differences of their coordinates [Cha, 2007].

$$\text{Manhatten Distance}(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{k}|\mathbf{x}_i - \mathbf{y}_i| \tag{2}$$

Again, the individual components of the feature vector need to be of a consistent type for this to make sense.

Cosine Similarity: This essentially measures the angle between a pair of feature vectors via a normalised inner product. The inner (dot) product of a vectors **x** and **y** of length n can be written as

$$\mathbf{x}.\mathbf{y} = \sum_{i=1}^{n}\mathbf{x}_i\mathbf{y}_i = |\mathbf{x}||\mathbf{y}|cos\theta$$

where $|\mathbf{x}|$ is the length of **x** *etc* and θ is the angle between $|\mathbf{x}|$ and $|\mathbf{y}|$. Re-arranging and expanding, we obtain

$$\text{sim}(\mathbf{x},\mathbf{y}) \equiv \cos\theta = \frac{\mathbf{x}.\mathbf{y}}{|\mathbf{x}||\mathbf{y}|} = \frac{\sum_{i=1}^{n}\mathbf{x}_i\mathbf{y}_i}{\sqrt{\sum_{i=1}^{n}\mathbf{x}_i^2}\sqrt{\sum_{i=1}^{n}\mathbf{y}_i^2}} \tag{3}$$

The cosine of 0 is unity, and for any other angle is less than unity. Two vectors with the same orientation have a cosine similarity of 1, two perpendicular vectors have a similarity of 0, and two opposed vectors have a similarity of -1 [Bora et al., 2014]. Again, the elements of **x** and **y** really need to be measured in a consistent framework for this to work correctly.

Mahalanobis Distance: The Mahalanobis distance between two vectors **x** and **y** can be calculated from the expression:

$$\text{Mahalanobis Distance}(\mathbf{x},\mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})^T S^{-1}(\mathbf{x}-\mathbf{y})} \tag{4}$$

where S is the covariance matrix [De Maesschalck et al., 2000]. As this is scaled in terms of the covariance matrix, one can argue that the elements of **x** and **y** do not necessarily have to be of the same type.

 ISBN 978-0-9934207-2-6

Figure 1: The output of face recognition using PCA and LDA. (a) shows probe images and (b) the closest match in the database using Euclidean distance.

4 McNemar's test

McNemar's test is a statistical test that can be applied to a pair of algorithms to explore where one is more effective than the other [Kanwal, 2013]. For each test, one identifies the outcome, success or failure, reported by the two algorithms; because each individual test forces a decision to be made into two possible outcomes, the underlying statistical distribution is binomial. One counts the number of cases in which the first algorithm succeeds and the second fails, N_{sf}, and *vice versa*, N_{fs} — it is only those cases in which the algorithms perform differently that are of interest. One then calculates the so-called Z-score using:

$$Z = \frac{|N_{sf} - N_{fs}| - 1}{\sqrt{N_{sf} + N_{fs}}} \tag{5}$$

where the -1 is a continuity correction. Clearly, if the performances of the two algorithms being considered are identical, $Z = 0$, and its value increases as the number of discrepancies in performance increase. Confidence limits can be associated with the value of Z, and the one most commonly used is 1.96, which indicates that there is a probability of 0.05 (*i.e.*, one in twenty) that the results obtained could be an artefact of the data used [Yimyam and Clark, 2012]. As a rule of thumb, when $N_{sf} + N_{fs} > 20$, the test is reliable.

5 Experimental Work

Face database: Frontal images from the Caltech Faces Database were used [Grgic and Delac, 2013] in this work. It contains 10–21 different images of each of 27 distinct subjects. For some subjects, the images were taken at different times, with varying lighting, facial expressions (smiling or not smiling, open or closed eyes, *etc*) and facial details (with or without glasses). Further images have been generated from the original images in the database by changing lightness and using various filters. All the images were taken in an upright, frontal position. Image alignment — cropping, rotating, scaling — was performed for all images in the database. Some 433 labelled face images were used for training and 100 for testing. Figure 1 illustrates a face recognition system based on PCA and LDA respectively.

Ear database: The IIT Delhi Ear Database has been used in this work [Kumar and Wu, 2012]. The database consists of 497 images of 125 different subjects. For each individual, 4 or 5 images were taken in an indoor

 ISBN 978-0-9934207-2-6

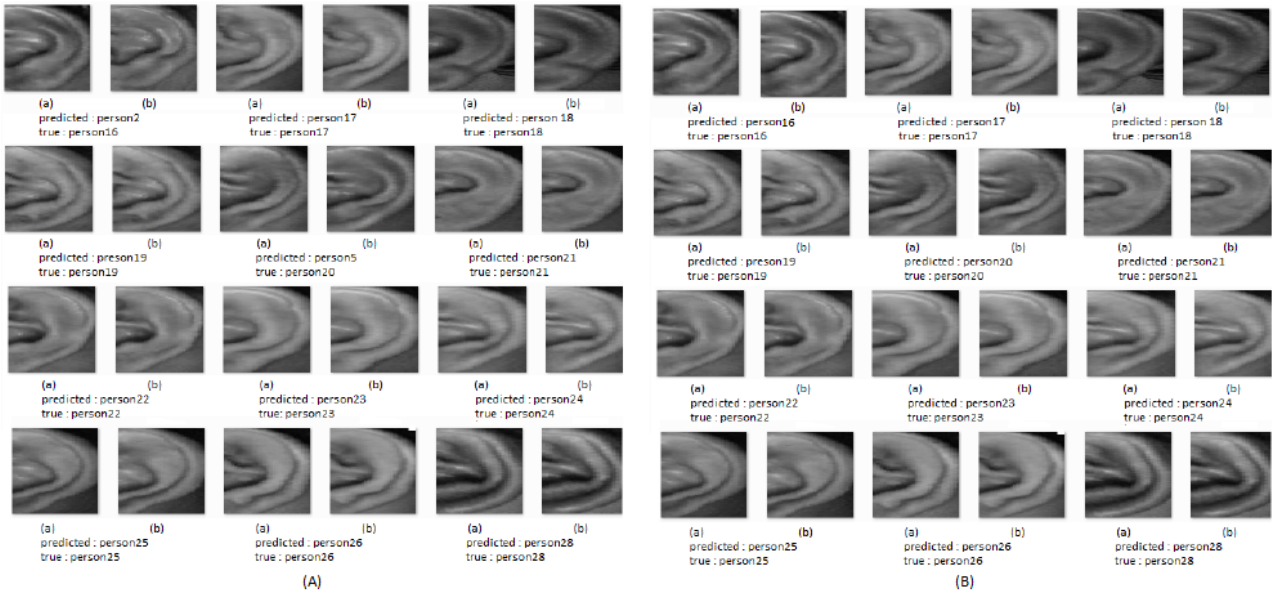


Figure 2: Ear recognition using PCA. (a) shows probe images and (b) the closest match in the database using Euclidean distance.

Figure 3: Palm recognition using PCA. (a) shows probe images and (b) the closest match in the database using Euclidean distance.

 ISBN 978-0-9934207-2-6

LDA algorithm	PCA algorithm							
Biometric	Euclidean		Manhattan		Mahalanobis		Cosine	
	N_{sf}	N_{fs}	N_{sf}	N_{fs}	N_{sf}	N_{fs}	N_{sf}	N_{fs}
Face	37	**1**	19	**0**	58	**0**	37	**3**
Ear	59	**0**	12	**3**	46	**4**	41	**1**
Palm	28	**4**	22	**4**	80	**0**	36	**0**

Table 1: Counts of the discrepancies between algorithm outputs using the various distance measures

LDA algorithm	PCA algorithm			
Biometric	Euclidean	Manhattan	Mahalanobis	Cosine
Face	5.678	4.130	7.485	5.218
Ear	7.551	2.066	5.798	6.018
Palm	4.066	3.333	8.832	5.833

Table 2: Z-values between LDA and PCA using the various distance measures

environment. The resolution of these images is 492 × 702 pixels and all these images are stored in JPEG format. Further images have been generated from the original images within the database by changing lightness and using various filters. The database was split into 397 training images and 100 testing images. Figure 2 shows ear recognition system based on PCA and LDA respectively.

Palm database: The palm database consists of 288 images with 50 subjects. All the images were collected in an indoor environment and all the subjects were aged 12–57 years. The resolution of these images is 800 pixels. Further images have been generated from the original one by changing lightness and using various filters. Some 288 images were used for training and 100 for testing. Figure 3 shows a palm recognition system based on PCA and LDA respectively [Kumar, 2008, Kumar and Shekhar, 2011].

The results found when running PCA and LDA on face, ear, and palm datasets with the distance measures discussed earlier are presented in Table 1, while Table 2 and Figure 4 show Z-values between PCA and LDA using the various distance measures. In all cases considered $Z > 1.96$, showing that LDA outperformed PCA by a statistically significant amount on all the datasets considered at the 5% level, and did so irrespective of the distance measure employed. In a sense, this is reassuring, especially as McNemar's test is regard as being statistically 'weak.' In the context of evaluating algorithms, a weak test is attractive in that it requires larger body of supporting evidence, meaning that, given the natural variability of imagery, there must be a good number of occasions on which one algorithm out-performs the other.

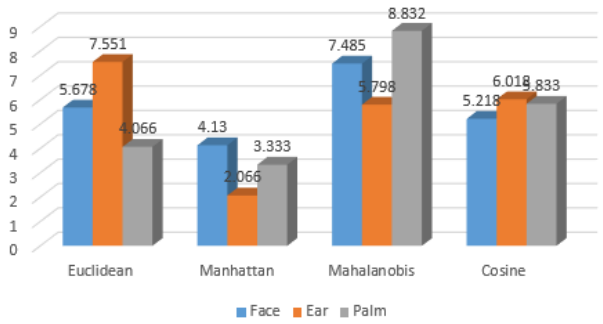


Figure 4: Z-values between LDA and PCA techniques

 ISBN 978-0-9934207-2-6

Looking along the rows of Table 2, one can compare how effective the various measures are at identifying these performance differences. The most clear conclusion here is that the Z values from the Manhattan distance are consistently lower than those of the other measures. Conversely, the Mahanobolis distance yields the largest Z in two of the three cases considered, but the second-lowest in the third case. Indeed, it is easy to conceive a situation in which some of the measures indicate that there is a significant performance difference while others do not.

We are forced to conclude that none of the distance measures considered is consistently best at identifying performance differences. What is not clear at this juncture is why the Mahanobolis distance is less effective at identifying performance differences in the case of the ear database — or perhaps why the other measures exaggerate performance differences. Clearly, this will depend on the content of the databases.

6 Conclusions

Four similarity measures have been compared for PCA- and LDA-based face, ear and palm biometrics. Classification was performed using a nearest neighbour classier using Manhattan, Euclidean, cosine similarity and Mahalanobis distance measures. The experimental results show that both PCA and LDA perform well if presented with an image in the test set which is similar to an image in the training set. LDA shows a significantly better recognition performance in all cases as evidenced by McNemar's test, suggesting that it is better at handling variation in lighting and expression. However, the fact that the magnitudes of the similarity measures were not consistently rank-ordered shows that the choice of similarity measure can affect the conclusions drawn. There is clearly some feature or property of the content of datasets that affect the similarity measures. We are exploring ways in which this might be determined.

References

[Al Mashagba, 2016] Al Mashagba, E. F. (2016). Human identification based on geometric feature extraction using a number of biometric systems available: Review. *Computer and Information Science*, 9(2):140.

[Bora et al., 2014] Bora, M., Jyoti, D., Gupta, D., and Kumar, A. (2014). Effect of different distance measures on the performance of k-means algorithm: An experimental study in matlab. *arXiv preprint arXiv:1405.7471*.

[Cha, 2007] Cha, S.-H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *City*, 1(2):1.

[De Maesschalck et al., 2000] De Maesschalck, R., Jouan-Rimbaud, D., and Massart, D. L. (2000). The mahalanobis distance. *Chemometrics and intelligent laboratory systems*, 50(1):1–18.

[Eleyan and Demirel, 2007] Eleyan, A. and Demirel, H. (2007). *Pca and lda based neural networks for human face recognition*. INTECH Open Access Publisher.

[Ghimire and Lee, 2013] Ghimire, D. and Lee, J. (2013). Geometric feature-based facial expression recognition in image sequences using multi-class adaboost and support vector machines. *Sensors*, 13(6):7714–7734.

[Greenacre and Primicerio, 2008] Greenacre, M. and Primicerio, R. (2008). Measures of distance between samples: Euclidean. *Fundacion BBVA Publication (December 2013). ISBN*, pages 978–84.

[Grgic and Delac, 2013] Grgic, M. and Delac, K. (2013). Face recognition homepage. *Zagreb, Croatia (www. face-rec. org/databases)*, 324.

[Kanwal, 2013] Kanwal, N. (2013). *Low-level image features and navigation systems for visually impaired people*. PhD thesis, University of Essex.

[Kumar, 2008] Kumar, A. (2008). Incorporating cohort information for reliable palmprint authentication. In *Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on*, pages 583–590. IEEE.

[Kumar and Shekhar, 2011] Kumar, A. and Shekhar, S. (2011). Personal identification using multibiometrics rank-level fusion. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(5):743–752.

[Kumar and Wu, 2012] Kumar, A. and Wu, C. (2012). Automated human identification using ear imaging. *Pattern Recognition*, 45(3):956–968.

[Martínez and Kak, 2001] Martínez, A. M. and Kak, A. C. (2001). Pca versus lda. *IEEE transactions on pattern analysis and machine intelligence*, 23(2):228–233.

[Naz et al., 2006] Naz, E., Farooq, U., and Naz, T. (2006). Analysis of principal component analysis-based and fisher discriminant analysis-based face recognition algorithms. In *Emerging Technologies, 2006. ICET'06. International Conference on*, pages 121–127. IEEE.

[Pflug and Busch, 2012] Pflug, A. and Busch, C. (2012). Ear biometrics: a survey of detection, feature extraction and recognition methods. *Biometrics, IET*, 1(2):114–129.

[Pissarenko, 2002] Pissarenko, D. (2002). Eigenface-based facial recognition. *December 1st*.

[Shailaja and Gupta, 2006] Shailaja, D. and Gupta, P. (2006). A simple geometric approach for ear recognition. In *Information Technology, 2006. ICIT'06. 9th International Conference on*, pages 164–167. IEEE.

[Sharifara et al., 2014] Sharifara, A., Rahim, M., Shafry, M., and Anisi, Y. (2014). A general review of human face detection including a study of neural networks and haar feature-based cascade classifier in face detection. In *Biometrics and Security Technologies (ISBAST), 2014 International Symposium on*, pages 73–78. IEEE.

[Singh and Kumar, 2012] Singh, A. and Kumar, S. (2012). *Face Recognition using PCA and Eigen face approach*. PhD thesis, National Institute of Technology Rourkela, India.

[Sirovich and Kirby, 1987] Sirovich, L. and Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A*, 4:519–524.

[Wagner, 2012] Wagner, P. (2012). Face recognition with python.

[Wu et al., 2005] Wu, X., Wang, K., and Zhang, D. (2005). Palmprint authentication based on orientation code matching. In *Audio-and Video-Based Biometric Person Authentication*, pages 83–132. Springer.

[Xu et al., 2015] Xu, Y., Fei, L., and Zhang, D. (2015). Combining left and right palmprint images for more accurate personal identification. *IEEE transactions on image processing*, 24(2):549–559.

[Yimyam and Clark, 2012] Yimyam, P. and Clark, A. F. (2012). Agricultural produce grading by computer vision using genetic programming. In *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on*, pages 458–463. IEEE.

[Zhang et al., 2003] Zhang, D., Kong, W.-K., You, J., and Wong, M. (2003). Online palmprint identification. *IEEE Transactions on pattern analysis and machine intelligence*, 25(9):1041–1050.

[Zhao et al., 2003] Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. (2003). Face recognition: A literature survey. *Acm Computing Surveys (CSUR)*, 35(4):399–458.

[Ziedan et al., 2016] Ziedan, I. E., Mohamed, S., and Farouk, H. (2016). Comparitive study on human identification using ear biometrics. *International Journal of Computer Science and Information Security*, 14(11):1040.

Classification of Alzheimer's disease subjects from MRI using the principle of consensus segmentation

Aymen Khlif and Max Mignotte

Department of Computer Science and Operations Research
University of Montreal, Canada
{khlifaym,mignotte}@iro.umontreal.ca

Abstract

In thiss paper, we develop an original and reliable detection and classification framework for Alzheimer's Disease (AD) in structural Magnetic Resonance Images (MRI). This work exploits recent advances made in segmentation and multimedia indexing and classification for Content Based Visual Information Retrieval (CBVIR). More precisely, It exploits the concept of consensus segmentation to define two segmentation prototypes (Normal Control and Alzheimer's disease) of the brain in terms of cerebral Grey Matter (GM), White Matter (WM) and Cerebro-Spinal Fluid (CSF). The classification is then performed by computing the Pott distance of each image w.r.t the two prototypes. Based on a threshold on the computed distance, brain images are classified using either Minimal Distance (MD) or K-Nearest-Neighbors (KNN) classifier. Our approach has been evaluated on the baseline MR images of 98 subjects from the Open Access Series of Imaging Studies (OASIS) database. The used subset consists of 49 subjects who have been diagnosed with very mild to mild AD and 49 non-demented individuals. The experimental results show that our classification of patients with AD versus NC subjects achieves accuracy of 86%. Results demonstrate very promising classification performance while being simple compared to the state-of-the-art classical volumetric AD diagnosis methods.

Keywords: Consensus segmentation, Alzheimer's disease, MRI, Image classification, Machine learning techniques.

1 Introduction

Alzheimer's disease (AD) is an irreversible neurodegenerative dementia that occurs most frequently in older adults and that gradually destroys regions of the brain that are responsible for memory, learning, thinking, and behavior [Papakostas et al., 2015]. Current estimates indicate that 5.3 million Americans of all ages are afflicted with this illness and this number is expected to increase to 16 million people by 2050, unless a cure is found. The socio-economic consequences of this increase are cumbersome and makes early diagnosis of AD a public health emergency.

Medical information from structural Magnetic Resonance Imaging (sMRI) has long time been the most used neuroimaging modality to detect brain atrophy in AD studies. In fact, two main families of methods can be distinguished to extract features from MRI for AD classification. Several studies report the use of sophisticated measurement techniques that assess anatomical changes in areas compromised by AD such as the Hippocampal Volume (HV), the Lateral Ventricles Volume (LVV), CSF Volume (CSVV), etc., These (so-called volumetric) methods are only based on form, size and/or shape derived features extracted from the brain structures.

Aside from volumetric approaches, morphometric methods have gained great interest among which we can distinguish: Voxel Based Morphometry (VBM) [Ashburner and Friston, 2000] which is a widely used whole-brain analysis method, which allows an exploration of the differences in local concentrations of grey matter and

white matter. Tensor Based Morphometry (TBM) [Wolz et al., 2011] was proposed to identify local structural changes from the gradients of deformations fields. Object Based Morphometry (OBM) [Mangin et al., 2003] was introduced to perform shape analysis of anatomical structures and recently, Features Based Morphometry (FBM) [Toews et al., 2010] was proposed as a method for relevant brain features comparison using a probabilistic model on local image features in scale-space [Ahmed et al., 2015]. Voxel based methods work directly on the voxel grid and are computationally very efficient. An advantage of these approaches, compared to the ROI-based volumetric methods, is the fact that they do not require *a priori* assumptions about the location, the size or number of ROIs to be analyzed, since they provide voxel wise measures determined in the entire brain. More, there is no evidence that other regions (except hippocampus and entorhinal cortex) did not provide any information for AD and NC [Zhang et al, 2015]. Recent studies on AD diagnosis found that the quantity of CSF is a biomarker of AD [Shaw et al., 2009]. Indeed, smaller hippocampal volume is associated with greater CSF amount [Ott et al., 2010].

With the features estimated, either by volumetric or voxel-based methods, on training cases, a classifier can be trained and applied to predict the diagnosis of a testing case, whose features are extracted in the same way. Among the most popular classifiers: Linear Discriminant Analysis (LDA) [French et al., 1997], Neural Network (NN) [Savio et al., 2009], Support Vector Machines (SVM) either with linear or non-linear [Magnin et al., 2009, Mesrob et al., 2008, Oliveira et al., 2010, Savio et al., 2011], K-Nearest-Neighbors (KNN) [El-Dahshan et al., 2010] and Bayes classifier.

Many of the classification studies on the detection of AD were done with both men and women. However, it has been demonstrated that brains of women are different from men to the extent that it is possible to discriminate the gender *via* MRI analysis. Moreover, it has been shown that VBM is sensitive to the gender differences [Lao et al, 2004]. For these reasons, we have been very cautious in this study; therefore, as proposed in [Papakostas et al., 2015, Savio et al, 2011, Savio et al, 2009, Savio et al., 2009], we have selected a set of 98 MRI women's brain volumes. It must be noted that this is a large number of subjects compared with the other studies referred above as it has also been mentioned in [Savio et al, 2009].

In this work we propose an automatic content analysis based framework for recognition of AD using MRI scans. Image content analysis and classification methodologies are now more and more used for medical information mining and retrieval [Muller and Deserno, 2010, Kumar et al., 2013] with the aim of Computer-Aided Diagnosis (CAD). The proposed method exploits recent advances made in segmentation and multimedia indexing and classification for Content Based Visual Information Retrieval (CBVIR) and, more precisely, the concept of consensus segmentation to define two segmentation prototypes (prototype_AD and prototype_NC) of the brain in terms of cerebral grey matter, white matter, and cerebro-spinal fluid. This two prototypes are then cleverly combined with the KNN algorithm to increase the classification results.

2 Experimental study

In order to investigate the detection performance of the proposed MDC-kNN (Minimum Consensus Distance-KNN) classifier, a set of appropriate experiments were conducted. For the experimental purposes, specific software was developed in C++. All experiments were executed in an Intel i7 3.3 GHz PC with 16 GB RAM.

2.1 Dataset description

The data analyzed in this paper was obtained from the OASIS database[1]. As proposed in [Papakostas et al., 2015, Savio et al, 2011, Savio et al, 2009, Savio et al., 2009], a subset of the OASIS database [Marcus et al., 2007] was selected in order to evaluate the detection performance of the proposed classifier. This two-class dataset has across-sectional collection of 416 subjects covering the adult lifespan aged $(18-96)$ including individuals with early-stage Alzheimer's disease. A subset of the original OASIS dataset including 98 right-handed women (aged $65-96$ years) is considered herein. More precisely, the used subset consists of 49 subjects who have been

[1] http://www.oasis-brains.org

 ISBN 978-0-9934207-2-6

diagnosed with very mild to mild AD (class 1) and 49 non-demented (class 2). The designation of demented and non-demented is based on the Clinical Dementia Rate (CDR). A CDR of 0 corresponds to the normal cognitive state, CDR > 0 to dement. The demographic information of these subjects is summarized in Table 1.

	Very mild to mild AD	**Normal Controls**
No. of subjects	49	49
Age	78.08 (66-96)	77.77 (65-94)
Education	2.63 (1-5)	2.87 (1-5)
Socioeconomic status	2.94 (1-5)	2.88 (1-5)
CDR (0.5/1/2)	31/17/1	0
MMSE	24 (15-30)	28.96 (26-30)

Table 1: Demographic information of the subjects in the two classification classes

2.2 MRI data preprocessing

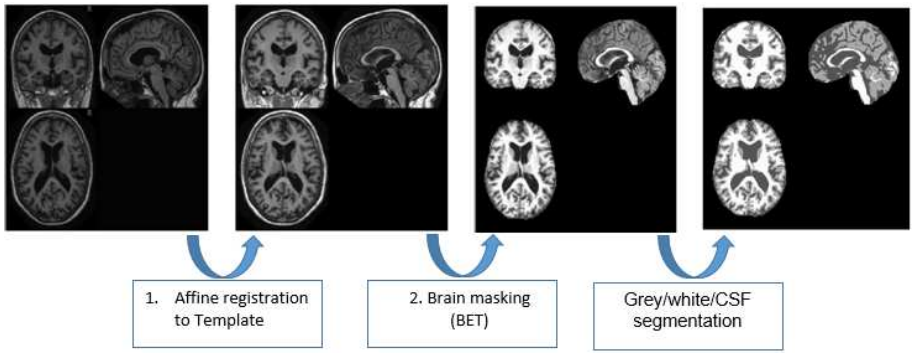


Figure 1: Snapshot of a specific subject. (a) One original scan. (b) Atlas-registered image. (c) Brain masked version of Figure 1(b). (d) The GM/WM/CSF segmentation image.

Before any further analysis, the images must be pre-processed to eliminate variations due to the different MR acquisitions. The preprocessing steps are summarized in Figure 1 for the T1 image of one subject.

All (axial section) MRI images were: (a) corrected for inter-scan head movement [Marcus et al., 2007] and rigidly aligned to the Talairach and Tournoux space [Talairach, 1988], (b) transformed to a template with a 12-parameter affine registration an merged into a 1-mm isotrpic image [Marcus et al., 2007], (c) skull-stripped and corrected for intensity inhomogenity [Smith, 2002], and finally (d) segmented into $K = 3$ classes corresponding to the three existing cerebral tissues, respectively; "cerebrospinal fluid", "white matter" and "grey matter". This 3-class segmentation is obtained by applying firstly, 10 times the K-Means clustering algorithm with different seeds and different number of neighbors of the pixel to be classified and secondly by combining them with a simple majority vote scheme[2]. In our application, the majority vote is achieved with a spatial window (of size 3×3 pixels and centered on the pixel s to be classified) that collects the class labels of the 10 segmentation results obtained by each K-mean clustering and by finally assigning to that central pixel s, the class label that has the majority vote. This strategy ensures both an efficient spatial regularization of the final segmentation result and also a reliable decision fusion between results obtained by these K-mean clusterings. In this segmentation, the "CSF", the "white matter" and the "grey matter" are represented by a dark, a grey, and a white region respectively, in order to visually express the activity level of the blood flow.

[2]We have noticed that these segmentation technique provides better classification results than a SPM software (Statistical Parametric Mapping) based segmentation technique (the classification accuracy is better by 2%).

 ISBN 978-0-9934207-2-6

2.3 Prototypes NC and AD

From the above-mentioned segmentations (416 subjects), two segmentation prototypes (prototype_AD and prototype_NC), in terms of cerebral GM, WM and CSF are built by combining all segmentations related to subjects who have been diagnosed with very mild to mild AD (class 1 which contains 100 subjects) and those related to the non-demented (class 2 which contains 316 subjects) with the majority vote scheme. The processing is illustrated in Figure 2, whereas the overall procedure is explained in the next section.

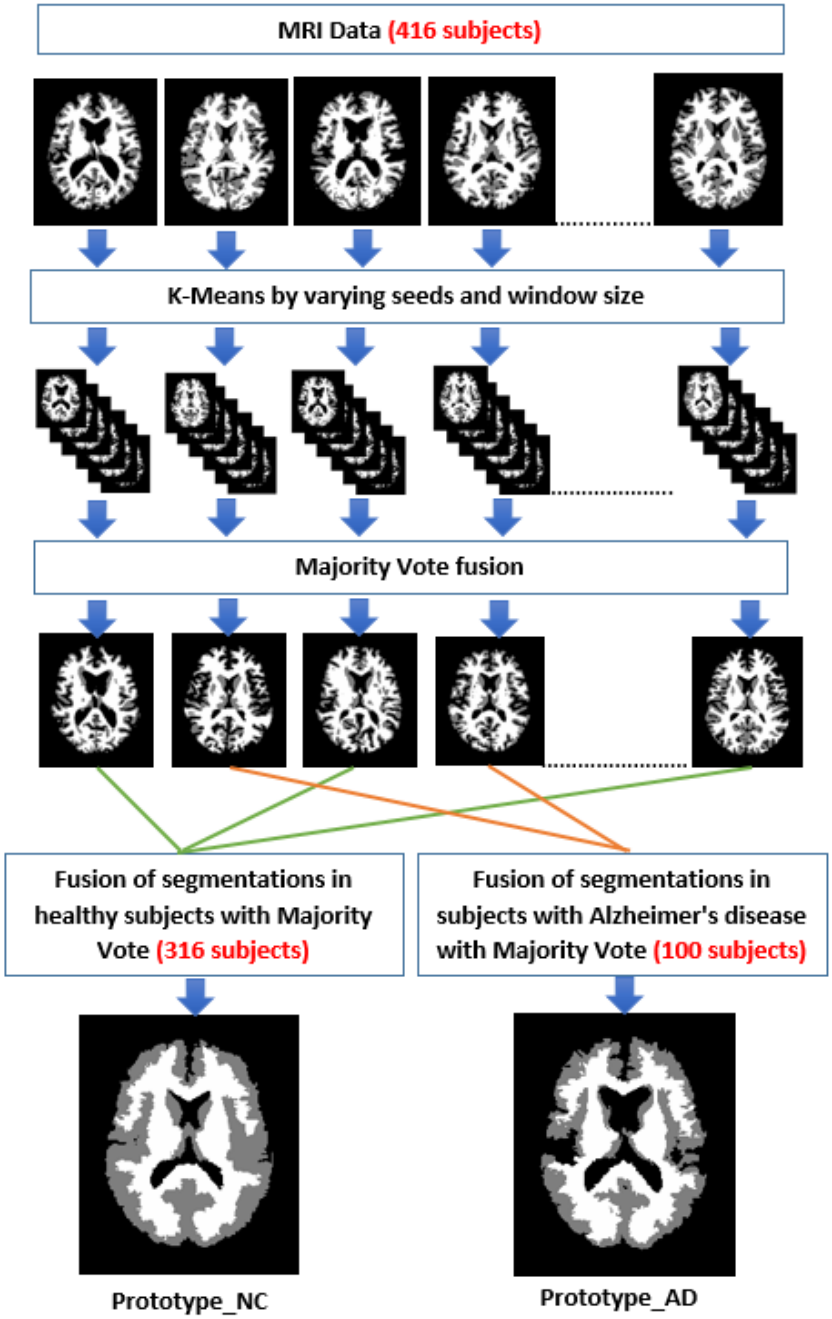


Figure 2: Methodology for the creation of NC and AD prototypes

2.4 Hybrid classification

The proposed hybrid classification technique named Minimum Consensus Distance-KNN (MDC-KNN), combining the previously estimated prototypes (Prototype_NC and Prototype_AD) and the KNN algorithm (weighted

 ISBN 978-0-9934207-2-6

as proposed in [Bailey and jain, 1978] with the Pott distance in our case) consists of three stages:

1. Calculate the two Pott distances (the normalized number of differences in percentage[3]) between the segmentation related to an input MRI image and respectively the Prototype_NC (D_NC) and the Prototype_AD (D_AD).
2. Choose the classifier KNN or (prototype-based) Minimal distance (MD). The choice of classifier (KNN or MD) depends on the difference between D_NC and D_AD. If the difference is large (greater than a threshold T which was set to 1.5%[4]) then, in this case (case 1 or 2 of Figure 4), we choose the MD classifier since in this case we are certain that the segmentation is very close to one of the two prototypes and consequently we can thus rely on this classification procedure. Otherwise, we apply the KNN classifier (case 3 of Figure 4) in with the Pott distance between the segmentation related to an input MRI image and the 3-class segmentations obtained for each image of the test set.
3. Classification.

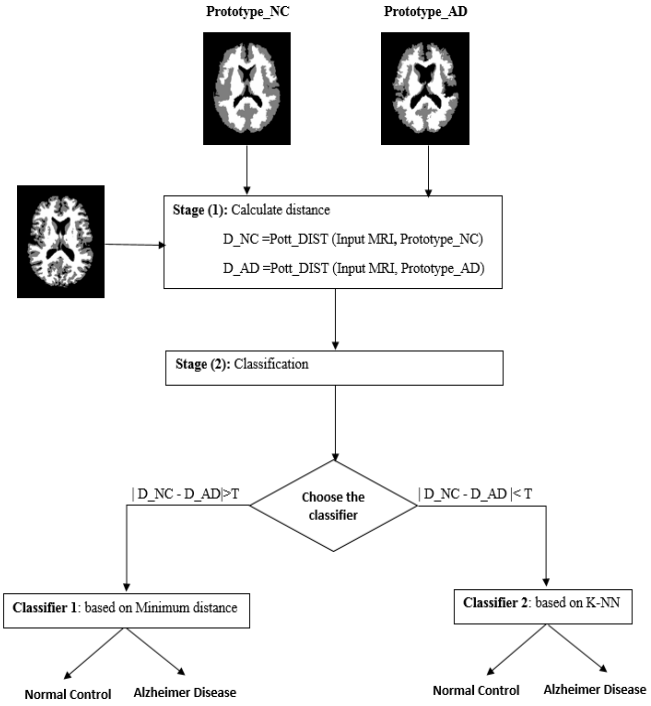


Figure 3: The methodology of the proposed method.

The schematic diagram for proposed methodology is shown in Figure 3.

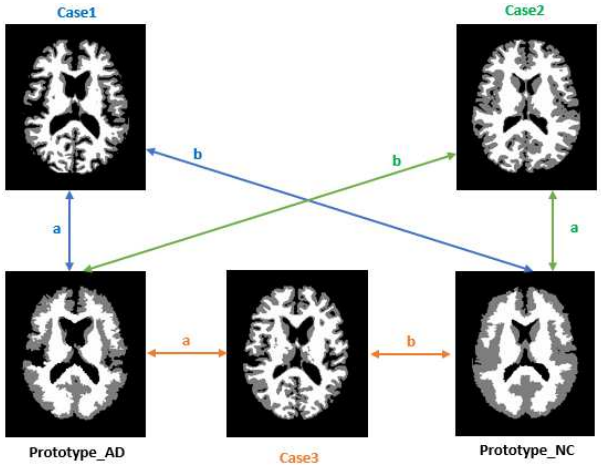


Figure 4: $|D\_NC - D\_AD|$ large (case 1 or 2) or very low (case 3)

[3] $Pott\_Dist(S1, S2) = \frac{\sum_{i=1}^{N} 1-\delta_{x_i^1 x_i^2}}{N}$
Where δ is the Kronecker delta function and the summation is over all the N pixels of the segmented image and x_i^1, x_i^2 is respectively the label of the first and second segmentation at pixel i.

[4] We have computed than the distance (in percentage of pixel difference) between Prototype_NC & Prototype_AD is 15%. Based on this, we have set this threshold T to an order of magnitude ($\div 10$) lower (i.e., $T = 1.5\%$).

 ISBN 978-0-9934207-2-6

3 Experimental results

We evaluate the performance of the proposed method in terms of sensitivity=TP/(TP + FN), specificity=TN/(TN + FP) and accuracy=(TN + TP)/(TN+TP+FN+FP). Where True Positives (TP) are AD patients correctly identified as AD, True Negatives (TN) are controls correctly classified as controls, False Negatives (FN) are AD patients incorrectly identified as controls and False Positives (FP) are controls incorrectly identified as AD. Sensitivity is the proportion of AD subjects correctly classified, and the specificity is the proportion of correctly classified controls.

The actual feature datasets (98 subjects: 49 AD and 49 NC) have been used in several works in the literature, hence results obtained with a variety of classifier models are publicly available for comparison. The AD detection statistics, accuracy, sensitivity and specificity, of the proposed MDC-KNN classifier by performing a leave-one-out cross validation test, in comparison with the state-of-the-art models [Savio et al, 2011, Chyzhyk et al, 2012, Papakostas et al., 2015] are summarized in Table 2.

By examining the results of Table 2, it follows that the proposed classifier MDC-KNN with $K = 3$ (This number was empirically tested) demonstrated superior performance than some conventional classifiers such as RBF, MLP-BP, PNN, Linear SVM (8% higher accuracy) and some advanced classification models (Indep-linear-SVM, Indep-rbf-SVM, linear-AB-SVM, rbf-AB-SVM, Kernel-LICA-DC, LVQ1, LVQ2, rbf-DAB-SVM). Let us note that the single KNN classifier (witout being combined with our prototype based MD classifier) reaches a classification value of 73% (See in Table 2 KNN-Pott-MV).

Classifier type	Accuracy	Sensitivity	Specificity
MDC-KNN (our approach)	0.86	0.85	0.87
rbf-DAB-SVM	0.85	0.78	0.92
LVQ2	0.83	0.74	0.92
LVQ1	0.81	0.72	0.90
LC-KNN	0.80	0.80	0.79
rbf-AB-SVM	0.79	0.78	0.80
MLP-BP	0.78	0.69	0.88
PNN	0.78	0.62	0.94
Linear SVM	0.78	0.72	0.88
Indep-rbf-SVM	0.75	0.56	0.95
Kernel-LICA-DC	0.74	0.96	0.52
Indep-Linear-SVM	0.74	0.51	0.97
KNN-Pott-MV	0.73	0.61	0.85
Linear-AB-SVM	0.71	0.54	0.88
RBF	0.66	0.65	0.68

Table 2: AD detection statistics of several classifiers (using a leave-one-out cross validation test and the same dataset).

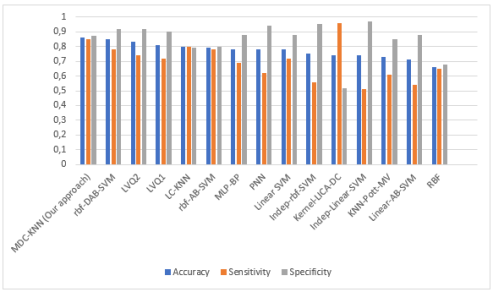


Figure 5: Classification results for the different methods

4 Conclusion and future works

In this paper, we presented a novel classification framework, based on prototype defined using consensus segmentation, is proposed for Alzheimer's disease. The proposed framework learns prototypes for Normal and AD class by first segmenting each image into CSF, WM, and GM (tissue classes) using K-means with 10 different seeds/number-of-neighbors, followed by a majority vote. All the segmented images (for both AD and NC) are then combined using majority vote to generate the prototypes. The classification is then performed by computing the Pott distance of each image w.r.t the two prototypes. Based on a threshold on this distance, brain images are then classified using either the MD or KNN classifier. Experiments are conducted using 98 subjects and compared with various classifiers.The aforementioned experimental results, in particular through the high performance of the proposed classifier in terms of accuracy, have demonstrated the remarkable potential of pro-

 ISBN 978-0-9934207-2-6

totypes (prototype_AD and prototype_NC) in classification applications. A possible extensions to the current work is the use of another more advanced fusion algorithm for the creation of prototypes. The strength of the proposed work consists in the following:

- Our approach is automatic and does not require the intervention of the clinician during the disease diagnosis.
- It is extensible to other diseases that can be diagnosed by brain MRI such as Schizophrenia and brain tumors.
- The method could be extended by combining axial, coronal, and sagittal MRI data for improving the classification accuracy.

References

[Ahmed et al., 2015] Ahmed, O., Benois-Pineau, J., Allard, M., Amar, C. B., and Catheline, G. (2015). Classification of alzheimer's disease subjects from MRI using hippocampal visual features. *Multimedia Tools and Applications*, 74(4):1249–1266.

[Ashburner and Friston, 2000] Ashburner, J. and Friston, K. (2000). Voxel-based morphometry-the methods. *Neuroimage*, 11(6):805–821.

[Bailey and jain, 1978] Bailey, T., Jain, A. K. (1978). A note on distance-weighted *k*-nearest neighbor rules. *IEEE Transactions on Systems, Man, and Cybernetics*, (4):311–313.

[Chyzhyk et al, 2012] Chyzhyk, D., Grana, M., Savio, A., Maiora, J. (2012). Hybrid dendritic computing with kernel-lica applied to alzheimer's disease detection in MRI. *Neurocomputing*, 75(1):72–77.

[El-Dahshan et al., 2010] El-Dahshan, E., Hosny, T., Salem, A. (2010). Hybrid intelligent techniques for MRI brain images classification. *Digital Signal Processing*, 20(2):433–441.

[French et al., 1997] French, B. M., Dawson, M. R., and Dobbs, A. R. (1997). Classification and staging of dementia of the alzheimer type: a comparison between neural networks and linear discriminant analysis. *Archives of neurology*, 54(8):1001–1009.

[Oliveira et al., 2010] Oliveira Jr, P. P. D. M., Nitrini, R., Busatto, G., Buchpiguel, C., Sato, J. R., Amaro Jr, E. (2010). Use of SVM methods with surface-based cortical and volumetric subcortical measurements to detect alzheimer's disease. *Journal of Alzheimer's Disease*, 19(4):1263–1272.

[Kumar et al., 2013] Kumar, A., Kim, J., Cai, W., Fulham, M., and Feng, D. (2013). Content-based medical image retrieval: a survey of applications to multidimensional and multimodality data. *Journal of digital imaging*, 26(6):1025–1039.

[Lao et al, 2004] Lao, Z., Shen, D., Xue, Z., Karacali, B., Resnick, S. M., Davatzikos, C. (2004). Morphological classification of brains via high-dimensional shape transformations and machine learning methods. *Neuroimage*, 21(1):46–57.

[Magnin et al., 2009] Magnin, B., Mesrob, L., Kinkingnéhun, S., Pélégrini-Issac, M., Colliot, O., Sarazin, M., and Benali, H. (2009). Support vector machine-based classification of alzheimer's disease from whole-brain anatomical MRI. *Neuroradiology*, 51(2):73–83.

[Mangin et al., 2003] Mangin, J., Rivière, D., Cachia, A., Papadopoulos-Orfanos, D., Collins, D., Evans, A., and Régis, J. (2003). Object-based strategy for morphometry of the cerebral cortex. In *Biennial International Conference on Information Processing in Medical Imaging*, pages 160–171. IPMI.

 ISBN 978-0-9934207-2-6

[Marcus et al., 2007] Marcus, D., Wang, T., Parker, J., Csernansky, J., Morris, J., and Buckner, R. (2007). Open Access Series of Imaging Studies (OASIS): cross-sectional MRI data in young, middle aged, nondemented, and demented older adults. *Journal of cognitive neuroscience*, 19(9):1498–1507.

[Mesrob et al., 2008] Mesrob, L., , B., Colliot, O., Sarazin, M., Hahn-Barma, V., Dubois, B., and Benali, H. (2008). Identification of atrophy patterns in alzheimer's disease based on SVM feature selection and anatomical parcellation. In *International Workshop on Medical Imaging and Virtual Reality*, pages 124–132.

[Muller and Deserno, 2010] Muller, H. and Deserno, T. M. (2010). Content-based medical image retrieval. *Biomedical Image Processing*, pages 471–494.

[Ott et al., 2010] Ott, B. R., Cohen, R. A., Gongvatana, A., Okonkwo, O. C., Johanson, C. E., Stopa, E. G., Donahue, J. E., Silverberg, G. D., Initiative, A. D. N., et al. (2010). Brain ventricular volume and cerebrospinal fluid biomarkers of alzheimer's disease. *Journal of Alzheimer's Disease*, 20(2):647–657.

[Papakostas et al., 2015] Papakostas, G. A., Savio, A., Gracia, M., and Kaburlasos, V. (2015). A lattice computing approach to alzheimer's disease computer assisted diagnosis based on MRI data. *Neurocomputing*, 150:35–42.

[Savio et al., 2009] Savio, A., Sebastian, M., Hernàndez, C., García, M., and Villanùa, J. (2009). Classification results of artificial neural networks for alzheimer's disease detection. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 641–648.

[Savio et al., 2011] Savio, A., García, M., and Villanùa, J. (2011). Deformation based features for alzheimer's disease detection with linear SVM. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 336–343.

[Savio et al, 2009] Savio, A., García, M., Grana, M., Villanùa, J. (2009). Results of an adaboost approach on alzheimer's disease detection on MRI. In *Bioinspired Applications in Artificial and Natural Computation*, pages 114–123.

[Savio et al, 2011] Savio, A., García, M. T., Chyzyk, D., Hernàndez, C., Grana, M., Sistiaga, A., Villanùa, J. (2011). Neurocognitive disorder detection based on feature vectors extracted from vbm analysis of structural MRI. *Computers in biology and medicine*, 41(8):600–610.

[Shaw et al., 2009] Shaw, L. M., Vanderstichele, H., Knapik-Czajk, M., Clark, C. M., Aisen, P. S., Petersen, R. C., and Dean, R. (2009). Cerebrospinal fluid biomarker signature in alzheimer's disease neuroimaging initiative subjects. *Annals of neurology*, 65(4):403–413.

[Smith, 2002] Smith, S. (2002). Fast robust automated brain extraction. *Human brain mapping*, 17(3):143–155.

[Talairach, 1988] Talairach, J., T. P. (1988). Co-planar stereotaxic atlas of the human brain. 3-dimensional proportional system: an approach to cerebral imaging.

[Toews et al., 2010] Toews, M., W., W., Collins, D., and Arbel, T. (2010). Feature-based morphometry: Discovering group-related anatomical patterns. *NeuroImage*, 49(3):2318–2327.

[Wolz et al., 2011] Wolz, R., Julkunen, V., Koikkalainen, J., Niskanen, E., Zhang, D., Rueckert, D., and Initiative, A. D. N. (2011). Multi-method analysis of MRI images in early diagnostics of alzheimer's disease. *PloS one*, 6(10):e25446.

[Zhang et al, 2015] Zhang, Y., Dong, Z., Phillips, P., Wang, S., Ji, G., Yang, J., Yuan, T. F. (2015). Detection of subjects and brain regions related to alzheimer's disease using 3D MRI scans based on eigenbrain and machine learning. *Frontiers in Computational Neuroscience*, 9:66.

 ISBN 978-0-9934207-2-6

Multi-Class U-Net for Segmentation of Non-Biometric Identifiers

Tomislav Hrkać, Karla Brkić, and Zoran Kalafatić

University of Zagreb
Faculty of Electrical Engineering and Computing
Unska 3, 10000 Zagreb, Croatia
{tomislav.hrkac, karla.brkic, zoran.kalafatic}@fer.hr

Abstract

Ubiquity of image and video recording devices, as well as the increasing ease of sharing multimedia contents containing people without their permission induces serious privacy risks. Despite considerable efforts in research on de-identification of such contents, potentially identity-revealing information present in soft and non-biometric identifiers is often neglected. We propose an approach for segmentation of non-biometric identifiers intended for use in a de-identification pipeline that takes into account potentially identity-revealing characteristics such as dressing style, hairstyle, personal items, etc. The proposed approach is based on an adaptation of U-Net fully convolutional deep neural network architecture.

Keywords: De-identification, Semantic segmentation, Deep learning

1 Introduction

Increasing presence of image and video recording devices in daily life has highly facilitated the efficacy of obtaining recordings of people. Social media have made it very easy to share photos of people without asking for their consent. In such circumstances, privacy protection of recorded people is becoming a pressing concern. The problem is further aggravated by the advancement of computer vision algorithms, especially facilitated by the recent success of deep learning. Significant progress has been made in solving difficult problems including face recognition [Parkhi et al., 2015, Kemelmacher-Shlizerman et al., 2016], person re-identification [Ahmed et al., 2015, Xiao et al., 2016], image classification [He et al., 2016], etc, making it easier than ever to automatically recognize people in images and videos and further invade their privacy. In such circumstances, many jurisdictions implement strict regulations for the protection of personal data (e.g. the Data Protection Directive of the European Union). According to such regulations, the identity-revealing data of recorded persons should be removed or obfuscated before making it publicly available.

The process of removing identity-revealing information from data is called de-identification. De-identification aims at protecting the privacy of recorded individuals, while simultaneously maintaining as much data utility and naturalness as possible. Identity-revealing features can be categorized as either *primary biometric* (i.e. distinctive, measurable, generally unique and permanent personal characteristics, such as face, iris or fingerprint), *soft biometric* (i.e. vague physical, behavioral or adhered characteristics that are not necessarily unique, but that can be strong cues for person identification, like body shape, gender, tattoos, skin marks, etc.) or *non-biometric* (temporary and changeable but convey contextual information about the person, like hairstyle or clothing) [Ribarić et al., 2016].

Simple de-identification methods like blurring, pixelization, etc., are often used, usually applied to face only, but they are not sufficient to de-identify soft and non-biometric identifiers [Reid et al., 2013] such as specifically colored and textured clothing, characteristic hairstyles and personal items, skin marks and tattoos, etc. (see Fig. 1). De-identification of non-biometric identifiers, which is the focus of this work is rarely addressed as a problem.

 ISBN 978-0-9934207-2-6

Figure 1: Examples of blurring the silhouette for privacy protection. Note how characteristic soft and non-biometric identifiers such as hair color, clothing, hairstyle etc. remain recognizable and potentially identity-revealing even after blurring. Images from the Clothing Co-Parsing (CCP) dataset [Yang et al., 2014].

In general, any de-identification method can be thought of as consisting of three steps: (i) detection of persons in image or video recordings; (ii) segmentation of identifiers to be de-identified; and (iii) applying the de-identifying transformation to the segmented area. We investigate the possibility of applying a semantic segmentation network to find precise regions corresponding to individual non-biometric identifiers - hairstyle, clothing, personal items such as bags and accessories, etc. – as a prerequisite for their subsequent de-identification that will be able to retain as much naturalness and data utility as possible. We assume that all persons present in the recording have been previously reliably found by a person detector. Although we approach the problem of segmentation of such images in the context of de-identification, we note that there are other potential applications of the proposed method (e.g. automatic generation of person description, automatic recognition based on descriptions, automatic re-identification across multiple cameras, virtual clothing try-on, garment recommendation based on personal clothing style, etc.).

2 Related work

Most research on de-identification of humans in images is still focused on de-identifying primary biometric features, predominately the face. Early approaches to face de-identification involved applying naive transformations such as blurring or pixelization. While these naive transformations can prevent human recognition, it has been shown [Gross et al., 2009] that they can be effectively circumvented using computer vision, by employing a technique called parrot recognition, i.e. by using a classifier trained on images on which the same transformation has been applied. A higher level of privacy protection can be obtained through replacing the face with another, unrelated face, either synthetic [Newton et al., 2005, Gross et al., 2006a, Gross et al., 2006b] of real [Bitouk et al., 2008].

However, face obfuscation does not prevent recognition of the recorded person based on soft biometric and non-biometric identifiers. This can be seen e.g. in the work of [Oh et al., 2016], who show that individuals in a social media setting can be identified even if their faces are masked with the black rectangle and their clothing varies across images.

To the best of our knowledge, existing research on de-identifying soft and non-biometric identifiers in im-

 ISBN 978-0-9934207-2-6

ages is scarce. Tattoo de-identification has been considered in works [Marčetić et al., 2014, Hrkać et al., 2016]. One simple approach to hairstyle de-identification is described in [Prinosil et al., 2015]. De-identification of other non-biometric identifiers has not yet been addressed, although there are some works attempting to perform full-body de-identification, i.e. obfuscate the whole human silhouette. An early example of such an approach is the work [Park and Trivedi, 2005], where a method for tracking and privacy protection in videos is introduced, with a de-identification scheme that covers individual human bounding boxes with colored rectangles. [Agrawal and Narayanan, 2011] propose a method for tracking, segmenting and de-identifying individuals in videos. Two de-identification strategies are considered, one based on exponential pixel blurring and another based on line integral convolution. Although privacy-preserving, these schemes do not maintain data utility and naturalness of images. In [Brkić et al., 2016], full body de-identification of humans in videos is obtained by using neural art. The content of the segmented humans is re-rendered using neural art-based style transfer from a randomly selected style image. While the results are very good in terms of biometric, soft biometric and non-biometric de-identification, the degree of naturalness of the de-identified image depends heavily on the original image, the selected style image and the internal workings of the neural network. A detailed recent overview of de-identification methods for privacy protection can be found in [Ribarić et al., 2016].

Semantic segmentation has also recently attracted a lot of research attention. Especially successful have been approaches based on deep learning. In an early work of [Ciresan et al., 2012], a neural network is trained in a sliding window setup to predict the class label of each pixel by looking at a local window around that pixel as input. The R-CNN method [Girshick et al., 2014] first produces object region proposals and then classifies them using a convolutional network, achieving very good detection and segmentation performance on PASCAL VOC dataset.

[Long et al., 2015] have proposed a fully convolutional network, i.e. a network consisting of only convolutional and pooling layers (no fully connected layers), followed by one up-convolution layer to restore original resolution, which has since then become a benchmark in semantic segmentation. This idea was refined by [Noh et al., 2015], where a symmetric structure is proposed, composed of convolutional and pooling layers followed by the same number of up-convolution and un-pooling layers. These models are linear models with no skip connections between non-neighbouring convolutional layers and they were successfully applied to general-purpose semantic segmentation tasks.

In the area of biomedical image segmentation, the U-Net model has recently been proposed by [Ronneberger et al., 2015]. Similarly to [Noh et al., 2015], this network has a symmetrical encoder-decoder structure, but adds so-called skip connections between corresponding encoding and decoding layer pairs. These connections enable a more precise matching of the final segmentation map with object structures and boundaries in the image of the original resolution. The network produces a binary foreground/background map corresponding to biomedical structures. A similar network called SegNet has recently been proposed for general-purpose segmentation [Badrinarayanan et al., 2017].

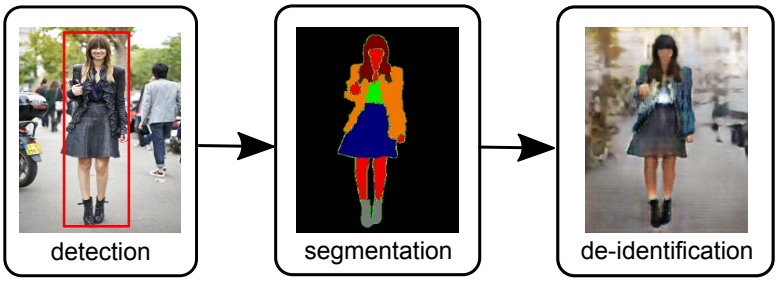


Figure 2: A pipeline for non-biometric de-identification

In contrast to the above described works, we develop a special purpose fully convolutional network for segmentation of non-biometric identifiers, intended as a step in a pipeline for non-biometric de-identification, see Fig. 2. The network is based on an adaptation of the U-Net architecture in order to accommodate multi-class segmentation inherent in our problem.

 ISBN 978-0-9934207-2-6

3 The proposed method

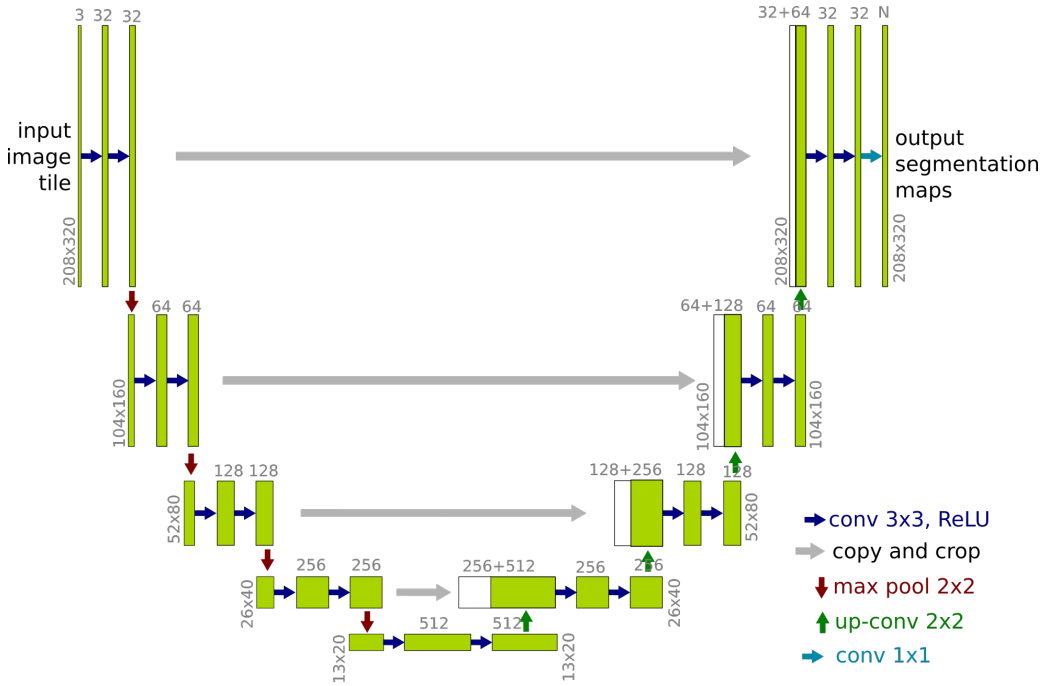


Figure 3: Architecture of the used U-Net, adapted from [Ronneberger et al., 2015]

In this work, we use the U-Net model to segment individual non-biometric identifiers (garments, accessories, hair and skin). As mentioned previously, the U-Net model was originally proposed by [Ronneberger et al., 2015] in the context of binary (foreground/background) segmentation of gray-level images for biomedical applications. It is a fully convolutional network with symmetrical structure, composed of a contracting and an up-sampling part. The contracting part is composed of four pairs of convolutional layers, each pair followed by a 2×2 max-pooling. All convolutions are 3×3 and use zero padding. As the resolution of each successive convolutional layer pair is reduced, the number of feature maps is doubled. The final max-pooling layer is followed by two more convolutional layers, and then by the symmetrically constructed up-sampling part, composed of four blocks of 2×2 up-sampling and pairs of convolutional layers (3×3 convolutions with zero padding). Moreover, corresponding pairs of convolutional layers in the contracting and up-sampling parts are connected by skip-connections. The skip-connections, realized as simple concatenations of feature maps, combine the output of an up-sampling layer with the corresponding features from the contracting part. In this way, successive convolution layers in the up-sampling part can learn to produce better localized output, allowing the network to perform more precise segmentations. All layers use ReLU non-linearity, except for the last layer where Softmax is used to select the best scoring category.

We follow the basic architecture proposed by [Ronneberger et al., 2015], but introduce several modifications in order to adapt it to a new problem of multi-class non-biometric segmentation: (i) RGB color images are used as input, instead of gray-level images; (ii) The output segmentation layer is expanded from 1 to N feature maps to enable multi-class segmentation; (iii) Batch normalization [Ioffe and Szegedy, 2015] is used in all layers to improve learning; (iv) The input resolution (and consequently the resolution of further layers) is modified to match the resolution of our input images; (v) The number of feature maps in all layers is reduced by factor 2 compared to the original architecture, in order to speed-up the learning process and enable the model to fit in the memory of the used GPU. The input resolution of the first layer of the network is set to 208×320 pixels. The final architecture of the used U-Net is depicted in Fig. 3.

Training the network to perform segmentation in a fully supervised manner is accomplished end-to-end, by minimizing the standard categorical cross-entropy function on a pixel-wise basis:

 ISBN 978-0-9934207-2-6

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)], \quad (1)$$

where y_i are true labels, $\hat{y}_i$ predicted labels, and N is the number of classes.

4 Experimental evaluation

4.1 Dataset

For experimental evaluation, we use the Clothing Co-Parsing (CCP) dataset [Yang et al., 2014]. The dataset consists of 1004 images with precise pixel-level annotations and further 1094 images with only image-level annotations. The labels correspond to 60 different categories of garments, accessories, jewellery, skin, hair and background. The images are of somewhat variable resolution, the average being 828×550. Some examples of images and annotations from the CCP dataset are shown in the top row of Fig 4.

Since the number of categories in the CCP dataset is too detailed for our application and some of the garment categories are quite similar (e.g. there are 10 different kinds of footwear in the original label set), we redefine the label set by grouping the original categories to 14 groups (the final 14 categories are shown in Table 1).

4.2 Experimental setup

For training and testing the network we use only the 1004 images from the CCP dataset with pixel-level annotations. Out of these 1004 images, the last 204 images comprise the test set. The remaining 800 images are divided into training and validation sets in a 9:1 ratio (i.e. 720 images are used for training and 80 for validation). The validation set was used to monitor the training process and to prevent overfitting. All images were resized to the resolution of 208×320 pixels, to match the input resolution of the network.

The network was implemented in TensorFlow using the Keras wrapper and trained for 100 epochs, using the *Adadelta* optimizer with default parameters.

4.3 Segmentation performance

The results obtained on the test set are presented in Table 1. We report the results in the form of per-class average precision as well as pixel accuracy for all classes. As we can see, the pixel accuracy, as well as per-class average precision for most categories, are satisfactory and comparable to other state-of-the-art general-purpose semantic segmentation approaches (e.g. [Long et al., 2015, Noh et al., 2015, Badrinarayanan et al., 2017]). The only exception are the categories for which there are very few or no examples in the training set, such as scarves/ties and underwear. Since to the best of our knowledge there has been no prior work on semantic segmentation of non-biometric identifiers specifically, we can not compare the obtained results to other works.

Category	Background	Hair	Skin	Accessories	Hat	Coat	Shirt
AP (%)	98.7	77.2	81.4	38.5	62.9	57.7	47.1
Category	Trousers	Dress	Skirt	Underwear	Socks	Scarf/Tie	Shoes
AP (%)	66.4	54.1	25.0	9.8	38.5	2.9	58.8
Pixel accuracy (%)**:**	89.0						

Table 1: Precision of semantic segmentation for fourteen categories and pixel accuracy

Several examples of segmentation from the test set are shown in Fig. 4.

 ISBN 978-0-9934207-2-6

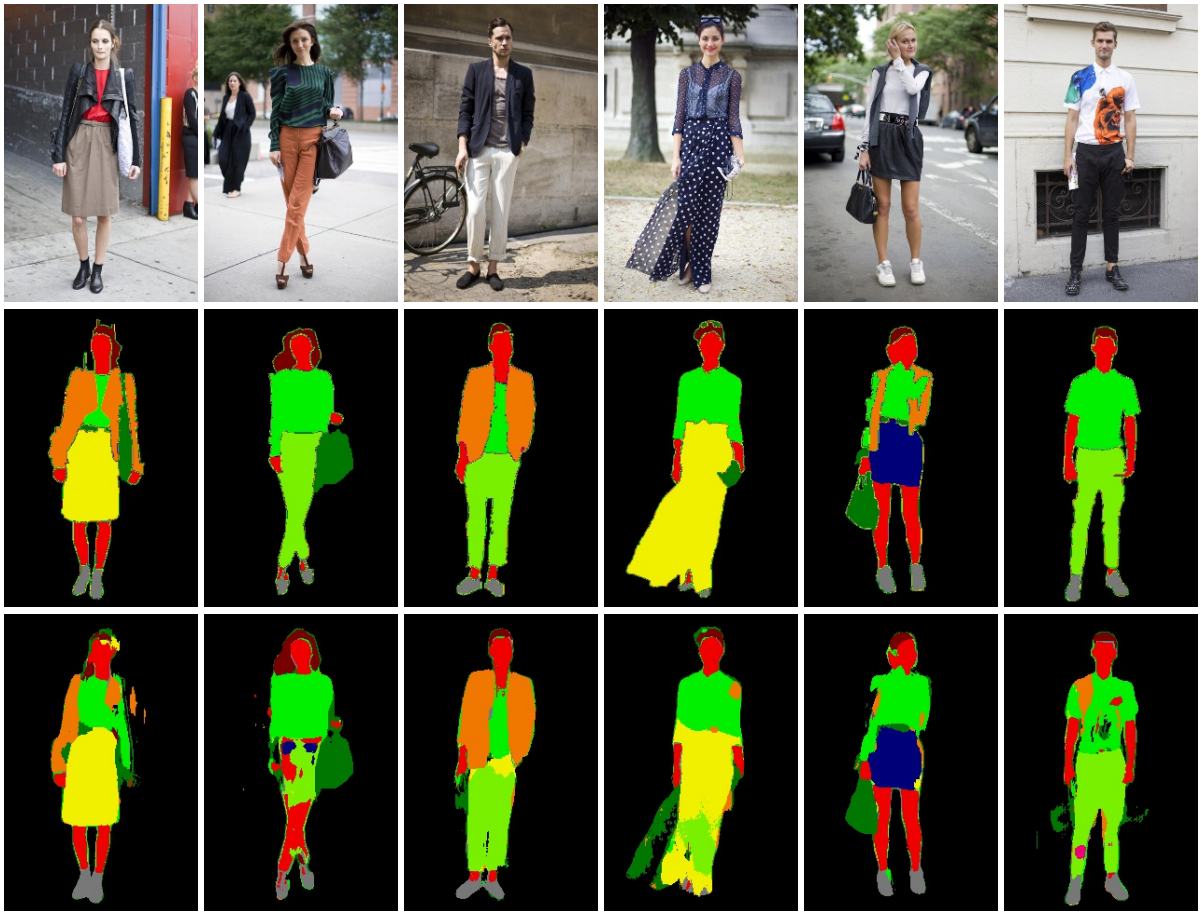


Figure 4: An illustration of the segmentation results. Top row: original images. Middle row: ground truth. Bottom row: segmentation results.

5 Conclusion and outlook

We addressed the challenging problem of semantic segmentation of non-biometric identifiers as a first step in a system for their de-identification. Our findings indicate that using an appropriate adaptation of a fully convolutional U-Net architecture can be a reliable way to perform this task. Like all approaches based on deep learning, the proposed approach is dependent on a large training dataset and the underrepresentation of certain categories in the dataset results in a lower segmentation performance for such categories. As we are not aware of any larger training datasets for non-biometric identifiers with pixel-level annotations, in future work we plan to investigate the possibility of further improving the performance by weakly supervised and semi-supervised learning, using available large datasets with image-level annotations. Furthermore, we plan to investigate the possibility of using the proposed system in a full non-biometric de-identification pipeline, where color, texture and type of garments and hairstyles can be altered to better protect the privacy of the recorded people.

Acknowledgments

This work has been supported by the Croatian Science Foundation, within the project "De-identification Methods for Soft and Non-Biometric Identifiers" (DeMSI, UIP-11-2013-1544). This support is gratefully acknowledged.

References

[Agrawal and Narayanan, 2011] Agrawal, P. and Narayanan, P. J. (2011). Person de-identification in videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(3):299–310.

[Ahmed et al., 2015] Ahmed, E., Jones, M., and Marks, T. K. (2015). An improved deep learning architecture for person re-identification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Badrinarayanan et al., 2017] Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[Bitouk et al., 2008] Bitouk, D., Kumar, N., Dhillon, S., Belhumeur, P., and Nayar, S. K. (2008). Face swapping: Automatically replacing faces in photographs. *ACM Trans. Graph.*, 27(3):39:1–39:8.

[Brkić et al., 2016] Brkić, K., Sikirić, I., Hrkać, T., and Kalafatić, Z. (2016). De-identifying people in videos using neural art. In *Proc. IEEE Int. Conf. on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–6.

[Ciresan et al., 2012] Ciresan, D., Giusti, A., Gambardella, L. M., and Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25*, pages 2843–2851. Curran Associates, Inc.

[Girshick et al., 2014] Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Gross et al., 2006a] Gross, R., Airoldi, E., Malin, B., and Sweeney, L. (2006a). *Privacy Enhancing Technologies: 5th International Workshop*, chapter Integrating Utility into Face De-identification, pages 227–242. Springer Berlin Heidelberg.

[Gross et al., 2009] Gross, R., Sweeney, L., Cohn, J. F., De la Torre, F., and Baker, S. (2009). *Protecting Privacy in Video Surveillance*, chapter Face De-identification, pages 129–146. Springer Publishing Company Incorporated.

[Gross et al., 2006b] Gross, R., Sweeney, L., de la Torre, F., and Baker, S. (2006b). Model-based face de-identification. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 161–161.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Hrkać et al., 2016] Hrkać, T., Brkić, K., Ribarić, S., and Marčetić, D. (2016). Deep learning architectures for tattoo detection and de-identification. In *Proc. SPLINE*, Aalborg. IEEE.

[Ioffe and Szegedy, 2015] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Bach, F. R. and Blei, D. M., editors, *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 448–456. JMLR.org.

[Kemelmacher-Shlizerman et al., 2016] Kemelmacher-Shlizerman, I., Seitz, S. M., Miller, D., and Brossard, E. (2016). The megaface benchmark: 1 million faces for recognition at scale. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

 ISBN 978-0-9934207-2-6

[Long et al., 2015] Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440.

[Marčetić et al., 2014] Marčetić, D., Ribarić, S., Štruc, V., and Pavešić, N. (2014). An experimental tattoo de-identification system for privacy protection in still images. In *Proc. MIPRO*, volume 1, pages 1288–1293, Opatija. MIPRO.

[Newton et al., 2005] Newton, E. M., Sweeney, L., and Malin, B. (2005). Preserving privacy by de-identifying face images. *IEEE Transactions on Knowledge and Data Engineering*, 17(2):232–243.

[Noh et al., 2015] Noh, H., Hong, S., and Han, B. (2015). Learning deconvolution network for semantic segmentation. *CoRR*, abs/1505.04366.

[Oh et al., 2016] Oh, S. J., Benenson, R., Fritz, M., and Schiele, B. (2016). Faceless person recognition; privacy implications in social media. *CoRR*, abs/1607.08438.

[Park and Trivedi, 2005] Park, S. and Trivedi, M. M. (2005). A track-based human movement analysis and privacy protection system adaptive to environmental contexts. In *Advanced Video and Signal Based Surveillance, 2005. AVSS 2005. IEEE Conference on*, pages 171–176.

[Parkhi et al., 2015] Parkhi, O. M., Vedaldi, A., and Zisserman, A. (2015). Deep face recognition. In *Proceedings of the British Machine Vision Conference (BMVC)*.

[Prinosil et al., 2015] Prinosil, J., Krupka, A., Riha, K., Dutta, M. K., and Singh, A. (2015). Automatic hair color de-identification. In *Proc. ICGCIoT*, pages 723–736.

[Reid et al., 2013] Reid, D., Samangooei, S., Chen, C., Nixon, M., and Ross, A. (2013). Soft biometrics for surveillance: an overview. In *Machine Learning: Theory and Applications*, 31, pages 327–352. Elsevier.

[Ribarić et al., 2016] Ribarić, S., Ariyaeeinia, A., and Pavešić, N. (2016). De-identification for privacy protection in multimedia content: A survey. *Signal Processing: Image Communication*, 47:131 – 151.

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*, pages 234–241. Springer International Publishing, Cham.

[Xiao et al., 2016] Xiao, T., Li, H., Ouyang, W., and Wang, X. (2016). Learning deep feature representations with domain guided dropout for person re-identification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Yang et al., 2014] Yang, W., Luo, P., and Lin, L. (2014). Clothing co-parsing by joint image segmentation and labeling. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '14, pages 3182–3189, Washington, DC, USA. IEEE Computer Society.

A Restricted-Domain Dual Formulation for Two-Phase Image Segmentation

Jack Spencer

Department of Mathematics,
University of Liverpool, UK.

Abstract

In two-phase image segmentation, convex relaxation has allowed global minimisers to be computed for a variety of data fitting terms. Many efficient approaches exist to compute a solution quickly. However, we consider whether the nature of the data fitting in this formulation allows for reasonable assumptions to be made about the solution that can improve the computational performance further. In particular, we employ a well known dual formulation of this problem and solve the corresponding equations in a restricted domain. We present experimental results that explore the dependence of the solution on this restriction and quantify imrovements in the computational performance. This approach can be extended to analogous methods simply and could provide an efficient alternative for problems of this type.

Keywords: Image Processing, Segmentation, Total Variation, Convex Relaxation, Dual Formulation.

1 Introduction

Image segmentation is the meaningful partitioning of an image based on certain characteristics. In two-phase segmentation this consists of determining the foreground and background of a domain $\Omega \in \mathbb{R}^2$, i.e. find a closed boundary separating subregions Ω_1 and Ω_2. This is distinct from multiphase approaches, where more than two separate regions are determined. Our work concerns the continuous setting, which we will briefly discuss in the next section. Equivalent problems in the discrete setting have been well studied with details found in [Boykov and Kolmogorov, 2004]. A comprehensive background behind the following functional can also be found in [Chambolle and Pock, 2016]. Briefly, the aim is to determine an indicator function, $u(x)$, that labels the foreground and background by minimising the following energy:

$$\min_{u\in\{0,1\}} \left\{ \int_\Omega |\nabla u(x)| \, dx + \lambda \int_\Omega f(x)u(x)dx \right\}. \tag{1}$$

The function $f(x)$ is typically referred to as the fitting term, determining how the segmentation solution corresponds to the data. It is balanced by a regularisation term, in this case the total variation (TV) semi-norm which penalises the length of the segmentation boundary. For large λ the following will hold precisely:

$$\begin{aligned} f(x) < 0, &\quad \text{foreground,} \\ f(x) > 0, &\quad \text{background.} \end{aligned}$$

When λ is varied the solution for u will fit the data with more regularity, i.e. some areas where $f < 0$ will be background and some areas where $f > 0$ will be foreground. However, this is most likely where f is close to 0. With that in mind our work considers what improvements can be made by concentrating on regions in the domain where $f(x) \approx 0$. An exception to this concerns cases where f has strong noise, which we will return to

 ISBN 978-0-9934207-2-6

later. A number of choices for f exist depending on the application, including piecewise-constant segmentation based on the work of [Chan and Vese, 2001]]:

$$f(x) = (z - c_1)^2 - (z - c_2)^2, \tag{2}$$

where c_1 and c_2 are intensity constants indicating average foregound and background intensitites of the image $z(x)$, respectively. We also consider a selection based fitting term such as [Spencer and Chen, 2015]

$$f(x) = (z - c_1)^2 - (z - c_2)^2 + \gamma P(x), \tag{3}$$

where $\gamma P(x)$ is a distance selective term based on user input. In Section 4 we present results using equations (2) and (3). This approach is not limited to the fitting terms mentioned above, and can be extended to any segmentation problem in this framework. Alternatives for future consideration include bias field segmentation [Chen et al., 2013] and interactive convex active contours [Nguyen et al., 2012].

A restricted-domain approach is analogous to banded segmentation methods such as [Rommelse et al., 2003] and [Zhang et al., 2014], among many others. This work differs in the sense that the restriction is based on the values of the fitting term, rather than the location of the boundary at an iteration, which is potentially simpler computationally. We compute an approximation of the global minimiser of the energy (1), with the accuracy determined by the level of domain restriction. This is based on the following initialisation of the indicator function: $u^{(0)} = H(-f)$, where $H(\cdot)$ is the Heaviside function.

In the following we will briefly introduce existing methods for finding the global minimisers of two-phase segmentation problems, introducing the dual formulation of [Bresson et al., 2007] based on [Chambolle, 2004]. We then discuss the proposed approach where a restricted domain based on the fitting term is considered, before detailing the method and how it relates to [Bresson et al., 2007]. Finally, we present some results for three examples for various restrictions on the domain, quantifying the accuracy and computational performance in comparison to the original method. We then offer some concluding remarks.

2 Convex Relaxation for Two-Phase Segmentation

We now introduce the details of the approach we consider in this work. Again, a comprehensive background of this work can be found in [Chambolle and Pock, 2016] and many others. Essentially, convex relaxation in this case involves relaxing the binary constraint in the original functional (1), i.e. $u \in [0, 1]$. The seminal work here is [Chan et al., 2006] who found global minimisers of the two-phase piecewise-constant Mumford-Shah model [Mumford and Shah, 1989] (assuming fixed intensity constants, c_1, c_2). Therefore, the problem considered here is:

$$\min_{u \in [0,1]} \left\{ \int_\Omega |\nabla u(x)| \, dx + \lambda \int_\Omega f(x) u(x) dx \right\}. \tag{4}$$

In [Chan et al., 2006] they introduce a penalty function to enforce the constraint on u and solve using time marching. It is also possible to use additive operator splitting [Spencer and Chen, 2015], split Bregman [Goldstein et al., 2010], and Chambolle-Pock [Chambolle and Pock, 2011] among many others. However, initially we intend to implement our restricted-domain approach on the dual formulation used in [Bresson et al., 2007]. We breifly detail this approach next.

2.1 Dual Formulation

The dual formulation of this problem was first introduced by [Bresson et al., 2007], based on the work of [Chambolle, 2004], [Aujol et al., 2006] and the references therein. The idea is to introduce a new variable, $v(x)$, and minimise the following functional alternately:

$$\min_{u,v} \left\{ \int_\Omega |\nabla u(x)| \, dx + \frac{1}{2\theta} \int_\Omega (u(x) - v(x))^2 \, dx + \int_\Omega \lambda f(x) v(x) + \alpha \psi(v) \, dx \right\}, \tag{5}$$

 ISBN 978-0-9934207-2-6

where $\psi(v) = \max\{0, 2|v-1/2|-1\}$. By splitting the variables in this way, the minimisation of u concentrates on the TV term, and the minimisation of v satisifes the fitting and constraint requirements. In [Bresson et al., 2007] the regularisation term is weighted, however here we concentrate on the original problem (4). The parameter α ensures the constraints on the indicator function $u(x)$ in (5) are met, and can be set automatically [Chan et al., 2006]. The minimistion of u and v can be achieved iteratively by the following steps. With fixed v, the solution of u is given by

$$u(x) = v(x) - \theta\nabla \cdot \rho(x),$$

where $\rho = (\rho^1, \rho^2)$ is the solution of

$$\nabla(\theta\nabla \cdot \rho - v) - |\nabla(\theta\nabla \cdot \rho - v)|\rho = 0,$$

which can be solved by a fixed point method. With fixed u, the solution for v is given as

$$v(x) = \min\{\max\{u(x) - \theta\lambda r(x), 0\}, 1\}.$$

This is repeated until convergence. Further details can be found in [Bresson et al., 2007], including the definition of the discrete gradient and divergence operators from [Chambolle, 2004].

3 Proposed Approach: Segmentation in a Restricted Domain

We now introduce our approach for reducing the computation time for this type of problem. We begin by assuming the solution in certain parts of the discretised domain based on the values of $f(x)$ for a given problem. The indicator function is then fixed at 1 or 0 at these points for the foreground (Fg) and background (Bg) respectively. We solve the equation in the remaining region, which we call the restricted domain (RD). Let us define $q \in [0,1]$ such that the following thresholding of the fitting function holds. We define a value $\hat{q} \in \mathbb{R}$ such that the percentage of nodes in the restricted domain is q ($\times 100$):

$$\begin{cases} Fg &= \{x : x \in \Omega,\ f(x) \le -\hat{q}\} \\ Bg &= \{x : x \in \Omega,\ f(x) \ge \hat{q}\} \\ RD &= \{x : x \in \Omega \setminus Fg \setminus Bg\}. \end{cases}$$

The value of $\hat{q}$ is initially 0 and is increased until the selected value of q is satisfied. In other words, for $q = 0$, $RD = \emptyset$ and the solution is a zero-thresholding of the fitting function, or equivalent to selecting a large λ in the original problem (4). For $q = 1$, $Fg = \emptyset$ and $Bg = \emptyset$ and we consider the problem in a conventional manner with no restriction on the domain. For $q \in (0,1)$ we consider a restricted domain of varying degrees as illustrated in Figure 1, where the corresponding region of interest is given in grey. This means that we need to minimise the energy in a restricted domain which, when combined with the dual formulation of [Bresson et al., 2007] and [Chambolle, 2004], means that we are solving the following equation for ρ at certain points:

$$\nabla(\theta\nabla \cdot \rho - v) - |\nabla(\theta\nabla \cdot \rho - v|\rho = 0. \tag{6}$$

This involves making certain assumptions about the solution for ρ. In [Bresson et al., 2007], it is initialised as $\rho^{(0)} = 0$ and the solution of (6) is clearly dependent on u and v. However, for the initialisation $u^{(0)} = -H(f)$ the solution of ρ has a predictable form. In particular, $\rho \approx 0$ where $|f|$ is largest and $\rho \in [-1,1]$ where $|f|$ is closer to 0. If q is selected sensibly (we will return to this later), when $u^{(0)} = -H(f)$ it is reasonable to assume that the solution of (6) for $x \in Fg \cap Bg$ is $\rho^*(x) = 0$. Clearly, the larger q is the less reliable this assumption becomes and the corresponding solution for ρ will be less accurate. However, part of this work concerns what consists of a sensible selection for q and whether it is possible to make reasonable assumptions about f that can improve the efficiency of minimising the original formulation (4).

We now ellaborate on the details behind minimising (4) with a dual formulation in a restricted domain. We first consider the following minimisation problem:

$$\min_u \left\{ \int_\Omega |\nabla u(x)|\ dx + \frac{1}{2\theta} \int_\Omega (u(x) - v(x))^2\ dx \right\}.$$

 ISBN 978-0-9934207-2-6

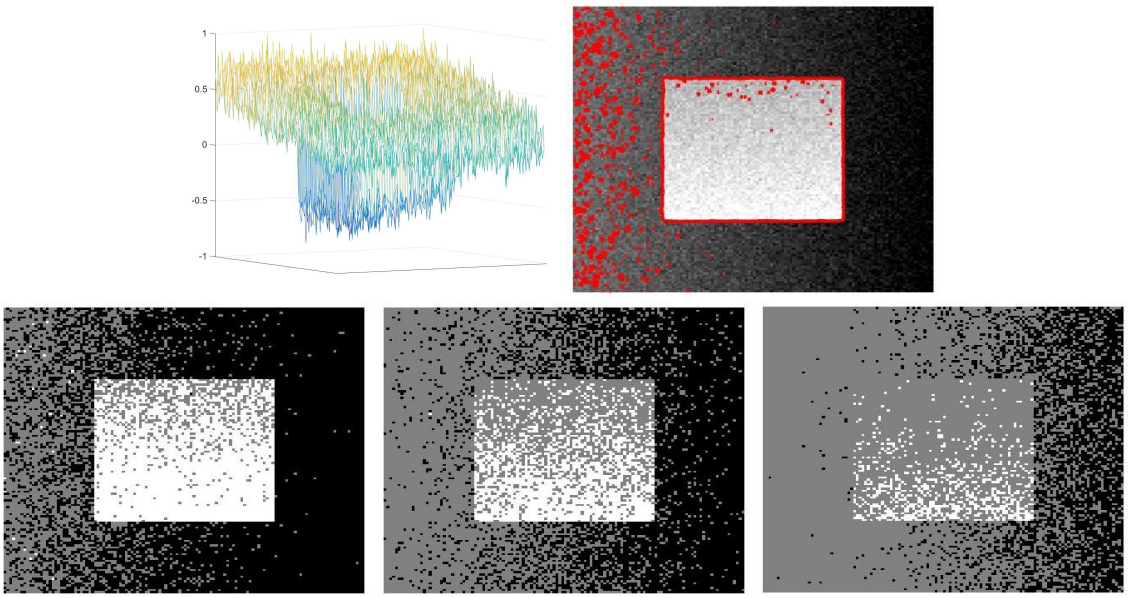


Figure 1: Restricted Domain: The top row shows the fitting term (left) and the image (right) with the zero level-set of $f(x)$ given in red. The bottom row indicates the corresponding regions Fg (white), Bg (black), and RD (grey) for values of $q = 0.25, 0.5, 0.75$ from left to right, respectively.

The solution, based on our approach, is given by

$$u(x) = \begin{cases} 1, & \text{for } x \in Fg \\ 0, & \text{for } x \in Bg \\ v - \theta \nabla \cdot \rho, & \text{for } x \in RD, \end{cases} \tag{7}$$

where $\rho = (\rho_1, \rho_2)$ satisfies

$$\rho = \begin{cases} 0, & \text{for } x \in Fg \cap Bg \\ \nabla(\theta \nabla \cdot \rho - v) - |\nabla(\theta \nabla \cdot \rho - v)|\rho = 0, & \text{for } x \in RD. \end{cases} \tag{8}$$

For $x \in RD$ the following fixed point method, with time step τ, will solve the equation for ρ:

$$\rho^{n+1} = \frac{\rho^n + \tau \nabla(\nabla \cdot \rho^n - v/\theta)}{1 + \tau |\nabla \rho^n - v/\theta|}.$$

As before, the following minimisation problem is then solved with u fixed:

$$\min_v \left\{ \frac{1}{2\theta} \int_\Omega (u(x) - v(x))^2 \, dx + \int_\Omega \lambda f(x) v(x) + \alpha \psi(v) \, dx \right\}.$$

We combine our assumptions about u and ρ with the work of [Bresson et al., 2007] to give the corresponding solution as

$$v(x) = \begin{cases} 1, & \text{for } x \in Fg \\ 0, & \text{for } x \in Bg \\ \min\{\max\{u(x) - \theta\lambda f(x), 0\}, 1\}, & \text{for } x \in RD. \end{cases} \tag{9}$$

As with [Bresson et al., 2007], as discussed in the previous section, u and v are minimised alternately until convergence. The main advantage of this approach concerns finding the solution of ρ at each iteration with the fixed point method detailed above. Based on the choice of q it is possible that significant advantages exist in terms of computation time with minimal compromise on the quality of the solution. We will discuss some exceptions to this, as well as future considerations in the following sections.

 ISBN 978-0-9934207-2-6

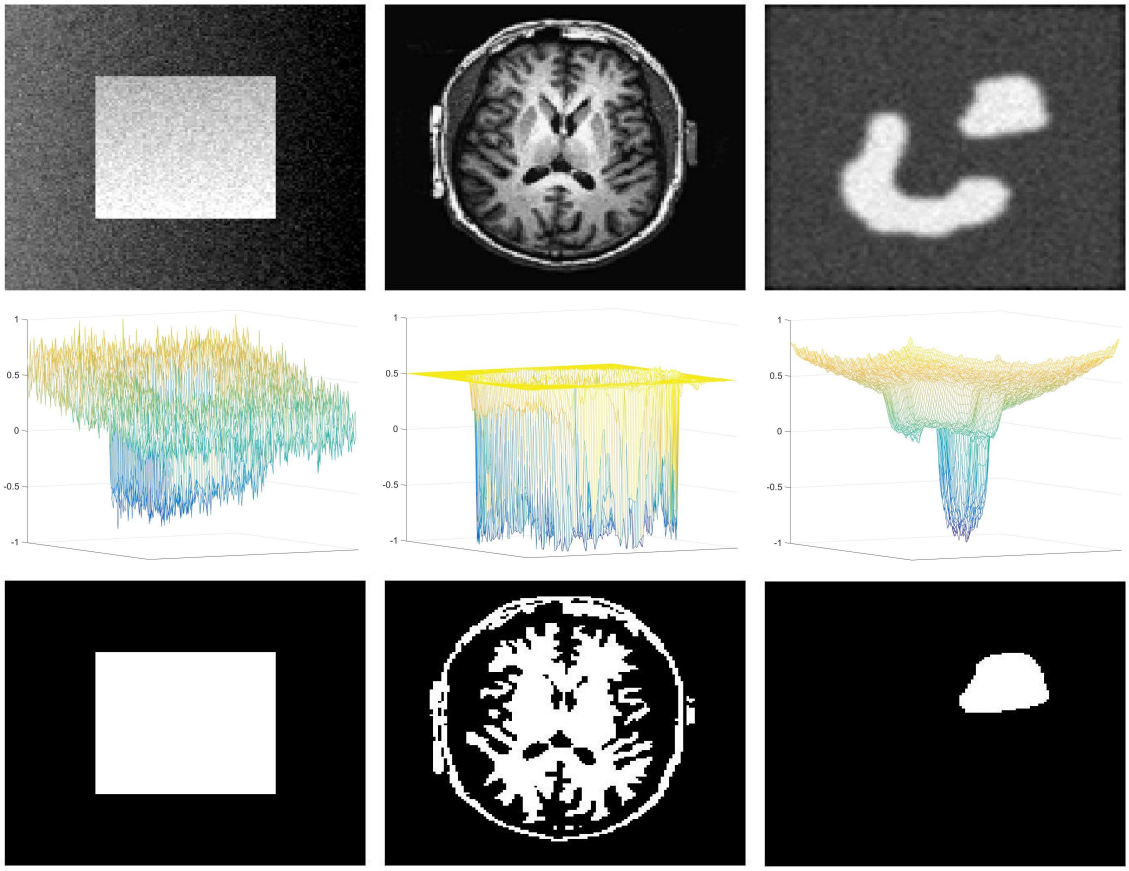


Figure 2: Test Problems: From left to right are Examples 1-3. From top to bottom are the image (z), fitting term (f), and thresholded segmentation result (GT) using the original method of [Bresson et al., 2007].

q	Example 1		Example 2		Example 3	
	E_1	E_2	E_1	E_2	E_1	E_2
0	0.876	5.64×10^{2}	0.943	2.49×10^{2}	0.920	5.59×10^{1}
0.1	0.961	1.59×10^{2}	0.946	1.47×10^{2}	0.962	1.19×10^{1}
0.2	0.987	5.16×10^{1}	0.964	8.90×10^{1}	0.969	1.05×10^{1}
0.3	0.995	1.94×10^{1}	0.982	4.94×10^{1}	0.974	8.02×10^{0}
0.4	0.999	6.76×10^{0}	0.987	3.47×10^{1}	0.989	3.45×10^{0}
0.5	1.000	3.24×10^{0}	0.988	2.98×10^{1}	0.995	7.55×10^{-1}
0.6	1.000	2.65×10^{0}	0.989	2.90×10^{1}	0.999	6.03×10^{-1}
0.7	1.000	1.24×10^{0}	0.989	2.90×10^{1}	1.000	4.78×10^{-1}
0.8	1.000	5.34×10^{-1}	0.989	2.90×10^{1}	0.999	4.24×10^{-1}
0.9	1.000	2.49×10^{-1}	0.989	2.73×10^{1}	1.000	3.30×10^{-1}
1	1.000	2.10×10^{-5}	1.000	3.28×10^{-2}	1.000	2.10×10^{-1}

Table 1: Results: For Examples 1-3 we vary $q \in [0, 1]$ and provide E_1 and E_2.

4 Experimental Results

In this section we introduce some results for the test problems shown in Figure 2, using (2) for f in Examples 1 and 2 and (3) for f in Example 3. The focus of these results is to determine the dependence on q, i.e. to what extent can we restrict the domain for problems of this type? As a comparison, we use a result from [Bresson et al., 2007] (for a particular choice of λ in each case). Specifically, we iterate until the following stopping criterion is met at the ℓ^{th} iteration:

$$\max\left\{\| u^{(\ell)} - u^{(\ell-1)} \|, \| v^{(\ell)} - v^{(\ell-1)} \|\right\} \leq \delta.$$

 ISBN 978-0-9934207-2-6

For $\delta = 10^{-10}$ we set $u^{GT}(x) = u^{(\ell)}$ in the original dual formulation. We also use the thresholded result:

$$GT(x) = \begin{cases} 1, & \text{for } x \in u^{GT}(x) > \epsilon \\ 0, & \text{for } x \in u^{GT}(x) \leq \epsilon. \end{cases}$$

Following convention we set $\epsilon = 0.5$. We refer to the solution for the proposed method (with $\delta = 10^{-2}$) as u^* and its corresponding thresholded result as Ω_1^*. This allows us to define the two error measurements that we use in discussing the results when varying the parameter in the proposed method, q. The first is the Tanimoto Coefficient between the thresholded results, and the second is the L^2 difference between the proposed solution and the original solution:

$$E_1 = \frac{N(GT \cap \Omega_1^*)}{N(GT \cup \Omega_1^*)}, \qquad E_2 = \int_\Omega (u^* - u^{GT})^2 \, dx.$$

Here $N(\cdot)$ refers to the number of nodes in the enclosed region and $E_1 \in [0,1]$, with $E_1 = 1$ indicating a perfect result. With the second error measurement, clearly we expect E_2 to approach 0 as q increases. We don't necessarily expect E_2 to be 0 for $q \leq 1$ as u^{GT} is not binary precisely and in the tests we use $\delta = 10^{-2}$, so there is likely to be a minor difference. Whilst E_2 is a useful measure to demonstrate the correspondence between the value of q and the original method, we are primarily interested in E_1 as the crucial indicator of a successful segmentation result.

In Table 1 we present the main results for $q \in [0,1]$. We include $q = 0$ (i.e. a completely thresholded result) and $q = 1$ (i.e. the original method) to demonstrate the full effect of the choice of q. Both error measurements are included (to 3 s.f.) and we can see that increasing q to 0.4 is enough to produce a very good result ($E_1 > 0.98$) in all examples. For Example 2 we can see that for $q < 1$, E_1 does not reach 1 meaning that restricting the domain of the dual formulation always changes the segmentation result for this fitting term. However, there is a very close correspondence between the results even for small values of q, which is encouraging. As expected E_2 tends to decrease as q increases, and the solution in the restricted domain is reasonably close to the original solution. To put these results in context the size of all images tested here are 128×128. These results demonstrate that using a dual formulation in a restricted domain is a viable approach for problems of this type.

In Figure 3 we include the computation time (in seconds, to 1 d.p.) for different choices of $q \in [0,1]$. Clearly the expectation is that as q increases t should also increase, but this does not quite hold absolutely. This is likely to be down to features in these fitting terms that mean minor increases to q, perhaps counterintuitively, slightly simplify the problem. For the original method of [Bresson et al., 2007] the computation time was $t = 6.0s$ for Example 1, $t = 14.0s$ for Example 2, and $t = 8.2s$ for Example 3 (which we will refer to as t_1, t_2, and t_3 respectively). For these results, and in the following, we set $\delta = 10^{-2}$ as a stopping criterion, and use them as a benchmark in each case. From Figure 3 it can be seen that for Examples 1-3 t is only greater than t_1, t_2, or t_3 for $q > 0.9$. For smaller values of q it is possible to make significant gains in terms of computation time. For Example 1, when $E_1 > 0.98$ the average time is $t = 2.8s$ for $q \leq 0.9$. Similarly for Examples 2 and 3, the average times are $t = 8.7s$ and $t = 5.5s$ respectively. This corresponds to a time saving of 53%, 38%, and 33% for Examples 1-3, for cases with an accurate segmentation. Extending these tests to a wider choice of fitting terms, and investigating the effect of changing λ and δ, would help further determine the effectiveness of the proposed approach.

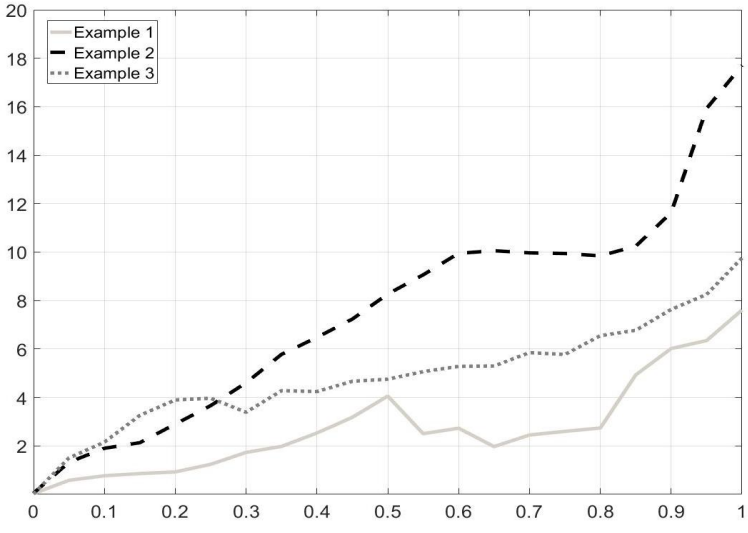


Figure 3: Computation Time, $t(q)$.

 ISBN 978-0-9934207-2-6

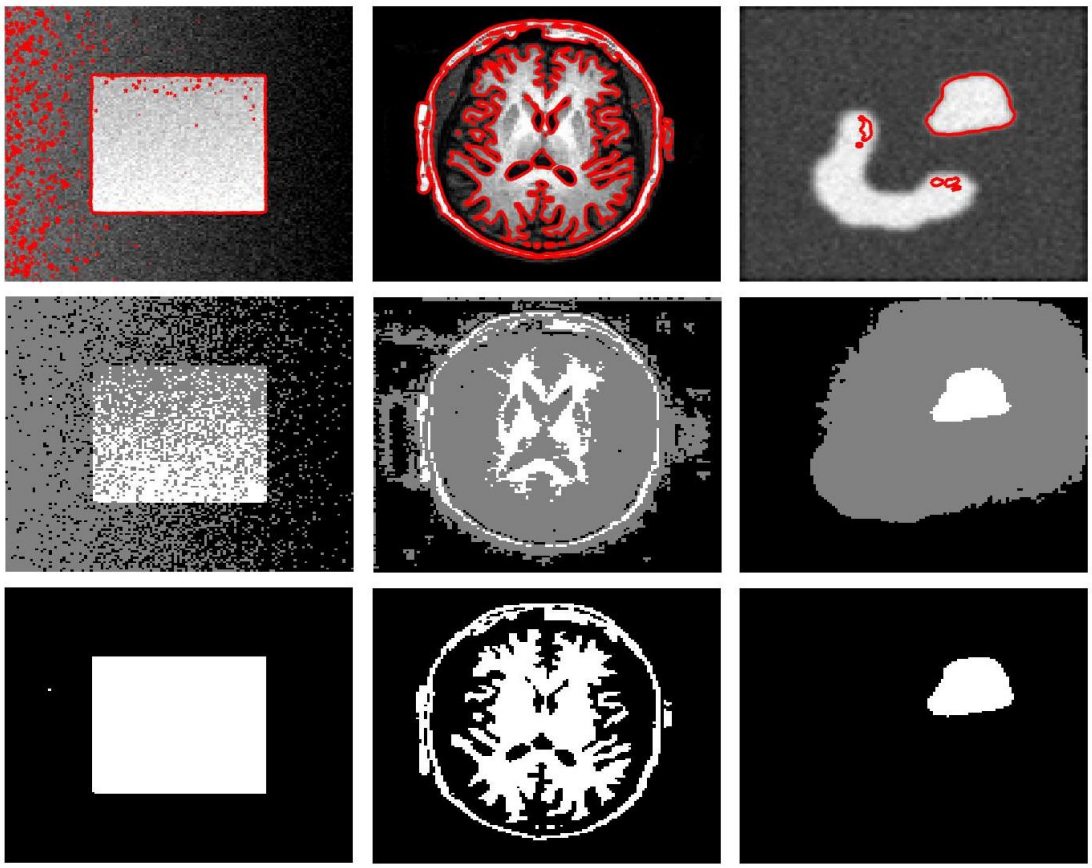


Figure 4: Results: From left to right are Examples 1-3, respectively. From top to bottom are the image with the zero level-set of f in red, the regions Fg (white), Bg (black), RD (grey), and Ω_1^* for $q = 0.5$.

In Figure 4 we present some example results for $q = 0.5$. For Example 1, $E_1 = 1.000$ and $t = 4.1s$. For Example 2, $E_1 = 0.988$ and $t = 8.3s$. For Example 3, $E_1 = 0.995$ and $t = 4.8s$. Compared to the original method, $t_1 = 6.0s$, $t_2 = 14.0s$, and $t_3 = 8.2s$. In each case a significant improvement can be made in terms of computation time with minimal compromise on the quality of the result as measured by E_1.

5 Conclusion

The results presented support the idea that the domain can be restricted for problems of this type, without compromising the quality of the result. This allows for significant gains in terms of computation time. Additional testing to verify these findings would be beneficial, particularly for a wider variety of fitting terms. An example would be where f contains high levels of noise. Further considerations might be necessary to restrict the domain in a robust way, and developing the required understanding would help develop this approach further. For $q = 1$ the computation times for Examples 1-3 are $t = 7.6s, 17.7s$, and $9.8s$ respectively, which corresponds to approximately a 25% increase for the proposed method when no restriction of the domain is considered. If the efficiency of the domain restriction could be improved this would allow for higher values of q to be selected for a reduced cost, which could be particularly beneficial in cases of high noise in the fitting term.

The results presented are for images of size 128×128 in order to explore the viability of restricting the domain for this problem. Improvements in t for larger images, or 3D problems, could be particularly valuable. Extending this approach to these cases is of interest, and would help support the proposed approach further. We are also considering the extension of this approach beyond the dual formulation of [Bresson et al., 2007], such as split-Bregman [Goldstein et al., 2010] and additive operator splitting [Spencer and Chen, 2015]. Following the framework introduced here, assumptions about the solution of (4) can be adapted to other methods in a similar way. However, the initial results presented here are encouraging.

 ISBN 978-0-9934207-2-6

Acknowledgements

The author would like to acknowledge the support of the EPSRC grant EP/N014499/1.

References

[Aujol et al., 2006] Aujol, J. F., Gilboa, G., Chan, T., and Osher, S. (2006). Structure-texture decomposition–modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67(1):111–136.

[Boykov and Kolmogorov, 2004] Boykov, Y. and Kolmogorov, V. (2004). An experimental comparison of min-cut/max-flow algorithms for minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137.

[Bresson et al., 2007] Bresson, X., Esedoglu, S., Vandergheynst, P., Thiran, J. P., and Osher, S. (2007). Fast global minimization of the active contour/snake model. *Journal of Mathematical Imaging and Vision*, 28(2):151–167.

[Chambolle, 2004] Chambolle, A. (2004). An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20:89–97.

[Chambolle and Pock, 2011] Chambolle, A. and Pock, T. (2011). A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40:120–145.

[Chambolle and Pock, 2016] Chambolle, A. and Pock, T. (2016). An introduction to continuous optimization for imaging. *Acta Numerica*, 25:161–319.

[Chan et al., 2006] Chan, T., Esedoglu, S., and Nikolova, M. (2006). Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM Journal on Applied Mathematics*, 66(5):1632–1648.

[Chan and Vese, 2001] Chan, T. and Vese, L. (2001). Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277.

[Chen et al., 2013] Chen, D., Yang, M., and Cohen, L. (2013). Global minimum for a variant Mumford-Shah model with application to medical image segmentation. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 1(1):48–60.

[Goldstein et al., 2010] Goldstein, T., Bresson, X., and Osher, S. (2010). Geometric applications of the split bregman method. *Journal of Scientific Computing*, 45(1-3):272–293.

[Mumford and Shah, 1989] Mumford, D. and Shah, J. (1989). Optimal approximation by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42:577–685.

[Nguyen et al., 2012] Nguyen, T., Cai, J., Zhang, J., and Zheng, J. (2012). Robust interactive image segmentation using convex active contours. *IEEE Transactions on Image Processing*, 21:3734–3743.

[Rommelse et al., 2003] Rommelse, J., Lin, H., and Chan, T. (2003). A robust level set algorithm for image segmentation and its parallel implementation. *UCLA CAM Report*, 03-05.

[Spencer and Chen, 2015] Spencer, J. and Chen, K. (2015). A convex and selective variational model for image segmentation. *Communications in Mathematical Sciences*, 13(6):1453–1472.

[Zhang et al., 2014] Zhang, J., Chen, K., Yu, B., and Gould, D. (2014). A local information based variational model for selective image segmentation. *Inverse Problems and Imaging*, 8(1):293–320.

Spatio-temporal tube segmentation through a video metrics-based patch similarity measure

Patricia Vitoria[1,2], Vadim Fedorov[1], and Coloma Ballester[1]

[1] *Universitat Pompeu Fabra, Spain and* [2] *Technical University of Munich, Germany*

Abstract

This paper presents a new method for video simplification which identifies moving or static objects in a video by grouping together all the pixels corresponding to homogeneous texture and temporal coherent spatio-temporal regions, the so-called *tubes*. This is achieved through the proposal of video metrics defined on the video domain, which provide patches that intrinsically and automatically adapt its spatio-temporal shape and size, and a patch-based comparison measure that is able to capture the scene similarities. Building upon this video metrics-based patch similarity measure we propose a video simplification method that results in tubes made of the pixels with the same vicinity texture content regardless camera point of view or object position changes. We present experiments analyzing the performance of the proposed approach.

Keywords: patch-based method, supervoxel, tube computation, patch similarity, video segmentation.

1 Introduction

Decomposition and understanding dynamic scenes remains a significant challenge in computer vision. One of the first steps in the scene analysis is the segmentation of it into supervoxels. The output data - supervoxels - has more significant information than each of its pixels and the complexity of the input video is reduced. Changes of position (thus producing affine or projective distortions), illumination, and the interaction of the objects with the surrounding (thus sometimes disappearing or being partially occluded or disoccluded) makes the problem even more challenging.

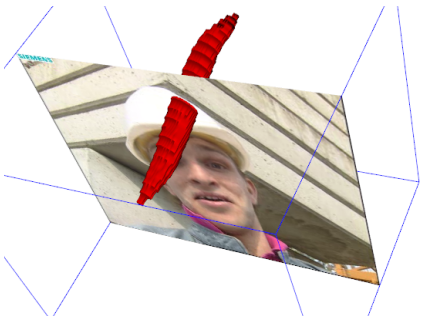


Figure 1: Spatio-temporal adaptive patch (in red) corresponding to the metric tensor of a point close to the helmet boundaries of the well-known Foreman video sequence where the man speaks and moves his head.

Video segmentation is the basis for many applications including video representation, compression, tracking, motion analysis, object recognition and dynamic scene analysis and visualisation.

In this paper, we seek to obtain a spatio-temporal over-segmentation of the video, called *tubes*. The proposed tubes will encode a texture-based and temporal coherent segmentation of the regions in the video that respect boundaries and allow large displacements. It is able to deal with the significant region appearance changes that may happen over the frames due to the suffered perspective distortions caused by changes in the camera viewpoint or object motion. We attain this by, firstly, analysing the video using appropriate varying Riemannian metrics defined on the video domain that capture those distortions and provide adaptive spatio-temporal neighborhoods or patches. Secondly, we use an appropriate video similarity measure which uses these metrics to automatically transform the patches to compare them in an affine invariant manner. These tools are incorporated in our proposal for spatio-temporal tube computation which grows a tube by relevant patch comparison.

 ISBN 978-0-9934207-2-6

The remainder of the paper is organized as follows. In section 2 we revise previous work in supervoxel segmentation and video segmentation. Section 3 presents the proposed approach. Section 4 explains our algorithm and the most important implementation details while some results are presented in Section 5. Finally Section 6 concludes the paper and discusses future work.

2 State of the Art

Superpixel segmentation is an over-segmentation technique that simplifies an input image by grouping together into regions pixels following a similarity criterion. The output data has more compact and meaningful information than a single pixel and can be used as a preprocessed input data for several algorithms. Supervoxels extend the notion of superpixels along frames or over a volumetric object giving a 3-dimensional segmented region, also called *tube*, which aims to represent the trajectory in the scene of each moving object [Grundmann et al., 2010, Lezama et al., 2011, Trichet and Nevatia, 2013, Brendel and Todorovic, 2009]. The main application for supervoxels is video segmentation, although it has also been used for many others including video representation, compression or recognition.

Video segmentation can be associated with labeling or tracking an object or region over time. Some authors addressed video segmentation via analyzing each frame separately. First, individuals frames are segmented independently (superpixel extraction) and second, regions from frame to frame are matched. To find the same region over several frames, the optical flow [Trichet and Nevatia, 2013] or a distance function can be used [Oneata et al., 2014, Kwak et al., 2015]. This is a challenging approach, due to the changes in illumination, point of view, rotation, etc, that can suffer each region from one frame to another, and often gives an inconsistent segmentation. In addition, the estimation of the optical flow to subsequently segment the sequence is seen as a chicken-egg problem. A solution proposed by [Cremers, 2007] is to jointly solve the segmentation and motion estimation problems by minimizing a single functional in a Bayesian inference framework. Other authors use deep learning to face the problem. In [Khoreva et al., 2016], the algorithm learns how to refine the detected masks frame by frame, by using the detection of the previous frame together with optical flow and post-processing with Conditional Random Fields. In [Jampani et al., 2016], the training of Convolutional Neural Networks (CNN) is combined with bilateral filtering. In [Caelles et al., 2016], each frame is segmented independently using a fully CNN architecture using as a input the mask of the first frame.

Last, it is important to remark, that most of the tracking scenarios often use bounding boxes that does not adapts to the contour of the object [Nam and Han, 2016, Hare et al., 2016, Zhong et al., 2012]. Supervoxels, by contrast, seek to adapt the region to the boundary of the object.

3 Proposed model

Video segmentation methods aim at characterizing the spatio-temporal regions of the video domain having homogeneous texture regardless of differences in the point of view or suffered perspective distortion due to the movement and interactions of the objects. To this goal we propose to first analyse and describe the video by using appropriate spatially varying Riemannian metrics defined on the video domain that capture those distortions and provide adaptive spatio-temporal neighborhoods or patches. Morever, we use an appropriate similarity measure which uses these metrics to automatically transform the video patches to compare them in an affine invariant manner. Finally, we propose a method for video simplification and spatio-temporal tube computation. Sections 3.1, 3.2 and 3.3 present, respectively, each of the three ingredients of our proposal.

3.1 A video metric given by appropriate structure tensors

Given a video $u(\mathbf{x}, t)$ defined on a video domain $\mathscr{M} = \{(\mathbf{x}, t) : \mathbf{x} \in \Omega \subset \mathbb{R}^2, t \in \mathbb{R}\}$, our first aim is to endow the video domain $\mathscr{M}$ with a suitable metric depending on the video sequence which incorporates the temporal distances along the visible trajectory of the object pixels and also edge preserving anisotropies. Trajectories are

 ISBN 978-0-9934207-2-6

defined by the dense optical flow, that is, a vector field that recovers the apparent motion of two consecutive frames. We build on the video metrics that appeared in [Calderero and Caselles, 2014] for multiscale analyses classification purposes. Structure tensors were identified there as the natural metric obtained by considering the image (or video) as a manifold of patches. In the image processing and computer vision literature, the structure tensor, also referred as the second-moment matrix, has been considered as a metric in the image domain [Brox et al., 2006], [Peyré, 2009], [Weickert, 1998], [Fedorov et al., 2015]. It contains information about the predominant directions of the gradient in a specific neighborhood of a point.

In our work, we stem from a video structure tensor definition and propose an iterative method that enforces affine covariance. Those approximate affine covariant structure tensors will be intrinsically endowed with affine covariant neighborhoods providing adaptive space and time patches. Affine invariance, viewed as a simplified projective invariance, is an essential requirement for the analysis of natural scenes. In the video setting, the affine covariance is again an interesting property: if we turn a camera around its axis with a fixed angle, or if we change the format of the images, we expect to identify the same similarities among the moving objects (up to the discretization problems) [Guichard, 1998].

Let us recall that, in general, if $u : \mathbb{R}^N \to \mathbb{R}$ is any given image defined on $\mathbb{R}^N$, an image-dependent tensor field T_u is defined as a function that associates to each point $x \in \mathbb{R}^N$, a tensor, that is a symmetric, positive semi-definite $N \times N$ matrix $T_u(x)$. The tensor field is said to be *affine covariant* if for any affinity A given by a non-singular $N \times N$ matrix, T_u satisfies $T_{u_A}(x) = A^T T_u(Ax) A$, where $u_A(x) := u(Ax)$ denotes the affined transformed version of u. Several affine covariant structure tensors were proposed in [Fedorov et al., 2015]. Given a tensor $T_u(x)$ we can naturally associate to it an ellipsoidal or elliptical region of "radius" $r > 0$ centered at x

$$B_{T_u}(x, r) = \{y : \langle T_u(x)(y - x), (y - x)\rangle \leq r^2\}. \tag{1}$$

When the tensor is affine covariant, we have $AB_{T_u(x,r)} = B_u(Ax, r)$. In other words, the associated regions are affine covariant; they geometrically transform appropriately via an affinity. We refer to these regions as *shape-adaptive* or *ellipsoidal patches*. Figure 1 shows and example for the video case ($N = 3$) for the video structure tensor proposed below. For a point $\mathbf{x}$ close to the helmet boundaries in the *Foreman* sequence, the corresponding shape-adaptive patch is displayed in red.

To present the structure tensor we propose to use, let us denote by $\mathbf{v}$ the dense optical flow of the input video u. That is, $\mathbf{v}(\mathbf{x}, t)$ denotes the apparent motion between a pixel $\mathbf{x}$ on frame at time t and the corresponding at time $t + 1$, for $(\mathbf{x}, t) \in \mathscr{M}$. Let us consider the following video structure tensor [Calderero and Caselles, 2014]

$$T_u(\mathbf{x}, t) = \int_{\mathscr{E}_g((\mathbf{x},t),r)} D_{\mathbf{x}t} u((\mathbf{x}, t) + (\mathbf{y}, \tau)) \otimes D_{\mathbf{x}t} u((\mathbf{x}, t) + (\mathbf{y}, \tau)) \mu(\mathbf{y}, \tau) |G|^{1/2} \, d\mathbf{y} d\tau. \tag{2}$$

where $\mathscr{E}_g((\mathbf{x}, t), r) = \left\{Y = (\mathbf{y}, s) : g_{ij}(\mathbf{x}, t) Y^i Y^j \leq r^2\right\}$, g is an initial metric on $\mathscr{M}$, $|G|$ denotes the determinant of the symmetric matrix $G = (g_{ij})$ and μ is a weight measure on $\mathscr{E}_g((\mathbf{x}, t), r)$, be either the usual Lebesgue measure or a weighting function. Our initial metric g is given by

$$g(\mathbf{x}, t)\big((\mathbf{y}, \tau), (\mathbf{y}, \tau)\big) = a(\mathbf{x}, t)(\mathbf{y} - \mathbf{v}(\mathbf{x}, t)\tau)^2 + b(\mathbf{x}, t)\tau^2 \quad \text{(equivalently, } G(\mathbf{x}, t) = \begin{pmatrix} aI & -a\mathbf{v} \\ (-a\mathbf{v})^t & b + a|\mathbf{v}|^2 \end{pmatrix}(\mathbf{x}, t)\text{)}, \tag{3}$$

where $(\mathbf{y}, \tau)$ denote coordinates in an infinitesimal neighborhood of $(\mathbf{x}, t)$ and the functions a and b are defined as $a(\mathbf{x}, t) = \alpha_1 + \alpha_2 |\nabla_{\mathbf{x}} u|^p$, $b(\mathbf{x}, t) = \beta_1 + \beta_2 (\partial_{\mathbf{v}} u)^p$, with $\alpha_1, \alpha_2, \beta_1, \beta_2 > 0$, $p > 0$ (usually $p = 1, 2$), and $\partial_v u = \mathbf{v} \cdot \nabla_x u + u_t$ denotes the convective derivative. The meaning of the metric g is that being at the point $(\mathbf{x}, t)$ of a video a displacement $(\mathbf{y}, \tau)$ of a point is not penalized if it follows the law of motion at $(\mathbf{x}, t)$. g was tested in [Calderero and Caselles, 2014] in the context of video filtering.

The video structure tensor $T_u(\mathbf{x}, t)$ in (2) is computed on the spatio-temporal neighborhood $(\mathbf{x}, t) + \mathscr{E}_G(\mathbf{x}, t)$, which was proven to be equivalent infinitesimally to a motion compensated patch of radius r. We compute it using the compensated video. We propose to enforce affine covariance by an iterative scheme similar to the one used in [Fedorov et al., 2015] for the 2D case. In our case, the computation begins with the initial video region given by the initial metric G and proceeds with alternating computation of the structure tensor $T_u^k(\mathbf{x}, t)$

 ISBN 978-0-9934207-2-6

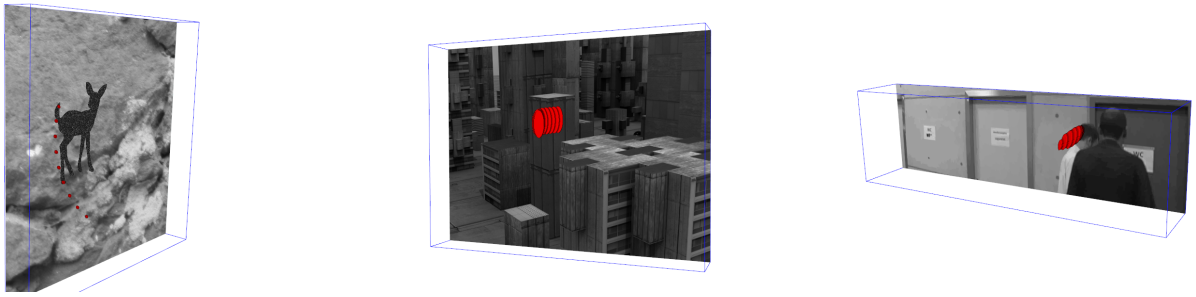


Figure 2: From left to right: Visualization of 3D-shape adaptive patches (in red) corresponding to the tensor on a point on the tail of the dear, on the wall of a building with vertical lines/texture and on the man's head (more details on the sequence movement are given in the text).

and its corresponding region $B_{T_u^k}((\mathbf{x},t),r)$ defined as in (1), where $T_u^k(\mathbf{x},t)$ denotes the structure tensor (2) computed on $B_{T_u^{k-1}}((\mathbf{x},t),r)$, normalized by its volume. In Section 4, we give the details of our algorithm for this $k-$iterative computation. Figure 2 shows some examples of adaptive patches in video. Let us notice, e.g., that the patch *follows the motion* (the deer rotates in the left-most sequence -see also Figure 5- and the camera moves in the *Urban2 Middlebury* sequence in the middle) or *stops* and adapts its shape when there are some occlusions in time (like in the right-most example). Let us remark that r is a free parameter which controls its size. Nevertheless, the size of each of them is also affected by the texture in the vicinity of the corresponding point allowing to an automatic and intrinsic adaptation.

3.2 A video similarity measure

Using affine covariant structure tensors as Riemannian metrics in $\mathbb{R}^N$, explicit formulas for multiscale affine invariant similarity measures were obtained in [Fedorov et al., 2015]. The patch similarity used in our work is an adaptation to the video case of one of them. One of the key points is to exploit the fact that the Riemannian metrics defined on the center point of the two patches in comparison (say $\mathbf{x}$ and $\mathbf{y}$) allow to automatically transform the patches to compare them in an appropriate manner. Moreover, if the local vicinities of $\mathbf{x}$ and $\mathbf{y}$ are related by a local affininity A such that $\mathbf{y} = A\mathbf{x}$, it was shown in [Fedorov et al., 2015] that from the affine covariant structure tensors we can recover the affine distortion up to a rotation. Indeed, $A = T_u(A\mathbf{x})^{\frac{-1}{2}} R T_{u_A}(\mathbf{x})^{\frac{1}{2}}$ for some orthogonal transformation R. This formula is at the basis of the affine invariant multiscale similarity measures studied there. It involves an intuitive idea: In order to compare the image (or video) content on patches centered at points $\mathbf{x}$ and $\mathbf{y}$, we can proceed in three steps: First by applying $T_u(\mathbf{x})^{\frac{1}{2}}$ the shape-adaptive patch, given by (1), is transformed to a disc (or sphere) of radius r. The resulting patch is called *normalized patch*. Second, we rotate the normalized patch using R. Finally, $T_u(\mathbf{y})^{\frac{-1}{2}}$ maps the rotated normalized disc to the shape-adaptive patch at $\mathbf{y}$. The rotation R is determined from the image content at the shape-adaptive patches and, in practice, is split into two parts such that $R = \tilde{R}(\mathbf{y})^{-1}\tilde{R}(\mathbf{x})$, where $\tilde{R}(\mathbf{x})$ and $\tilde{R}(\mathbf{y})$ depends on the image content around only one point, $\mathbf{x}$ or $\mathbf{y}$, respectively.

Let us now introduce the video patch similarity measure used in this work. We present it in the (slightly more general) context of comparing patches of two video sequences u_1 and u_2 defined on $\mathcal{M}_1 = \{(\mathbf{x},t) : \mathbf{x} \in \Omega_1,\, t \in \mathbb{R}\}$ and $\mathcal{M}_2 = \{(\mathbf{x},t) : \mathbf{x} \in \Omega_2,\, t \in \mathbb{R}\}$, respectively. Obviously, we might well have $u_1 = u_2$. Let $T_1(\mathbf{x}_1,t_1)$ be the structure tensor of u_1 at $(\mathbf{x}_{u_1},t_1)$ defined as in previous Section 3.1, for each point $(\mathbf{x}_{u_1},t_1)$ of the video domain $\mathcal{M}_1$. Analogously, let $T_{u_2}(\mathbf{x}_2,t_2)$ defined as in Section 3.1 be the structure tensor of u_2 at $(\mathbf{x}_2,t_2)$. These structure tensor fields provide metrics on $\mathcal{M}_1$ and $\mathcal{M}_2$, respectively. Then, the patch distance to compare the patches of u_1 and u_2 at $(\mathbf{x}_1,t_1)$ and $(\mathbf{x}_2,t_2)$ is defined by

$$\mathscr{D}((\mathbf{x}_1,t_1),(\mathbf{x}_2,t_2),s) = \int_{\Delta_s} g_s(y,\tau)\left(u_1\left((\mathbf{x}_1,t_1)+T_1(\mathbf{x}_1,t_1)^{-\frac{1}{2}}R_{u_1}^{-1}(\mathbf{x}_1,t_1)(y,\tau)\right)-u_2\left((\mathbf{x}_2,t_2)+T_2(\mathbf{x}_2,t_2)^{-\frac{1}{2}}R_{u_2}^{-1}(\mathbf{x}_2,t_2)(y,\tau)\right)\right)^2 dyd\tau. \quad (4)$$

where s is the so-called *scale*, g_s is a weighting function (for instance a Gaussian of variance s) and Δ_s denotes

 ISBN 978-0-9934207-2-6

a sphere centered at the origin with radius proportional to s and big enough such that g_s has effective support in Δ_s. Let us give a geometrical interpretation of (4): The two shape-adaptive ellipsoidal patches at $(\mathbf{x}_1, t_1)$ and $(\mathbf{x}_2, t_2)$ are first normalized by $R_{u_1} T_1(\mathbf{x}_1, t_1)^{\frac{1}{2}}$ and $R_{u_2} T_2(\mathbf{x}_2, t_2)^{\frac{1}{2}}$, respectively, to spheres of the same size and then compared. Section 4 presents the implementation details.

3.3 Proposed method for spatio-temporal tube computaion

Our tube computation algorithm is based on grouping together all the pixels with similar spatio-temporal local structure. To compute the similarity between two points $(\mathbf{x}_1, t_1)$ and $(\mathbf{x}_2, t_2)$ in the domain of a given video u with optical flow $\mathbf{v}$, we propose to use the patch similarity distance (4) between the shape-adaptive patches defined by the proposed approximate affine covariant structure tensors $T_u(\mathbf{x}_1, t_1)$ and $T_u(\mathbf{x}_2, t_2)$ on those points, and also take into account the Euclidean distance between the corresponding optical flow at $(\mathbf{x}_1, t_1)$ and $(\mathbf{x}_2, t_2)$.

More precisely, given a point $(\mathbf{x}_1, t_1)$, in order to obtain the spatio-temporal tube of this point, we proceed by a region growing procedure where we iteratively check if the neighboring points $(\mathbf{x}_2, t_2)$ (in a neighborhood $U(x_1, t_1)$, see Section 4) belong to the same tube. A point $(\mathbf{x}_2, t_2)$ is considered as belonging to the tube of $(\mathbf{x}_1, t_1)$ if the distance

$$\widetilde{\mathscr{D}}\left((\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), s\right) = \mathscr{D}\left((\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), s\right) + \alpha \mathscr{D}_E\left(\mathbf{v}(\mathbf{x}_1, t_1), \mathbf{v}(\mathbf{x}_2, t_2)\right) \tag{5}$$

is below a given threshold $\epsilon > 0$, where $\alpha \geq 0$ is weight factor and s is the scale. We iteratively verify if the neighboring points of the new point of the tube belong to the tube. In the following section we give a summary of all the steps of our algorithm, and the most important details of it.

4 Algorithm

Given a video u with optical flow $\mathbf{v}$, firstly, the normalized video structure tensor for each point $(\mathbf{x}, t)$ is computed with the following iterative process, where $k > 0$ denotes the iteration:

$$NT_u^{(k)}(\mathbf{x}, t) = \frac{\int_{B_{NT_u^{(k-1)}}((\mathbf{x},t),r)} D_{\mathbf{x}t} u((\mathbf{x}, t) + (\mathbf{y}, \tau)) \otimes D_{\mathbf{x}t} u((\mathbf{x}, t) + (\mathbf{y}, \tau)) \mu(\mathbf{y}, \tau) |G|^{1/2} \, d(\mathbf{y}, \tau)}{\text{Volume}\left(B_{NT_u^{(k-1)}}((\mathbf{x}, t), r)\right)} \tag{6}$$

and

$$B_{NT_u^{(k)}}((\mathbf{x}, t), r) = \begin{cases} \left\{(\mathbf{y}, \tau) : \langle G((\mathbf{x}, t))\left((\mathbf{y}, \tau) - (\mathbf{x}, t)\right), \left((\mathbf{y}, \tau) - (\mathbf{x}, t)\right)\rangle \leq r^2\right\} & \text{when } k = 0 \\ \left\{(\mathbf{y}, \tau) : \langle NT_u^{(k)}(\mathbf{x}, t)\left((\mathbf{y}, \tau) - (\mathbf{x}, t)\right), \left((\mathbf{y}, \tau) - (\mathbf{x}, t)\right)\rangle \leq r^2\right\} & \text{when } k > 0 \end{cases} \tag{7}$$

where G is the initial metric defined in (3). The tensors are computed on the motion compensated neighborhood. Equations (6) and (7), constitute an iterative scheme that provide approximate affine covariant tensors together with their shape-adaptive neighborhoods at each point $(\mathbf{x}, t)$ in the video domain. To simplify the notation, we will denote by $T_u(\mathbf{x}, t)$ the structure tensor $NT_u^{(k)}(\mathbf{x}, t)$ for a fixed number of iterations k and a given value of r.

The algorithm to compute spatio-temporal tubes, each one labeled with a label L, is the following. For fixed parameters $\epsilon > 0$, $1 > \alpha \geq 0$ and $s > 0$, let $(\mathbf{x}_0, t_0)$ be a given seed point in the video domain. Then,

1. Set $(\mathbf{x}_1, t_1) = (\mathbf{x}_0, t_0)$.
2. If $(\mathbf{x}_1, t_1)$ has no label, label it with a *new* label L.
3. Consider all the 26-connected pixels of $(\mathbf{x}_1, t_1)$ **in the motion compensated neighborhood**. We denote by $U(\mathbf{x}_1, t_1)$ this neighborhood.
4. For each point $(\mathbf{x}_2, t_2) \in U(\mathbf{x}_1, t_1)$ which has no label, compute $\mathscr{D}\left(((\mathbf{x}_1, t_1), ((\mathbf{x}_2, t_2), s)\right.$ given by (4) (see details below) and the Euclidean distance $\mathscr{D}_E\left(\mathbf{v}(\mathbf{x}_1, t_1), \mathbf{v}(\mathbf{x}_2, t_2)\right)$.
5. If $\widetilde{\mathscr{D}}\left((\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), s\right) < \epsilon$, then label $(\mathbf{x}_2, t_2)$ with the same label L as $(\mathbf{x}_1, t_1)$.
6. Set $(\mathbf{x}_1, t_1) = (\mathbf{x}_2, t_2)$ and iterate steps 2 to 6.

Figure 3 illustrate the main steps of the proposed algorithm. Finally, let us add some details on the computation of the comparison measure (4) in step 3. We only need the corresponding structure tensors,

 ISBN 978-0-9934207-2-6

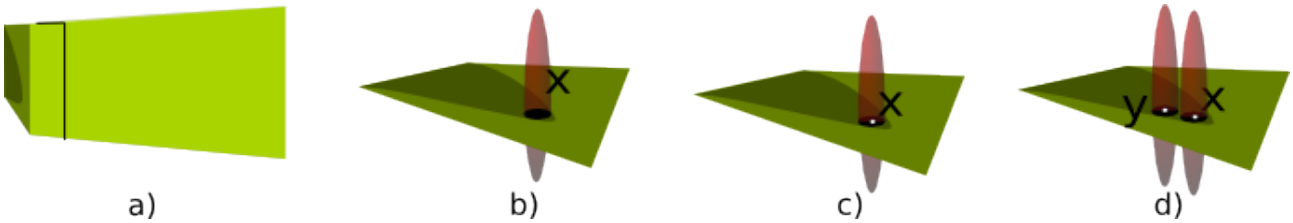


Figure 3: Diagram illustrating the main steps of the algorithm: a) video sequence; b) selected point $\mathbf{x} = (x_1, y_1, t_1)$, c) As $\mathbf{x}$ has no label, label it with a new label L = white; d) Compute the distance between $\mathbf{x}$ and $\mathbf{y} = (x_2, y_2, t_2) \in U(\mathbf{x})$; as they are similar, label $\mathbf{y}$ with same label as $\mathbf{x}$. These steps are performed iteratively.

$T_u(\mathbf{x}_1, t_1)$ and $T_u(\mathbf{x}_2, t_2)$, and the additional orthogonal transformation R which, as explained above, is split as $R = R_u^{-1}(\mathbf{x}_2, t_2)R_u(\mathbf{x}_1, t_1)$. $R_u(\mathbf{x}_1, t_1)$ and $R_u(\mathbf{x}_2, t_2)$ are computed by estimating dominant orientations of the gradient of u in the normalized patches of $(\mathbf{x}_1, t_1)$ and $(\mathbf{x}_2, t_2)$, respectively (i.e., the spheres obtained applying $T_u^{\frac{1}{2}}$) using weighted histograms of gradient orientations. This step is crucial since we align the dominant orientation of both patches to be able to compare them.

5 Experimental Results

Experiments on synthetics and real data sets are presented in this section. Each input sequence has its own resolution size and frame rate, which also helps us to test the robustness of the proposed approach based on video metrics and affine covariance properties to discrete videos. For each sequence, a tube corresponding to a fixed seed point $(\mathbf{x}, t)$ will be computed. Although the presented algorithm provides a supervoxel or tube representation of the full sequence, for simplicity and visualization purposes we will only present, for each sequence, the tube corresponding to a unique seed point $(\mathbf{x}_0, t_0)$.

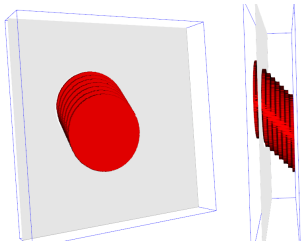


Figure 4: Tube computation from a point inside of the disc that moves, each frame, 10 pixels to the bottom.

5.1 Experiments using synthetic sequences

For the tests, synthetic sequences with a ground-truth optical flow have been created. We work with two types of synthetic sequences where a *disc* and a *deer* are the moving objects in the scene. To create consecutive frames successive translations and rotations are applied to the original frame. Figure 4 shows the tube resulting from a seed point inside of the disc which is translating downwards, while Figure 5 shows the tube of a point inside of the deer which is rotating. We can easily notice that the tube follows perfectly the trajectory of the disc and the deer, respectively, and fits its boundaries regardless of their long trajectories.

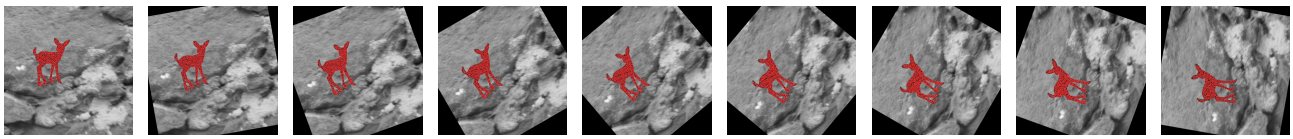


Figure 5: Tube computation starting from a point inside of a deer in a sequence obtained by rotating each frame an angle of 10 degrees counterclockwise.

5.2 Experiments using real sequences

In this section, we show the performance on our algorithm in well-known sequences. The ground-truth optical flow is not available, so an optical flow has been computed between each pair of frames using [Sánchez et al.,

 ISBN 978-0-9934207-2-6

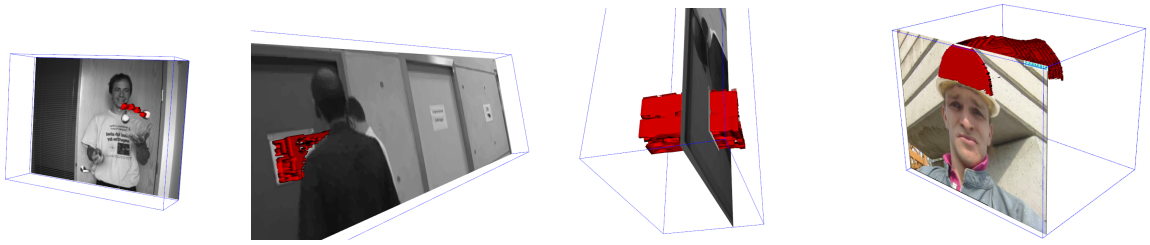


Figure 6: Spatio-temporal tube following a ball (left), the door sign (center) and the helmet (right).

2013]. Figure 6 shows, on the left, the tube corresponding to a seed point inside of the ball in the left hand of the *Beanbags Middlebury* sequence. Although the quality of the estimated optical flow is quite poor, our algorithm is able to follow the ball in the first four frames, losing the trajectory afterwards due to the disparity between the real motion and the optical flow values. In the second example, shown in the two middle images of Figure 6, the algorithm is able to separate the letters from the white background of the sign. In this sequence, the seed point is on the white part of the sign. Notice that is also able to adapt the tube once the sign is getting occluded by the man in the black t-shirt. The change in the shape or area of the tube from one frame to another, reflects the partial occlusions (when the area gets smaller) or disocclusion (when the area gets bigger). In other words, the proposed tubes can be used to estimate occlusion and disocclusions: The temporal boundaries of a tube give information about their birth and death. Lastly, in the right side of Figure 6, a tube computed from a seed point from the helmet of the *Foreman Middlebury* sequence is shown. In this sequence, the tube last along the frames, following the motion corresponding to the movement of the face and adapting the shape to the silhouette of the helmet.

6 Conclusion

We presented a video simplification approach which identifies moving or static objects in video and is able to group together all the pixels into a *tube* or *supervoxel*. Our method grounds on video metrics given by appropriate structure tensors and a video comparison measure. We have shown that, thanks to it, the tubes are recovered regardless of the perspective or affine transformation of the region over frames due to changes on point of view or object motion. We presented examples in synthetic and real sequences, demonstrating that the computed tubes adapt the shape to the boundaries of the region and follow the trajectory of the object.

Some of the problems of our algorithm is the sensitivity to similar color of neighboring regions, being sometimes a problem as it might lead a tube to grow outside the true region of the seed point. This could be solved via a variational optimization strategy building a functional measuring the quality of the decomposition of the original video into tubes with a fidelity data term based on our patch similarity measure over the whole video domain and a smoothness term measuring, for instance, the total area of the boundaries of the tubes. This solution would also achieve a complete partition of the video domain and thus a video segmentation solution.

Acknowledgments

All authors acknowledge partial support by MINECO/FEDER UE project, reference TIN2015-70410-C2-1-R and by GRC reference 2014 SGR 1301, Generalitat de Catalunya.

References

[Brendel and Todorovic, 2009] Brendel, W. and Todorovic, S. (2009). Video object segmentation by tracking regions. In *ICCV 2009*, pages 833–840.

[Brox et al., 2006] Brox, T., Boomgaard, R., Lauze, F., Weijer, J., Weickert, J., Mrázek, P., and Kornprobst, P. (2006). Adaptive structure tensors and their applications. *Visual. and Proces. of Tensor Fields*, pages 17–47.

[Caelles et al., 2016] Caelles, S., Maninis, K.-K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., and Van Gool, L. (2016). One-shot video object segmentation. *arXiv preprint arXiv:1611.05198*.

[Calderero and Caselles, 2014] Calderero, F. and Caselles, V. (2014). Multiscale analysis of images on riemannian manifolds. *SIAM Journal Imaging Sciences*, 7(2):1108–1170.

[Cremers, 2007] Cremers, D. (2007). Bayesian approaches to motion-based image and video segmentation. In *Complex Motion*, pages 104–123. Springer.

[Fedorov et al., 2015] Fedorov, V., Arias, P., Sadek, R., Facciolo, G., and Ballester, C. (2015). Linear multiscale analysis of similarities between images on riemannian manifolds: Practical formula and affine covariant metrics. *SIAM Journal on Imaging Sciences*, 8(3):2021–2069.

[Grundmann et al., 2010] Grundmann, M., Kwatra, V., Han, M., and Essa, I. (2010). Efficient hierarchical graph-based video segmentation. In *CVPR 2010*, pages 2141–2148.

[Guichard, 1998] Guichard, F. (1998). A morphological, affine, and galilean invariant scale-space for movies. *Image Processing, IEEE Transactions on*, 7(3):444–456.

[Hare et al., 2016] Hare, S., Golodetz, S., Saffari, A., Vineet, V., Cheng, M.-M., Hicks, S. L., and Torr, P. H. (2016). Struck: Structured output tracking with kernels. *IEEE-PAMI*, 38(10):2096–2109.

[Jampani et al., 2016] Jampani, V., Gadde, R., and Gehler, P. V. (2016). Video propagation networks. *arXiv preprint arXiv:1612.05478*.

[Khoreva et al., 2016] Khoreva, A., Perazzi, F., Benenson, R., Schiele, B., and Sorkine-Hornung, A. (2016). Learning video object segmentation from static images. *arXiv preprint arXiv:1612.02646*.

[Kwak et al., 2015] Kwak, S., Cho, M., Laptev, I., Ponce, J., and Schmid, C. (2015). Unsupervised object discovery and tracking in video collections. In *ICCV*, pages 3173–3181.

[Lezama et al., 2011] Lezama, J., Alahari, K., Sivic, J., and Laptev, I. (2011). Track to the future: Spatio-temporal video segmentation with long-range motion cues. In *CVPR 2011*, pages 3369–3376.

[Nam and Han, 2016] Nam, H. and Han, B. (2016). Learning multi-domain convolutional neural networks for visual tracking. In *CVPR 2016*, pages 4293–4302.

[Oneata et al., 2014] Oneata, D., Revaud, J., Verbeek, J., and Schmid, C. (2014). Spatio-temporal object detection proposals. In *European conference on computer vision*, pages 737–752.

[Peyré, 2009] Peyré, G. (2009). Manifold models for signals and images. *Computer Vision and Image Understanding*, 113(2):249–260.

[Sánchez et al., 2013] Sánchez, J., Meinhardt-Llopis, E., and Facciolo, G. (2013). TV-L1 Optical Flow Estimation. *Image Processing On Line*, 3:137–150.

[Trichet and Nevatia, 2013] Trichet, R. and Nevatia, R. (2013). Video segmentation with spatio-temporal tubes. In *AVSS 2013*, pages 330–335.

[Weickert, 1998] Weickert, J. (1998). *Anisotropic diffusion in image processing*, volume 1. Teubner Stuttgart.

[Zhong et al., 2012] Zhong, W., Lu, H., and Yang, M.-H. (2012). Robust object tracking via sparsity-based collaborative model. In *CVPR, 2012*, pages 1838–1845.

 ISBN 978-0-9934207-2-6

Computing the Uncertainty of Motion Fields for Human Activity Analysis*

Jorge S. Marques António R. Moreira João M. Lemos

INESC-ID
Institute for Systems and Robotics (ISR/IST), LARSyS,
Instituto Superior Técnico, University of Lisboa, Portugal.

Abstract

The motion of pedestrians in outdoor scenes has been represented using motion fields that characterize typical pedestrian velocities at each position in the image. Several methods have been proposed to estimate the motion fields parameters from a set of pedestrians trajectories detected in the video signal. However, up to now no attempts have been made to evaluate the uncertainty associated to the estimates. In fact, the estimated fields are usually used as if they were accurate at all pedestrians positions and this is not a reasonable assumption. This paper is a first attempt to calculate the uncertainty of the estimates as a function of the pedestrian position. We consider two estimation frameworks (ridge regression and Bayesian inference) and discuss uncertainty estimates provided by both of them. Experiments under controlled condition are performed to evaluate the merits of these approaches.

Keywords: pedestrian motion, vector fields, uncertainty estimation, regression, Bayesian methods

1 Introduction

The analysis of human activities in video sequences has been thoroughly studied in the last decade [Poppe, 2010], [Turaga et al., 2008]. Many of these works are focused in indoor activities, *e.g.*, the activity of people inside a house [Lara & Labrador, 2013]. However, other works address human motion in wide spaces such as parks or university campi, in order to understand how people explore the space and what kind of activities are performed: which movements are typical and which are rare or abnormal [Aggarwal & Ryoo, 2011]. Many of these models can also be applied to characterize the motion of other objects in the scene (*e.g.*, vehicles [Melo et al., 2006], or animals [Beyan et al., 2013, Poiesi & Cavallaro, 2015]).

One way to characterize human motion consists of extracting the trajectories of pedestrians in a scene and clustering them [Hu et al., 2006, Ferreira et al., 2013, Pires & Figueiredo, 2017], or representing them as the output of dynamical models [Porikli, 2004, Nascimento et al., 2010]. A recent trend describes pedestrian trajectories using a bank of motion fields [Nascimento et al., 2013]. These models are equipped with a switching mechanism in order to allow the switching between different motion regimes. Model learning algorithms have been proposed to estimate the motion fields from video data. O However, up to now no attempt has been made to characterize the uncertainty associated to the motion field estimates.

This paper is a first attempt to address this problem. This question is an important issue since it will provide measures of confidence on the field estimates that may depend on the position of the pedestrian in the scene and the amount of information that was collected in the neighborhood of that position. In addition, this confidence measure may allow the design of new learning strategies targeted at regions in which the confidence is lower and needs to be enhanced.

*Work supported by FCT under contracts PTDC/EEIPRO/0426/2014, UID/CEC/50021/2013 and UID/EEA/50009/2013.

The paper is organized as follows. This section provides an introduction to the problem. Section 2 addresses the motion field model including the motion field representation. Section 3 discusses motion field estimation and uncertainty measurement. Section 4 describes some experimental results and section 5 concludes the paper.

2 Velocity model

For the sake of simplicity, we will consider that all the pedestrians trajectories are generated by a dynamical model with a single motion field

$$x_{t+1} = x_t + T(x_t) + u_t \,, \tag{1}$$

where $x_t \in [0,1]^2$ denotes the position of the pedestrian in the image at time t ($[0,1]^2$ stands for the image plane), $T(x) \in \mathbb{R}^2$ denotes the motion field at position x and u_t is a realization of a white noise process, with Gaussian distribution $u_t \sim N(0, \sigma^2 I)$. The motion field T is unknown and we want to estimate it from the pedestrian trajectories extracted from the video data.

We will also assume that we tracked S pedestrian trajectories, $(x_1^{(s)}, x_2^{(s)}, \ldots, x_{m^{(s)}}^{(s)})$, with $s = 1, \ldots, S$, and that these trajectories are associated to a set of *position/velocity* pairs

$$\mathcal{T} = \{(x_i, v_i), i = 1, \ldots, L\} \,,$$

where $x_i \in [0,1]^2$ stands for the ith position of the pedestrian and $v_i = x_{i+1} - x_i \in \mathbb{R}^2$ is the corresponding velocity (displacement). We have excluded from $\mathcal{T}$ the last position of each trajectory since we cannot compute the displacement in such cases.

Let us assume that the motion field $T : [0,1]^2 \to \mathbb{R}^2$ to be estimated is defined on a $n \times n$ grid of nodes regularly distributed in the image, as proposed in [Nascimento et al., 2013]. The parameters to be estimated are the velocities $t^j \in \mathbb{R}^2$, $j = 1, \ldots, n^2$ at the grid nodes. If x denotes an arbitrary point in the image plane, the motion field at x will be obtained by interpolating the grid velocities

$$T(x) = \sum_{j=1}^{n^2} t^j \phi_j(x),$$

where $\phi_j(x)$ is the bilinear interpolation function associated to the j-th node (see [Nascimento et al., 2013] for details). This equation can be easily written in a simpler way, by using matrix notation

$$T(x) = \left[\begin{array}{cc|c|cc} \phi_1(x) & 0 & \ldots & \phi_{n^2}(x) & 0 \\ 0 & \phi_1(x) & \ldots & 0 & \phi_{n^2}(x) \end{array}\right] \begin{bmatrix} t^1 \\ \vdots \\ t^{n^2} \end{bmatrix} = \phi(x)\, T, \tag{2}$$

where T is a vector obtained by concatenating the velocities associated to all the grid nodes. If we assume that each velocity measured in the image, v_i, is corrupted by a random measurement noise, u_i, we can write

$$v_i = \phi(x_i)\, T + u_i \qquad i = 1, \ldots, L,$$

where, for the sake of simplicity, u_i is assumed to be a Gaussian random vector with zero mean and isotropic covariance matrix: $u_i \sim N(0, \sigma^2 I)$. The last equation can be written in a more elegant way if we aggregate all the observations into a single vector

$$\begin{bmatrix} v_1 \\ \vdots \\ v_L \end{bmatrix} = \begin{bmatrix} \phi(x_1) \\ \vdots \\ \phi(x_L) \end{bmatrix} T + \begin{bmatrix} u_1 \\ \vdots \\ u_L \end{bmatrix},$$

$$v = \Phi T + u \,,$$

where $v, u \in \mathbb{R}^{2L \times 1}, \Phi \in \mathbb{R}^{2L \times 2n^2}, T \in \mathbb{R}^{2n^2 \times 1}$. The conditional distribution of v, given T, is

$$v | T \sim N(\Phi T, \sigma^2 I) \,;$$

it is remarked that matrix Φ has a number of lines that equals the double of the number of data points.

 ISBN 978-0-9934207-2-6

3 Field and uncertainty estimation

3.1 Least squares and ridge regression

The simplest approach to estimate the field parameters is based on the minimization of a squared error (least squares) criterion

$$E_{\text{ls}} = \|v - \Phi T\|^2.$$

The least squares estimate of the grid node velocities is $\hat{T}_{\text{ls}} = (\Phi^T\Phi)^{-1}\Phi^T v$ and the covariance matrix associated with this estimate is [Hastie et al., 2009]

$$\text{Cov}\{\hat{T}_{\text{ls}}\} = \sigma^2(\Phi^T\Phi)^{-1}.$$

Both expressions require that the matrix $\Phi^T\Phi$ is non-singular. This is usually not true in this problem since there are regions in the image that are never crossed by a trajectory and for which there is no information available.

An alternative to this approach is ridge regression which includes a regularization term that attracts the motion estimates towards zero [Hastie et al., 2009]. The estimation of T given the observations v can be achieved by minimizing

$$E_{\text{ridge}} = \|v - \Phi T\|^2 + \lambda\|T\|^2,$$

where $\|\cdot\|$ denotes the ℓ_2 norm. In this case, the ridge estimate for the vector field is $\hat{T}_{\text{ridge}} = (\Phi^T\Phi + \lambda I)^{-1}\Phi^T v$ and the uncertainty is

$$\text{Cov}\{\hat{T}_{\text{ridge}}\} = \sigma^2(\Phi^T\Phi + \lambda I)^{-1}\Phi^T\Phi(\Phi^T\Phi + \lambda I)^{-1},$$

which can be easily computed. It is stressed that the inverse exists even if no trajectories are observed in the vicinity of a grid node or a set of grid nodes.

The final question is: what is the uncertainty associated to the velocity estimates at an arbitrary position $x \in [0,1]^2$? This can be answered using (2). The velocity estimate is

$$\hat{T}(x) = \phi(x)\hat{T},$$

and the covariance matrix is

$$\Sigma(x) = \phi(x)\text{Cov}\{\hat{T}\}\phi(x)^T. \tag{3}$$

Both can be computed for each point in the image. If we want to characterize the uncertainty by a single number, the trace of matrix $\Sigma(x)$ can be adopted.

3.2 Bayesian inference

Until now T is assumed to be an unknown deterministic variable. Let us assume now that T is a random vector with Gaussian *prior* distribution $T \sim N(T_0, P_0)$. In this case, the *a posteriori* distribution of T is also Gaussian

$$T|v \sim N(\hat{T}, P),$$

with mean vector, $\hat{T}$, and covariance matrix, P, given by the Kalman filter equations[Arulampalam et al., 2002]

$$\hat{T} = T_0 + K(v - \Phi T_0), \tag{4}$$

$$P = (I - K\Phi)P_0, \tag{5}$$

$$K = P_0\Phi^T\left(\Phi P_0\Phi^T + \sigma^2 I\right)^{-1}. \tag{6}$$

In this case, P is the covariance matrix of the state vector, T, after making the observations v, and describes the uncertainty of the motion field at the grid nodes. It should be stressed, however, that these equations involve the inversion of a matrix that may have a large dimension (twice the number of observations) which makes them more complex then the ones obtained for the ridge regression.

 ISBN 978-0-9934207-2-6

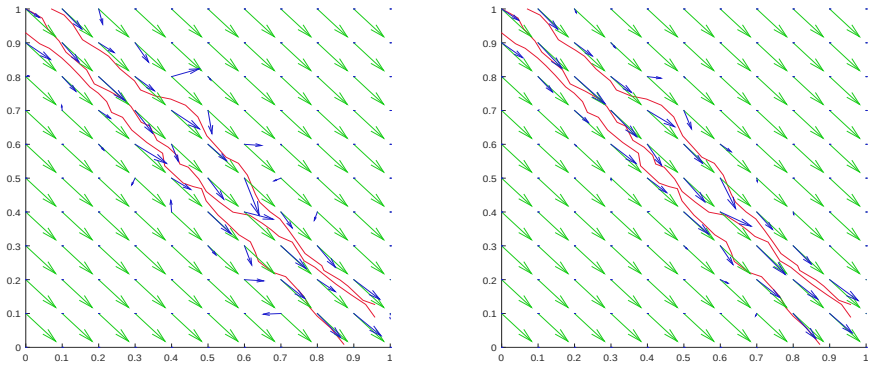


Figure 1: True motion field (green), synthetic trajectories (red), and estimated field (blue) for ridge regression (left) and Bayesian method (right)

4 Experimental results

We performed experiments with synthetic trajectories generated by equation (1) and with pedestrian trajectories extracted from a video signal. In the first case, we know the true motion field used to generate the trajectories. We can therefore compare the motion estimates with the ground truth information. We have also studied the effect of trajectory parameters (*e.g.*, number of trajectories, density) on the model uncertainty. In a third experiment we considered pedestrian trajectories extracted from a video signal, for which the true motion field is not known. All these experiments are carried out using a regular grid of 11×11 nodes, using ridge regression and Bayesian approach (Kalman filtering).

4.1 Bayesian vs ridge regression uncertainty

In the first experiment, we consider a uniform motion field and generated three random trajectories using the dynamic model (1). Then, we estimated the vector field from these trajectories using ridge regression and the Bayesian method under Gaussian hypothesis. Figure 1 shows the vector field used in these experiments (green), three trajectory realizations generated by the model (red) and the field estimates (blue) obtained by the ridge regression and Bayesian method.

We repeated this experiment N times ($N = 100$) and compared the velocity estimates, $\hat{t}_i^k$, obtained at each experiment k and grid node i, with the ground truth velocity vector, t_i^{gt}. The mean square error at the $i-th$ node,

$$E_i = \frac{1}{N} \sum_{k=1}^{N} \| \hat{t}_i^k - t_i^{gt} \|^2 , \tag{7}$$

is a measure of uncertainty. Figure 2 (1st row) shows the mean square error associated with both estimation methods.

This comparison is possible since we know the true value of the vector fields at the grid nodes (ground truth) but cannot be computed when we use real data since we do not have the true motion field in that case.

To circumvent this difficulty, we use the predicted variance associated to the ridge regression and Bayesian estimators (see (3,6)), displayed in Figure 2 (2nd line). They have different structures. The variance of the Bayesian estimates is high in all the regions that have not been visited by a trajectory. In such regions, there is no information available concerning pedestrian motion and the uncertainty is therefore very high since we do not know the direction of the pedestrian motion. On the contrary, the uncertainty is low in regions visited by trajectories, as expected.

The ridge regression performs, however, in a different way. Regions far from the trajectories have zero variance. This is counter intuitive but it is correct since the ridge regression assigns a zero value to the velocity estimates whenever there is no information. The variance of the estimates is therefore equal to zero in such

 ISBN 978-0-9934207-2-6

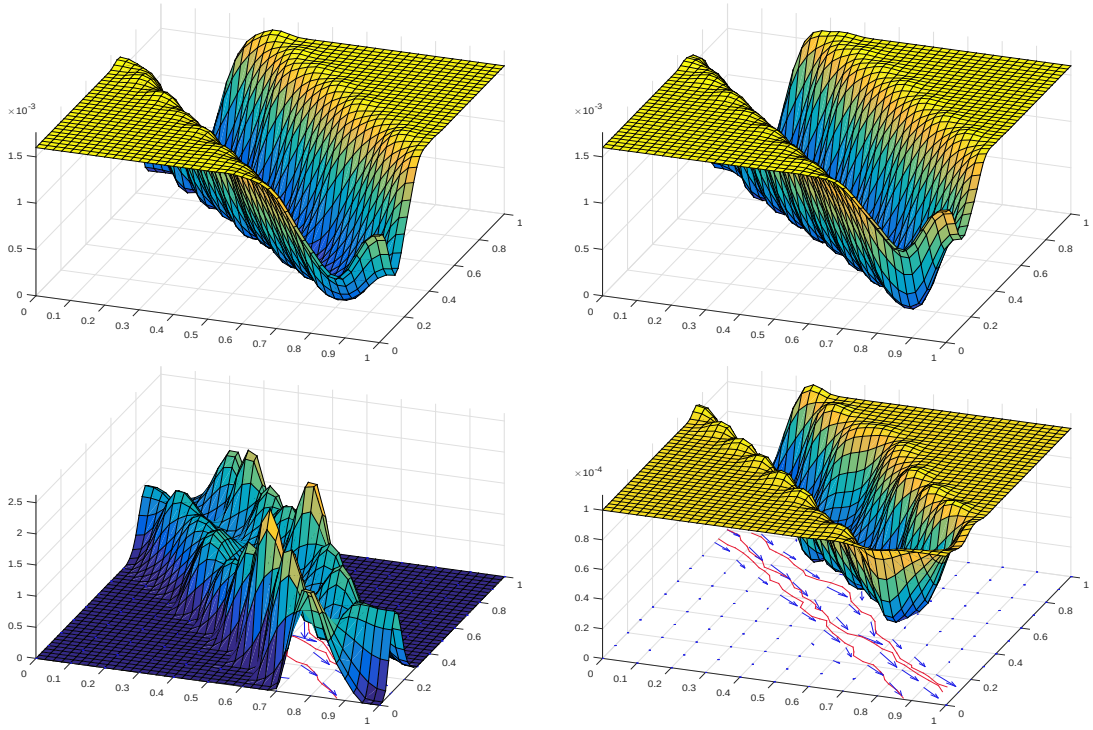


Figure 2: Mean square error estimation of the node velocities by Monte Carlo (1st line) and predicted variance (2nd line), for ridge regression (left) and Bayesian method (right). The Monte Carlo results (1st line) require the ground truth velocities while the variances (2nd line) do not.

cases. This does not mean that there is no error but simply that there is a strong bias associated to the motion estimate at each grid node and zero variance.

We conclude that the results obtained with the Bayesian approach are informative and a reliable measure of the uncertainty. On the contrary, the variance obtained for the ridge regression estimator does not provide a good quality measure because the uncertainty is dominated by a strong bias that cannot be computed from real data. Therefore, we will use the Bayesian approach from now on

4.2 Trajectory parameters

The motion field uncertainty is related to the amount of observations available in the vicinity of each image point. Taking the previous experiment as a reference (3 lines generated by a uniform field) we will i) change the number of trajectories, keeping the same average distance between them and ii) increase the density of trajectories. Typical examples are shown in Figs. 3,4.

The first example (Fig. 3) shows that the variance map changes (the valley gets wider) when the number of lines increases, as expected. The valley is made of points which are close to a trajectory where *close* means that the distance from the point to the trajectory is smaller than the grid step.

The second experiment keeps the width constant but increases the density of trajectories. In this case then valley of the variance map becomes deeper, as expected. This example shows that the uncertainty behaves in a simple and predictable way in this case.

4.3 Video surveillance

The main issue that remains to be tested is how does the uncertainty analysis behave with real data. The next experiment involves the estimation of a motion field from video data, in an outdoor scene (see Fig. 5). First we

 ISBN 978-0-9934207-2-6

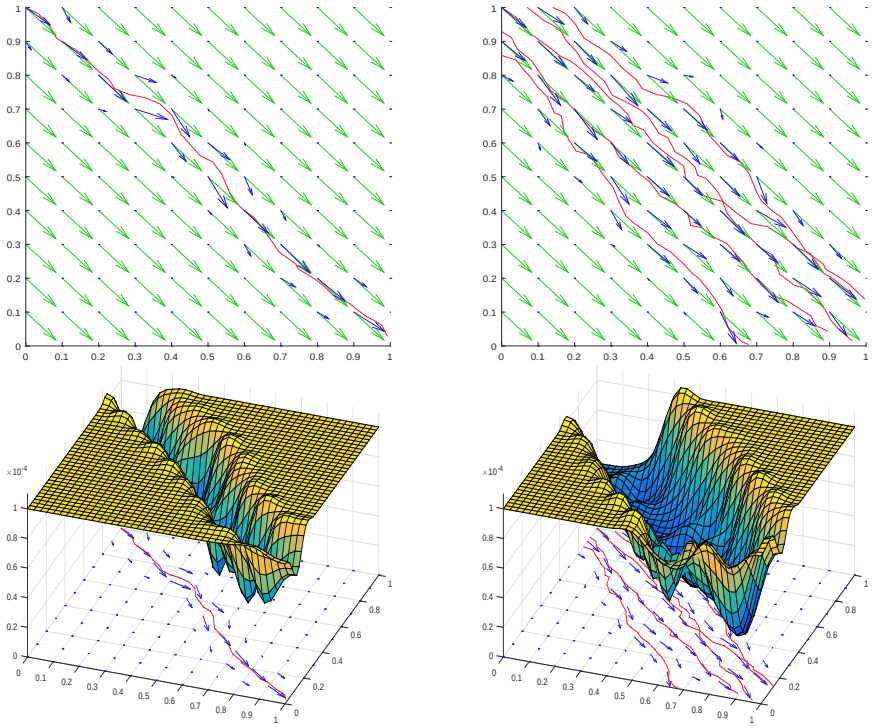


Figure 3: Estimated fields (1st line) and field variances (2nd line) obtained by the Bayesian method: 1 trajectory (left) and 5 equally spaced trajectories (right).

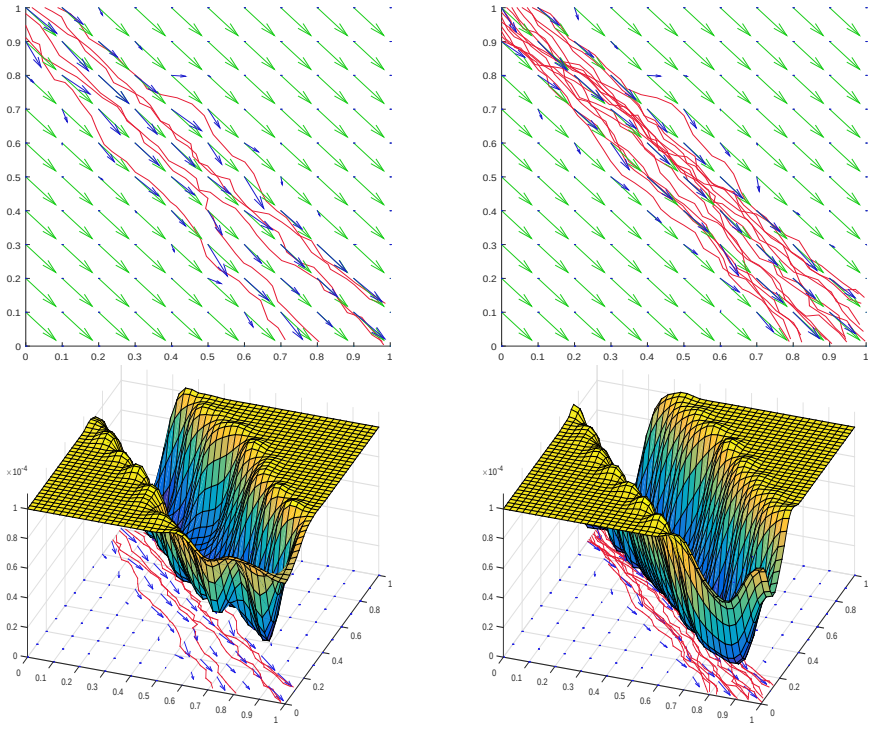


Figure 4: Estimated fields (1st line) and field variances (2nd line) obtained by the Bayesian method with different densities: 5 trajectory (left) and 15 trajectories (right).

 ISBN 978-0-9934207-2-6

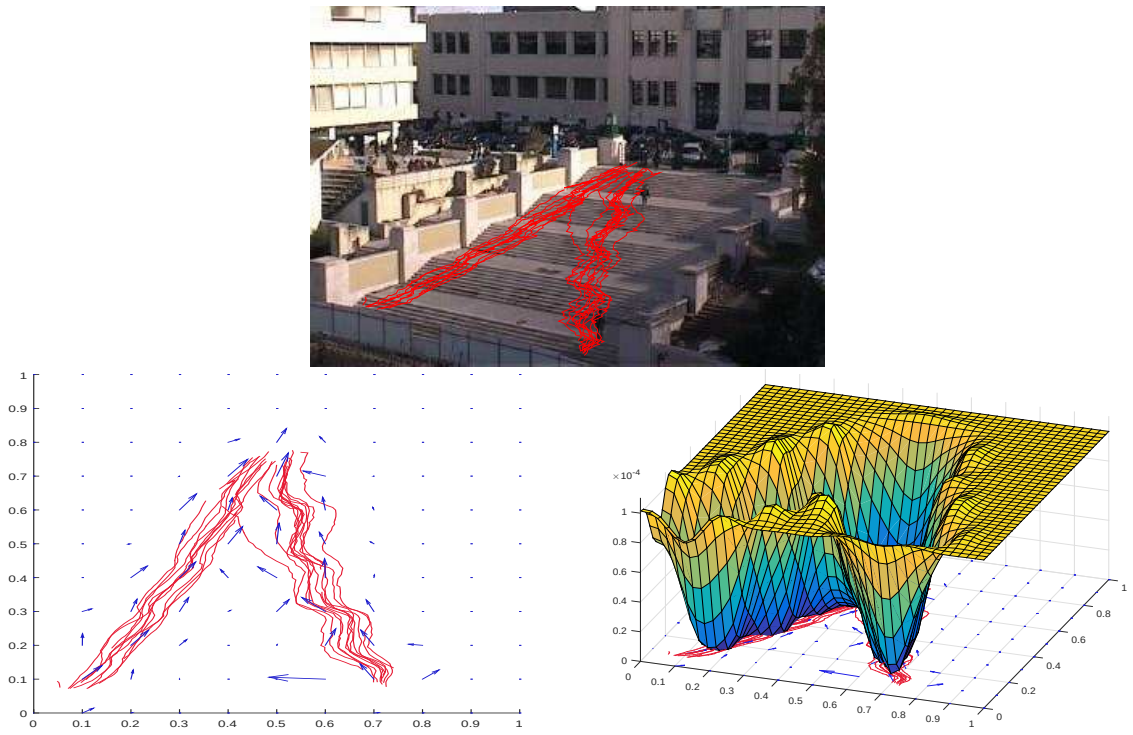


Figure 5: Staircase scene (1st line); 2nd line: pedestrians trajectories and estimated motion fields using the Bayesian method (left) and model variance (right).

extracted the pedestrians trajectories. The trajectories were then transformed by an homography to compensate for the perspective distortion. A motion field was estimated using the Kalman filter (6). Fig. 5 (2nd line left), shows the estimated field. There are some wrong estimates at nodes located far from the trajectories in which there is a large uncertainty. The variance map (Fig. 5 (2nd line right)) shows a small variance in regions close to the trajectories, as expected, and a large variance in regions that were not visited by trajectories.

This experiment shows that the uncertainty measures can be applied to pedestrian trajectories extracted from video data and produce acceptable results.

5 Conclusions

Several algorithms have been proposed to estimate motion fields from trajectories. However, the estimation of the uncertainty associated with the motion estimates is usually not addressed. This is an important issue since we must know what is the confidence associated to the velocity estimates in each position. It does not matter if we have velocity estimates in all image regions if the uncertainty is very high. Another important issue concerns data management. When we receive new trajectories we should be able to decide whether they should be accepted, in case they improve the motion field estimates, or if they should be rejected, in case there is already a lot of information in that region. This kind of decision can be taken, if we have uncertainty estimates. Furthermore, estimating the uncertainty provides a way to address the problem of finding the number and space distribution of trajectories that allows to estimate the field in the whole image with a maximum error bound.

This paper discusses two ways to estimate uncertainty. The one that achieves the best results is the Bayesian method, based on the Kalman update equations. The main conclusion can be loosely stated as follows: uncertainty is low at image points x such that the vicinity of x is crossed by a sufficient number of trajectories.

Future work should address the application of motion uncertainty in the management of new data, by defining criteria to accept new data for motion field update. The extension of these results to multiple motion

 ISBN 978-0-9934207-2-6

field models should also be considered and it is a challenging problem since the Gaussian hypothesis is no longer valid.

References

[Aggarwal & Ryoo, 2011] Aggarwal, J. & Ryoo, M. (2011). Human activity analysis: A review. *ACM Computing Surveys*.

[Arulampalam et al., 2002] Arulampalam, M., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on signal processing*, (pp. 174–188).

[Beyan et al., 2013] Beyan, C., Cigdem, & Fisher, R. (2013). Detection of abnormal fish trajectories using a clustering based hierarchical classifier. In *BMVC*.

[Ferreira et al., 2013] Ferreira, N., Silva, C., Klosowski, J., & Scheidegger, C. (2013). Vector field k-means: Clustering trajectories by fitting multiple vector fields. *Computer Graphics Forum*.

[Hastie et al., 2009] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning*. Springer.

[Hu et al., 2006] Hu, W., Xiao, X., Fu, Z., Xie, D., Tan, T., & Maybank, S. (2006). A system for learning statistical motion patterns. *IEEE transactions on pattern analysis and machine intelligence*, (pp. 1450–1464).

[Lara & Labrador, 2013] Lara, O. & Labrador, M. (2013). A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys and Tutorials*, (pp. 1192–1209).

[Melo et al., 2006] Melo, J., Naftel, A., Bernardino, A., & Santos-Victor, J. (2006). Detection and classification of highway lanes using vehicle motion trajectories. *IEEE Transactions on intelligent transportation systems*, (pp. 188–200).

[Nascimento et al., 2010] Nascimento, J. C., Figueiredo, M. A. T., & Marques, J. S. (2010). Trajectory classification using switched dynamical hidden markov models. *IEEE Transactions on Image Processing*, (pp. 1338–1348).

[Nascimento et al., 2013] Nascimento, J. C., Figueiredo, M. A. T., & Marques, J. S. (2013). Activity recognition using mixture of vector fields. *IEEE Transactions on Image Processing*, (pp. 1712–1725).

[Pires & Figueiredo, 2017] Pires, T. & Figueiredo, M. (2017). Shape-based trajectory clustering. In *International Conference on Pattern Recognition Applications and Methods*.

[Poiesi & Cavallaro, 2015] Poiesi, F. & Cavallaro, A. (2015). Tracking multiple high-density homogeneous targets. *IEEE Transactions on Circuits and Systems for Video Technology*, (pp. 623–637).

[Poppe, 2010] Poppe, R. (2010). A survey on vision-based human action recognition. *Image and vision computing*, (pp. 976–990).

[Porikli, 2004] Porikli, F. (2004). Learning object trajectory patterns by spectral clustering. In *IEEE Int. Conf. Multimedia Expo*.

[Turaga et al., 2008] Turaga, P., Chellappa, R., Subrahmanian, V., & Udrea, O. (2008). Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, (pp. 1473–1488).

IMAGE FUSION OF UNREGISTERED COLOUR DIGITAL PATHOLOGY IMAGES

Wael Saafin[*1], *Gerald Schaefer* [1], *Miguel Vega*[2], *Rafael Molina*[3], *Aggelos Katsagelos*[4]

[1] Dept. of Computer Science, Loughborough University, Loughborough, United Kingdom.
[2] Dept. of Languages and Information Systems, University of Granada, Granada, Spain.
[3] Dept. of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain.
[4] Dept. of Electrical Engineering and Computer Science, Northwestern University, Evanston, USA.
waelsaafin@hotmail.com, gerald.schaefer@ieee.org, mvega@ugr.es,
rms@decsai.ugr.es, aggk@eecs.northwestern.edu

ABSTRACT

Visual investigation of colour digital pathology (DP) images, which is better performed on improved captured images, has enormous interest. This work fuses multiple unregistered DP images to merge the information provided by each image. We propose a robust reconstruction approach that iteratively alternates between fused colour image recovery and registration parameter estimation. The image is found by optimising a function which combines fidelity to the data with a regulariser of the sought after image. We perform an accurate estimation of registration parameters which is crucial to obtain good recovered images for better visualisation and improved diagnosis. Using the proposed iterative approach, images are efficiently estimated at high convergence rates. Experiments were performed on simulated images and on real DP images as well, and the results have shown excellent quality of the estimated image.

Index Terms— Medical imaging, digital pathology, whole slide images, image fusion, image reconstruction,.

1. INTRODUCTION

Digital pathology (DP) imaging starts with tissue sample preparation, then scans the sample under a microscope using a virtual slide scanner, and finally the scanned view is sampled using a digital camera. The resulting digital whole slide image (WSI) represents high resolution digital images of complete glass histopathology or cytopathology slides, usually in standard red-green-blue (RGB) colour format. In the RGB representation, every pixel in the image plane is represented by three values corresponding to the red, green and blue components, which together constitute the overall colour of a pixel. A three channel RGB camera, typically a charge coupled device (CCD) camera, allows for colour detection and processing of the WSI. The resulting WSI usually is very large and a virtual slide viewer is required to investigate the tissue on display. WSI is useful for quality assurance, image analysis, and tracking how an image was viewed, and ultimately to aid in diagnosis [Weinstein et al., 2009, Glatz-Krieger et al., 2006].

DP studies the WSI instead of the sample fixed on a glass slide, and it is therefore recommended for histopathology and cytology samples. DP analysis techniques were proposed to detect diseases, determine its grading and type, cell structure, function, and chemistry [Hamilton et al., 2014, Irshad et al., 2014, Kothari et al., 2013, Veta et al., 2014, Saafin and Schaefer, 2017].

Any analysis of DP images should appropriately exploit the colour or spectral information residing in the obtained data. This is important to come up with stains and/or antibodies used to highlight microscopic components of the tissue under study, which selectively colour components in a tissue to enhance its visualisation.

∗ Corresponding Author.

Haematoxylin-eosin (H&E) stainings are routinely used to stain cell nuclei in blue/purple and cytoplasm and connective tissue in pink. Similarly; immunohistochemistry (IHC) highlights specific antigens in the tissue by injecting antibodies in the slide [Irshad et al., 2014].

In some cases a pathologist needs to visualise several stains at the same time and relate them to each other. This is a difficult task unless multiple slides can be merged in one image. This enables the pathologist to definitely locate different components in the image. Staining a sample with multiple stains can be done by either staining the slide with the required stains simultaneously, or by staining slides sequentially one stain at a time. The former approach necessitates unmixing techniques to separate stains that expectedly can result in new mixed colours when two or more stains add up at same location. Following the sequential approach each slide will have one stain, but will have multiple images that all are required to be included in analysis.

Medical image fusion combines two or more images to allow improved visualisation of abnormalities, ascertain differences in tissue, and to help in diagnosis. It has been applied to combine images of both the same and different modalities. A single modal fusion of medical images has been applied to ultrasound images to improve acquisition and visualisation of fetal heart [Gooding et al., 2010]. Although not applied to DP images, [Kobayashi et al., 2007] employed five quantum dots of similar sizes but different emission colours and captured observations in a sequential manner, one colour at a time. Then the observations were merged in one multicolour image to allow simultaneous visualisation and to predict the route of cancer metastasis into the lymph nodes. Medical image fusion has been applied also on different image modalities, where two images of different types are fused. The type might be magnetic resonance imaging (MRI), computed tomography (CT), position emission tomography (PET), and single photon emission computed tomography (SPECT) [Townsend and Beyer, 2002, Chajari et al., 2002, Sonn et al., 2014, Holupka et al., 1998]. Medical image fusion is a challenging problem due to the nature of the medical image modality [James and Dasarathy, 2014]. Image fusion can be classified into pixel-level and transform-based fusions. This work belongs to the pixel, or spatial level techniques [Galande and Patil, 2013].

To fuse two or more images the warping parameters should be either known or estimated for all observations. The warping parameters are those which can geometrically transform an image to fit with another, that is to realign images together. This work assumes the realistic case where the parameters are unknown and the images are unregistered then estimates them.

Image registration of DP images, which offers realignment of observations without fusing it, was addressed in [Déniz et al., 2015, Wang et al., 2014]. In [Déniz et al., 2015] sequentially stained images were used, where an observation is aligned to another. In [Wang et al., 2014] deformation problems like morphological distortions was included in a robust image registration method to align microscopic images. The output image in the two works does not offer visualisation or merging the information from the input observations. Our work shows simultaneously the visual information from all input observations into the output image making it possible to automatically or visually analyse the output image instead of the original observations. Moreover, the proposed method is applicable to two or more input images.

In this paper, we present an algorithm to fuse multiple colour DP images. This offers simultaneous visualisation of multiple images which can aid pathologists when investigating a WSI. To the best of our knowledge there is no published work addressing this exact approach of DP fusion. The proposed image reconstruction technique assumes unregistered input DP images, then aims to calculate one output image and hence estimates the fused image and registration parameters in a robust approach.

The rest of this paper is organised as follows: the problem is modelled and formulated in Section 2, while the proposed approach is presented in Section 3. Experimental results are presented in Section 4, then conclusions are stated in Section 5.

2. MODEL

Let us assume that we have access to a set of Q RGB colour images of the form

$$\mathbf{y}_{cq} = \mathbf{C}(\mathbf{s}_q)\mathbf{x}_c\,, \text{ for } q = 1, \ldots, Q, \tag{1}$$

 ISBN 978-0-9934207-2-6

where $c \in \{R, G, B\}$ denotes one of the three channels, $\mathbf{y}_{cq}$ represents the channel c of the q-th observation image, $\mathbf{C}(\mathbf{s}_q)$ is the $N \times N$ warping matrix of motion vector $\mathbf{s}_q = [\theta_q, c_q, d_q]^t$, where θ_q is the rotation angle, and c_q and d_q are, respectively, the horizontal and vertical translations of the q-th observation image with respect to the reference frame, $\mathbf{x}_c$ represents the channel c we want to estimate. $\mathbf{y}_{cq}$ and $\mathbf{x}_c$ are both $N \times 1$ vectors.

We recover the fused image channels $\mathbf{x}_c$ by solving

$$\min \sum_{c\in\{R,G,B\}} L(\mathbf{x}_c) \quad \text{s.t.} \quad \mathbf{y}_{cq} = \mathbf{C}(\mathbf{s}_q)\mathbf{x}_c, \quad \text{for } q = 1, \ldots, Q \text{ and } c \in \{R, G, B\}\,, \tag{2}$$

where

$$L(\mathbf{x}_c) = \mathbf{Q}(\mathbf{x}_c)\,, \tag{3}$$

with $\mathbf{Q}(\mathbf{x}_c)$ being the regularisation term proposed in [Saafin et al., 2015a, Saafin et al., 2015b] given by

$$\mathbf{Q}(\mathbf{x}_c) = \sum_{d\in\Delta}\sum_{i=1}^{N} \log_\epsilon(|\omega_d^{\mathbf{x}_c}(i)|)\,, \tag{4}$$

where

$$\log_\epsilon(|\omega_d^{\mathbf{x}_c}(i)|) = \begin{cases} \log(|\omega_d^{\mathbf{x}_c}(i)|), & |\omega_d^{\mathbf{x}_c}(i)| \geq \epsilon \\ \frac{|\omega_d^{\mathbf{x}_c}(i)|^2}{2\epsilon^2} - (\frac{1}{2} - \log(\epsilon)), & |\omega_d^{\mathbf{x}_c}(i)| \leq \epsilon \end{cases} \tag{5}$$

and $\omega_d^{\mathbf{x}_c}(i)$ is the i-th pixel of the filtered channel, that is,

$$\boldsymbol{\omega}_d^{\mathbf{x}_c} = \mathbf{F}_d\mathbf{x}_c\,, \tag{6}$$

where $\mathbf{F}_d$ is a high-pass filter operator, and the index $d \in \Delta$ denotes one of the filters in Δ. In this paper we use $\Delta = \{h, v, hv, vh\}$, where h, v represent respectively the first order horizontal and vertical difference filters, hv and vh the first order differences along diagonals.

3. RECONSTRUCTION APPROACH

Next we describe the optimisation approach to estimate the two main unknowns: the image $\mathbf{x}$ and the registration parameters $\mathbf{s}$. To convert the constrained optimisation problem in (2) into an unconstrained one utilising the alternating direction method of multipliers (ADMM), we define the following augmented Lagrangian functional

$$L(\mathbf{x}, \mathbf{s}, \boldsymbol{\lambda}) = \sum_{c\in\{R,G,B\}} L_c(\mathbf{x}_c, \mathbf{s}, \boldsymbol{\lambda}_c), \tag{7}$$

where

$$L_c(\mathbf{x}_c, \mathbf{s}, \boldsymbol{\lambda}_c) = \alpha L(\mathbf{x}_c) + \sum_{q=1}^{Q} \boldsymbol{\lambda}_{cq}^t(\mathbf{C}(\mathbf{s}_q)\mathbf{x}_c - \mathbf{y}_{cq}) + \frac{\beta}{2}\sum_{q=1}^{Q} \parallel \mathbf{C}(\mathbf{s}_q)\mathbf{x}_c - \mathbf{y}_{cq} \parallel^2, \tag{8}$$

and $L(\mathbf{x}_c)$ has been defined in (3), $\mathbf{s} = (\mathbf{s}_1, \ldots, \mathbf{s}_Q)$ is the set of motion vectors, $\boldsymbol{\lambda}_c = (\boldsymbol{\lambda}_{c1}, \ldots, \boldsymbol{\lambda}_{cQ})$ is the set of $N \times 1$ Lagrangian multiplier vectors $\boldsymbol{\lambda}_{cq}$, α and β are positive parameters, and $\parallel . \parallel$ denotes Euclidean norm. The ADMM leads to the following sequence of iterative unconstrained problems,

$$\mathbf{x}_c^{k+1} = \arg\min_{\mathbf{x}_c} L_c(\mathbf{x}_c, \mathbf{s}^k, \boldsymbol{\lambda}_c^k)\,, \tag{9}$$

$$\mathbf{s}^{k+1} = \arg\min_{\mathbf{s}} \sum_{c\in\{R,G,B\}} L_c(\mathbf{x}_c^{k+1}, \mathbf{s}, \boldsymbol{\lambda}_c^k)\,, \tag{10}$$

$$\boldsymbol{\lambda}_{cq}^{k+1} = \boldsymbol{\lambda}_{cq}^k - \beta[\mathbf{C}(\mathbf{s}_q^{k+1})\mathbf{x}_c^{k+1} - \mathbf{y}_{cq}]\,, \qquad q = 1, \ldots, Q, \tag{11}$$

 ISBN 978-0-9934207-2-6

where k is the iteration index. Notice that according to the ADMM formulation, $\mathbf{C}(\mathbf{s}_q)$ in (2) should not depend on the iteration index, as is not the case here. However, we have not encountered any convergence issues with this iterative procedure.

The first step is to optimise for the fused image $\mathbf{x}$ by combining (9) with (8), maximising-minimising the regularisation term $\mathbf{Q}(\mathbf{x}_c)$ in (4) for each colour component, we obtain the following update equation for $\mathbf{x}_c$ [Saafin et al., 2016],

$$\mathbf{x}_c^{k+1} = \left[\beta \sum_q \mathbf{C}^{k^t}(\mathbf{s}_q^k)\mathbf{C}^k(\mathbf{s}_q^k) + \alpha \sum_{d\in\Delta} \mathbf{F}_d^t \mathbf{\Omega}_{cd}^k \mathbf{F}_d\right]^{-1} \times \sum_q \mathbf{C}^k(\mathbf{s}_q^k)^t \left[\beta \mathbf{y}_{cq} - \lambda_{cq}^k\right], \tag{12}$$

where

$$\Omega_{cd}^k(i,i) = \min(1/|\omega_d^{\mathbf{x}_c^k}(i)|^2, 1/\epsilon^2). \tag{13}$$

The second step is to calculate the registration parameters using the estimated image $\mathbf{x}$. The approach we follow in this work is to simultaneously include all colour channels in the estimation procedure using the following equation

$$\mathbf{s}_q^{k+1} = \arg\min_{\mathbf{s}_q} \frac{\beta'}{2} \parallel \mathbf{C}(\mathbf{s}_q)\mathbf{z}_{\mathbf{x}^{k+1}} - \mathbf{z}_{\mathbf{y}_q} \parallel^2 . \tag{14}$$

where β' is a positive parameter, and $\mathbf{z}_\mathbf{x} = Y(\mathbf{x})$ where $Y(\mathbf{x})$ is calculated using

$$Y(\mathbf{x}) = 0.2989\,\mathbf{x}_R + 0.5870\,\mathbf{x}_G + 0.1140\,\mathbf{x}_B, \tag{15}$$

where $\mathbf{x}_R, \mathbf{x}_G$, and $\mathbf{x}_B$ are the R, G, and B channels of $\mathbf{x}$. $\mathbf{C}(\mathbf{s}_q)\mathbf{z}_\mathbf{x}$, can be approximated by expanding it into its first-order Taylor series around the previous value $\mathbf{s}_q^k$. We thus obtain (see [He et al., 2007])

$$\mathbf{C}(\mathbf{s}_q)\mathbf{z}_{\mathbf{x}^{k+1}} \approx \mathbf{C}(\mathbf{s}_q^k)\mathbf{z}_{\mathbf{x}^{k+1}} + \left[\mathbf{N}_1(\mathbf{s}_q^k)\mathbf{z}_{\mathbf{x}^{k+1}}, \mathbf{N}_2(\mathbf{s}_q^k)\mathbf{z}_{\mathbf{x}^{k+1}}, \mathbf{N}_3(\mathbf{s}_q^k)\mathbf{z}_{\mathbf{x}^{k+1}}\right] \times (\mathbf{s}_q - \mathbf{s}_q^k), \tag{16}$$

where $\mathbf{N}_i(\mathbf{s}_q^k)\mathbf{z}_{\mathbf{x}^{k+1}}$ is defined as

$$\left[\mathbf{N}_1(\mathbf{s}_q^k)\mathbf{z}_\mathbf{x}^{k+1}, \mathbf{N}_2(\mathbf{s}_q^k)\mathbf{z}_\mathbf{x}^{k+1}, \mathbf{N}_3(\mathbf{s}_q^k)\mathbf{z}_\mathbf{x}^{k+1}\right] = \left[(\mathbf{P}_1(\mathbf{s}_q^k)\mathbf{M}_1(\mathbf{s}_q^k) + \mathbf{P}_2(\mathbf{s}_q^k)\mathbf{M}_2(\mathbf{s}_q^k), \mathbf{M}_1(\mathbf{s}_q^k), \mathbf{M}_2(\mathbf{s}_q^k)\right] \tag{17}$$

and

$$\mathbf{M}_1(\mathbf{s}_q^k) = (\mathbf{I} - \mathbf{D}_{\mathbf{b}_q(\mathbf{s}_q)})(\mathbf{L}_{tr(\mathbf{s}_q)} - \mathbf{L}_{tl(\mathbf{s}_q)}) + \mathbf{D}_{\mathbf{b}_q(\mathbf{s}_q)}(\mathbf{L}_{br(\mathbf{s}_q)} - \mathbf{L}_{bl(\mathbf{s}_q)}) \tag{18}$$

$$\mathbf{M}_2(\mathbf{s}_q^k) = (\mathbf{I} - \mathbf{D}_{\mathbf{a}_q(\mathbf{s}_q)})(\mathbf{L}_{bl(\mathbf{s}_q)} - \mathbf{L}_{tl(\mathbf{s}_q)}) + \mathbf{D}_{\mathbf{a}_q(\mathbf{s}_q)}(\mathbf{L}_{br(\mathbf{s}_q)} - \mathbf{L}_{tr(\mathbf{s}_q)}) \tag{19}$$

$$\mathbf{P}_1(\mathbf{s}_q^k) = -[\mathbf{D}_\mathbf{u}\sin(\theta_q^k) + \mathbf{D}_\mathbf{v}\cos(\theta_q^k)] \tag{20}$$

$$\mathbf{P}_2(\mathbf{s}_q^k) = [\mathbf{D}_\mathbf{u}\cos(\theta_q^k) - \mathbf{D}_\mathbf{v}\sin(\theta_q^k)]. \tag{21}$$

where $\mathbf{I}$ is the identity matrix. $\mathbf{D}_{\mathbf{a}_q(\mathbf{s}_q)}$ and $\mathbf{D}_{\mathbf{b}_q(\mathbf{s}_q)}$ denote diagonal matrices with vectors $\mathbf{a}_q(\mathbf{s}_q)$ and $\mathbf{b}_q(\mathbf{s}_q)$ respectively in their diagonals which represent the horizontal and vertical displacements of observation q. $\mathbf{D}_\mathbf{u}$ and $\mathbf{D}_\mathbf{v}$ are diagonal matrices whose diagonals are vectors $\mathbf{u}$ and $\mathbf{v}$ respectively, representing pixel coordinates in $\mathbf{x}$. Matrices L_κ with $\kappa \in \{\mathbf{bl}(\mathbf{s}_q), \mathbf{br}(\mathbf{s}_q), \mathbf{tl}(\mathbf{s}_q), \mathbf{tr}(\mathbf{s}_q)\}$ are constructed in such a way that the product $L_\kappa\mathbf{z}$ produces pixels at the bottom-left, bottom-right, top-left, and top-right, locations of (u_q, v_q), respectively. Combining (16) with (14) and solving gives the update equation which we followed in our solution

$$\mathbf{s}_q^{k+1} = \mathbf{s}_q^k + \left[\lambda_q^k\right]^{-1} \Upsilon_q^k, \tag{22}$$

where λ_q^k and Υ_q^k correspond to the $q-$th observation at the $k-$th iteration with respectively $(i,j) \in \{1,2,3\}$ element and $i \in \{1,2,3\}$ element given by

$$\lambda_{qij}^k = \left[\mathbf{N}_i(\mathbf{s}_q^k)\mathbf{z}_{\mathbf{x}^{k+1}}\right]^t \mathbf{N}_j(\mathbf{s}_q^k)\mathbf{z}_{\mathbf{x}^{k+1}}, \tag{23}$$

$$\Upsilon_{qi}^k = \left[\mathbf{N}_i(\mathbf{s}_q^k)\mathbf{z}_{\mathbf{x}^{k+1}}\right]^t (\mathbf{z}_{\mathbf{y}_q} - \mathbf{N}_i(\mathbf{s}_q^k)\mathbf{z}_{\mathbf{x}^{k+1}}). \tag{24}$$

Utilising the estimated $\mathbf{s}_q^{k+1}$, the warping matrix can be calculated by [He et al., 2007, AlSaafin et al., 2016]

$$\begin{aligned}\mathbf{C}(\mathbf{s}_q)\mathbf{x} \approx \mathbf{D}_{b_q(\mathbf{s}_q)}(\mathbf{I} - \mathbf{D}_{a_q(\mathbf{s}_q)})\mathbf{L}_{\mathbf{bl}(\mathbf{s}_q)}\mathbf{x} + (\mathbf{I} - \mathbf{D}_{b_q(\mathbf{s}_q)})\mathbf{D}_{a_q(\mathbf{s}_q)}\mathbf{L}_{\mathbf{tr}(\mathbf{s}_q)}\mathbf{x} \\ + (\mathbf{I} - \mathbf{D}_{b_q(\mathbf{s}_q)})(\mathbf{I} - \mathbf{D}_{a_q(\mathbf{s}_q)})\mathbf{L}_{\mathbf{tl}(\mathbf{s}_q)}\mathbf{x} + \mathbf{D}_{b_q(\mathbf{s}_q)}\mathbf{D}_{a_q(\mathbf{s}_q)}\mathbf{L}_{\mathbf{br}(\mathbf{s}_q)}\mathbf{x}\,.\end{aligned} \tag{25}$$

The above approach for registering the observations assumes observations of the same sample, which is one application of the proposed algorithm. In cases where the observations are for the same tissue but with differences due to different staining the minimisation will look for the minimal distance between an observation and the estimated image, and hence the edges are not expected to co-localise but still optimal registration is obtained. The complete proposed algorithm is presented in Algorithm 1.

Algorithm 1 Proposed Image Fusion Algorithm

Require: Values α, β
Initialise $\mathbf{s}^0$, $\boldsymbol{\lambda}^0$, $\boldsymbol{\Omega}^0 = \{\boldsymbol{\Omega}_d^0, d \in \Delta\}$
$k = 0$
while convergence criterion is not met **do**
1. for $c \in \{R, G, B\}$
i Calculate $\mathbf{x}_c^{k+1}$ by solving (12).
ii For $d \in \Delta$, calculate $\boldsymbol{\Omega}_{cd}^{k+1}$ using (13).
iv For $q = 1, \ldots, Q$, update $\boldsymbol{\lambda}_{cq}^{k+1}$ using (11).
2. For $q = 1, \ldots, Q$, calculate $\mathbf{s}_q^{k+1}$ using (22).
3. Set $k = k + 1$
end while
return $\mathbf{x} = \left[\mathbf{x}_R^k, \mathbf{x}_G^k, \mathbf{x}_B^k\right]$.

4. EXPERIMENTS

We used DP images to perform two main experiments on normalised images. In the first experiment we used simulated DP observation images to mathematically assess the quality of the output image. The second experiment used real DP images. For both experiments, the stopping criteria are either a maximum number of iterations (25) or

$$\frac{\| \mathbf{x}^{k+1} - \mathbf{x}^k \|}{\| \mathbf{x}^k \|} \leq 10^{-3}\,. \tag{26}$$

In the first experiment, to simulate the observation images, we started with the image shown in Figure 1(a) referred later as the original image. It is for a tissue stained with haematoxylin and eosin and it is available on internet. Four images were generated from the original; each by performing a random warping of three parameters including horizontal, vertical and rotational displacements. Figures 1(b,c) show two of the four synthesised images which were fed to the proposed algorithm to find an estimate of the original image. The fused image is shown in Figure 1(d). To compare the estimated image with the original we use the peak signal to noise ratio (PSNR) measure for colour RGB images is calculated as

$$PSNR = 10\log(\frac{3N \cdot \max^2(\mathbf{x})}{\| \hat{\mathbf{x}} - \mathbf{x} \|^2})\,, \tag{27}$$

where $\max(\mathbf{x})$ represents the maximum possible value in $\mathbf{x}$, and N is the number of pixels $\mathbf{x}$. In the performed experiment the obtained PSNR was 45.09 dB. The number of iterations required to obtain the shown result was 8. The same experiment was performed for the original image shown in Figure 1(e) and the fused image is shown in Figure 1(h) giving PSNR=41.69 dB after performing 25 iterations. In both experiments the number of observations was 4 (although only two of them were shown in Figure 1).

 ISBN 978-0-9934207-2-6

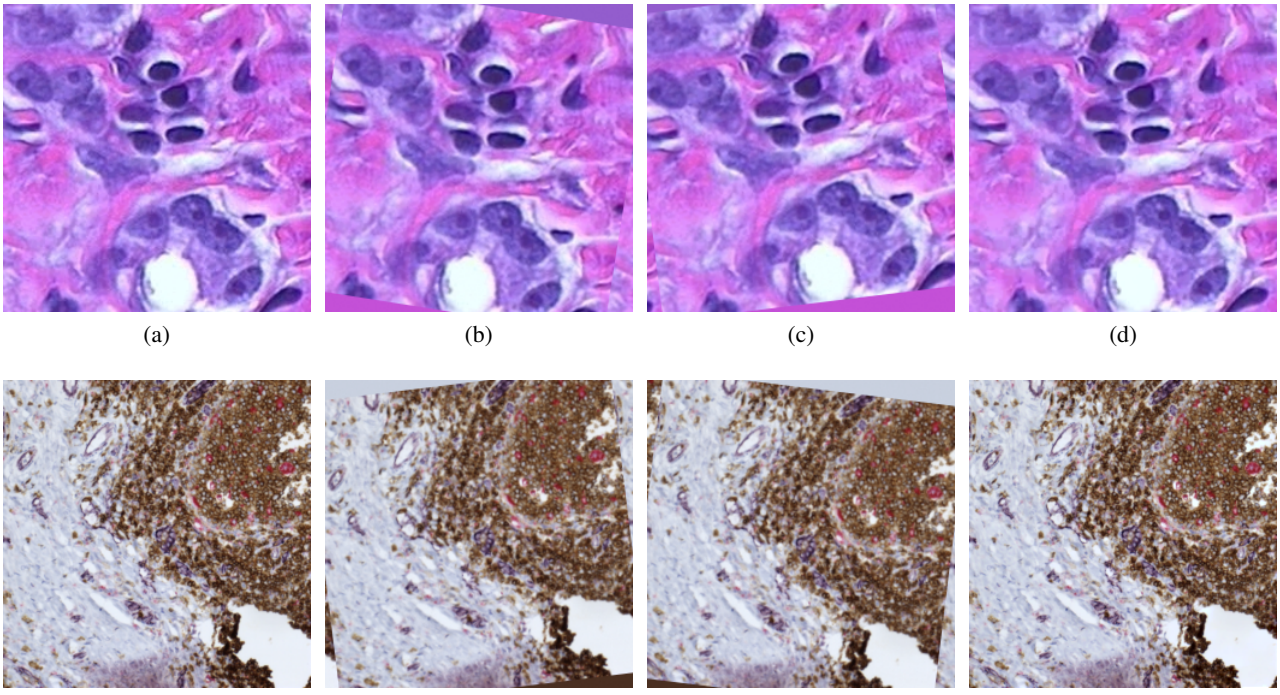


Fig. 1: Image fusion of simulated unregistered DP images. (a) original image, (b,c) 2 of 4 observations, (d) estimated image with PSNR=45.09 dB (e) original image, (f,g) 2 of 4 observations, (h) estimated image with PSNR=41.69 dB.

In the second experiment we used five images acquired during a sequential staining experiment for the breast tissue samples shown in Figure 2 of human epidermal growth factor receptor 2 (a), haematoxylin-eosin (b), estrogen receptor (c), Ki67 (d), and progesterone receptor (e) sections [Déniz et al., 2015]. The real observations were fused into the image shown in Figure 2(f). The visualisation and contrast of the estimated image can be further enhanced using a histogram stretching operation as shown in Figure 2(g).

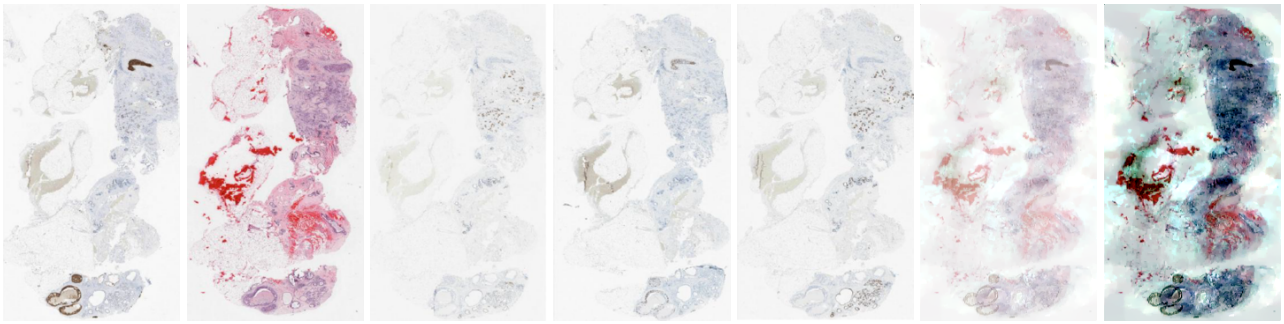
The high PSNR and the visual assessment of the output images in both experiments prove the excellent performance of our proposed algorithm.

5. CONCLUSIONS

Colour digital pathology images can be fused to combine information from different images of a sample. This can help in better visualising abnormalities, diagnosing the stage of a disease and observing the development of illness. Besides, the fused image can be analysed instead of analysing multiple images. This is important for automatic analysis where it is difficult to analyse multiple images at the same time. Accurate estimation of DP image registration parameters which is challenging but vital for diagnosis tasks has been achieved at high convergence rates. Similar application can benefit from this proposal to enhance diagnostics, analysis, and historical documentation. Future work will aim at extending this proposed work to remove the blur which is expected in DP images due to unfocussed regions that appear as a result of the three dimensional structure of a scanned slide.

Acknowledgement

This work was supported by the EC under Marie Curie grant actions, grant No. 612471, Academia and Industry Collaboration for Digital Pathology (AIDPATH) project (`http://aidpath.eu/`).



(a) (b) (c) (d) (e) (f) (g)

Fig. 2: Image fusion of unregistered sequentially stained real images of breast tissue WSI. (a) human epidermal growth factor receptor 2, (b) haematoxylin-eosin, (c) estrogen receptor, (d) Ki67 protein, (e) progesterone receptor, (f) estimated image, (g) estimated image after histogram stretching.

6. REFERENCES

[AlSaafin et al., 2016] AlSaafin, W., Villena, S., Vega, M., Molina, R., and Katsaggelos, A. K. (2016). Compressive sensing super resolution from multiple observations with application to passive millimeter wave images. *Digital Signal Processing*, 50:180–190.

[Chajari et al., 2002] Chajari, M., Lacroix, J., Peny, A., Chesnay, E., Batalla, A., Henry-Amar, M., Delcambre, C., Génot, J., Fruchard, C., and Bardet, S. (2002). Gallium-67 scintigraphy in lymphoma: is there a benefit of image fusion with computed tomography? *European Journal of Nuclear Medicine and Molecular Imaging*, 29(3):380.

[Déniz et al., 2015] Déniz, O., Toomey, D., Conway, C., and Bueno, G. (2015). Multi-stained whole slide image alignment in digital pathology. In *SPIE Medical Imaging*, pages 94200Z–94200Z.

[Galande and Patil, 2013] Galande, A. and Patil, R. (2013). The art of medical image fusion: A survey. In *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 400–405.

[Glatz-Krieger et al., 2006] Glatz-Krieger, K., Spornitz, U., Spatz, A., Mihatsch, M. J., and Glatz, D. (2006). Factors to keep in mind when introducing virtual microscopy. *Virchows Archiv*, 448(3):248–255.

[Gooding et al., 2010] Gooding, M., Rajpoot, K., Mitchell, S., Chamberlain, P., Kennedy, S., and Noble, J. (2010). Investigation into the fusion of multiple 4-d fetal echocardiography images to improve image quality. *Ultrasound in Medicine & Biology*, 36(6):957–966.

[Hamilton et al., 2014] Hamilton, P. W., Bankhead, P., Wang, Y., Hutchinson, R., Kieran, D., McArt, D. G., James, J., and Salto-Tellez, M. (2014). Digital pathology and image analysis in tissue biomarker research. *Methods*, 70(1):59–73.

[He et al., 2007] He, Y., Yap, K., Chen, L., and Chau, L.-P. (2007). A nonlinear least square technique for simultaneous image registration and super-resolution. *IEEE Transactions on Image Processing*, 16(11):2830–2841.

[Holupka et al., 1998] Holupka, E., Kaplan, I., and Burdette, E. (1998). Ultrasound localization and image fusion for the treatment of prostate cancer. US Patent 5,810,007.

[Irshad et al., 2014] Irshad, H., Veillard, A., Roux, L., and Racoceanu, D. (2014). Methods for nuclei detection, segmentation, and classification in digital histopathology: A review. *IEEE Reviews in Biomedical Engineering*, 7:97–114.

 ISBN 978-0-9934207-2-6

[James and Dasarathy, 2014] James, A. and Dasarathy, B. (2014). Medical image fusion: A survey of the state of the art. *Information Fusion*, 19:4–19.

[Kobayashi et al., 2007] Kobayashi, H., Hama, Y., Koyama, Y., Barrett, T., Regino, C., Urano, Y., and Choyke, P. (2007). Simultaneous multicolor imaging of five different lymphatic basins using quantum dots. *Nano Letters*, 7(6):1711–1716.

[Kothari et al., 2013] Kothari, S., Phan, J. H., Stokes, T. H., and Wang, M. D. (2013). Pathology imaging informatics for quantitative analysis of whole-slide images. *Journal of the American Medical Informatics Association*, 20(6):1099–1108.

[Saafin and Schaefer, 2017] Saafin, W. and Schaefer, G. (2017). Pre-processing techniques for colour digital pathology image analysis. In *Annual Conference on Medical Image Understanding and Analysis*, pages 551–560.

[Saafin et al., 2015a] Saafin, W., Vega, M., Molina, R., and Katsaggelos, A. K. (2015a). Image super-resolution from compressed sensing observations. In *IEEE International Conference on Image Processing (ICIP)*, pages 4268–4272.

[Saafin et al., 2016] Saafin, W., Vega, M., Molina, R., and Katsaggelos, A. K. (2016). Compressed sensing super resolution of color images. In *24th European Signal Processing Conference (EUSIPCO)*, pages 1563–1567.

[Saafin et al., 2015b] Saafin, W., Villena, S., Vega, M., Molina, R., and Katsaggelos, A. K. (2015b). Pmmw image super resolution from compressed sensing observations. In *23rd European Signal Processing Conference (EUSIPCO)*, pages 1815–1819.

[Sonn et al., 2014] Sonn, G., Chang, E., Natarajan, S., Margolis, D., Macairan, M., Lieu, P., Huang, J., Dorey, F., Reiter, R., and Marks, L. (2014). Value of targeted prostate biopsy using magnetic resonance–ultrasound fusion in men with prior negative biopsy and elevated prostate-specific antigen. *European Urology*, 65(4):809–815.

[Townsend and Beyer, 2002] Townsend, D. W. and Beyer, T. (2002). A combined pet/ct scanner: the path to true image fusion. *The British Journal of Radiology*, 75(suppl_9):S24–S30.

[Veta et al., 2014] Veta, M., Pluim, J. P., van Diest, P. J., and Viergever, M. A. (2014). Breast cancer histopathology image analysis: A review. *IEEE Transactions on Biomedical Engineering*, 61(5):1400–1411.

[Wang et al., 2014] Wang, C., Ka, S., and Chen, A. (2014). Robust image registration of biological microscopic images. *Scientific Reports*, 4.

[Weinstein et al., 2009] Weinstein, R. S., Graham, A. R., Richter, L. C., Barker, G. P., Krupinski, E. A., Lopez, A. M., Erps, K. A., Bhattacharyya, A. K., Yagi, Y., and Gilbertson, J. R. (2009). Overview of telepathology, virtual microscopy, and whole slide imaging: prospects for the future. *Human Pathology*, 40(8):1057–1069.

High Speed Reconstruction of a Scene Implemented Through Projective Texture Mapping

William Clifford[1*], Catherine Deegan[2], and Charles Markham[1]
[1]Department of Computer Science, Maynooth University, Maynooth, Co. Kildare, Ireland.
[2]Department of Engineering, Institute of Technology Blanchardstown, Dublin 15, Ireland.

Abstract

This paper describes a technique for reconstructing a scene from different points of view using a single frame of video. The technique operates by projecting a frame of video onto a simplified 3D model of the scene. The paper demonstrates that it is possible to reconstruct views, with a convincing standard of photorealism, at high speed, given an environment with little variation in its projective planes, such as tunnels or corridors. Other 3D reconstruction techniques are placed under consideration and discussed with respect to the proposed method. The projective texture mapping technique is implemented on a corridor environment and the resulting simulated images are compared with their real counterparts qualitatively. Quantitative measurements of frame rate and resolution reduction introduced by the technique are also presented. Finally, an image of the Dublin port tunnel is tested with this technique to demonstrate the potential of adding a steering effect to video based driving simulators.

Keywords: Driving Simulators, Projective Texture Mapping, Dashboard camera, Video.

1 Introduction

Driving simulators provide a safe and controlled environment in which to conduct experiments on driver behaviour. Most driving simulators use a model based approach to creating the space through which to drive [Engström et al., 2005, Strayer et al., 2003]. However, a limitation of this approach is that the driving experience can lack the fidelity and complexity of driving on a real road. One of the main issues identified with creating a driving simulator is that it is challenging to replicate aspects of the real world environment, for example the illumination [Lee, 2011]. Given a specification in the exact metrics (road measurements) of an environment, it is possible to construct 3D models that, although consistent with real-world measurements, lack convincing photorealism. The model based approach, although effective for driver training, is not so useful when studying behaviour in response to subtle changes in road layout (e.g. the effect of new infrastructure, signs and temporary installations). It is, therefore, a complicated task to generate new models to examine driving behaviour relating to scenarios found on specific roads.

Advanced methods have been developed to reconstruct a 3D scene from many images, these include Kintinuous and other techniques built on the point cloud library [Whelan et al., 2012, Newcombe et al., 2011]. However, these methods typically require more advanced acquisition systems and high performance hardware to render a realistic reconstruction at driving speeds.

The overall aim of this work is to demonstrate the feasibility of using a single video recorded from a dashboard camera (‘dash cam’) to provide the scene input to a simulator. This paper focusses on a new method based on Projective Texture Mapping that will be used to provide steering limited within the dash cam scene.

* William.Clifford@mu.ie

 ISBN 978-0-9934207-2-6

Given the pros and cons of the driving simulator approaches, which involve full reconstruction, it is clear there are two important factors when designing a driving simulator; (1) that it looks photorealistic and (2) that the environment is kept simple enough to limit computationally intense methods for the computer to interpret. For these reasons, it may be ideal to incorporate a partial reconstruction of the 3D environment. This puts more emphasis on texturing and makes the simulator experience more realistic.

In general, texture mapping involves matching the vertices of an object to texels (pixels on the texture image), this process is quite complicated for intricate objects. A proposed form of texture mapping, allows for a 3D reconstruction of an environment if coupled with a known characteristic of its projective planes [Everitt, 2001, Segal et al., 1992]. It is the central proposition of this paper that if a projector is set in a 3D environment that displays a texture onto a set of surfaces that approximate the geometry of the real world, then it is possible to view the perspective changes one would expect to see in the real world.

2 Summary of Model Generation Techniques

There are a number of methods that can build textured 3D models using stereo or RGBZ cameras. Typically, these cameras require a structured light source or time of flight camera to work. This limits the range and lighting conditions in which image acquisition is optimal. Dense stereo provides another method to obtain depth information required for full 3D reconstruction, this approach requires calibrated camera pairs and more advanced acquisition systems [Whelan et al., 2012]. Other methods such as Visual Structure from Motion (Visual SFM) work using standard cameras but are constrained by a limited reconstruction volume [Wu, 2011].

Earlier attempts have been made to create a photorealistic simulator based on video playback [Brogan et al., 2013]. This simulator allowed the user to accelerate and brake by speeding up and slowing down the frames per second based on the input from the accelerator and braking controls. The video and depth map approach used in this simulator produced a limited steering effect, however it was computationally intensive and created distortions in the run time display. These issues have been overcome by the approach presented here.

3 Theory

Projective texture mapping allows a 2D texture to be projected onto an arbitrary surface as part of the rendering process used by OpenGL. The basic technique of texture mapping is integrated into OpenGL, however the following mathematical discussion is specific to the simulation being developed [Wright & Sweet, 1999]. A simple model consisting of an open-ended box was used to model a corridor (and ultimately a tunnel), see Figure 1. The projector is placed in the reconstructed scene in the position and orientation equivalent to the position where the real camera was used to record the view (Figure 1).

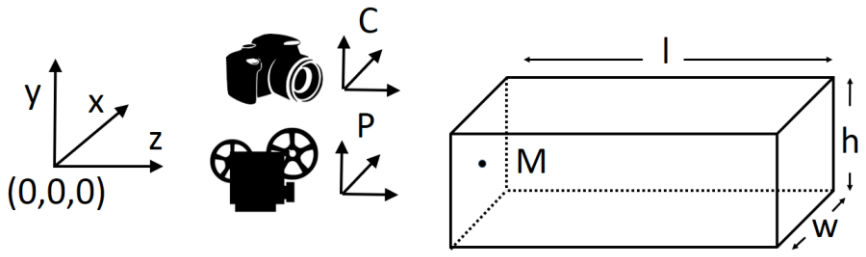


Figure 1 Co-ordinate system to bring camera (C), projector (P) and model (M) together

Vertex co-ordinates describing the object are multiplied by the inverse of the OpenGL MODEL_VIEW matrix, C. This moves the vertices in world co-ordinates back into the camera (eye) co-ordinate frame.

 ISBN 978-0-9934207-2-6

$$\begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} = C^{-1} \cdot \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \tag{3.1}$$

The vertex points are then converted to projector-space which allows measurement with respect to the projectors frame of reference. Taking the dot product of the vertex co-ordinates (*x, y, z, w*) with the planar coefficients for each of the three projector planes (*R, S, T*) generates (*s, t, r, q*) texture co-ordinates in eye space. The Q provides a homogeneous means of analysis.

$$s = S_1x' + S_2y' + S_3z' + S_4w' \tag{3.2}$$
$$t = T_1x' + T_2y' + T_3z' + T_4w' \tag{3.3}$$
$$r = R_1x' + R_2y' + R_3z' + R_4w' \tag{3.4}$$
$$q = Q_1x' + Q_2y' + Q_3z' + Q_4w' \tag{3.5}$$

The projector was orientated and located by defining planes (*S, T* and *R*) along the world co-ordinate *X, Y* and *Z* axes that were coincident with the world co-ordinate system, centred at (0,0,0) and projecting along the Z-axis. In this case (*s, t, r, q*) equalled (*x, y, z, w*), see Figure 2.

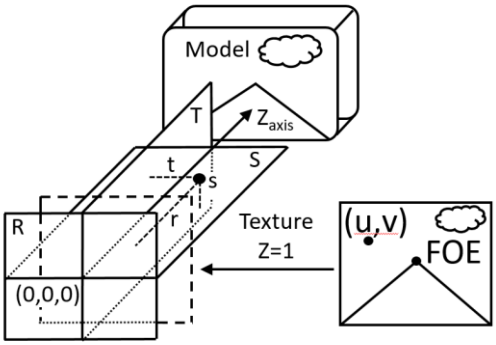


Figure 2 OpenGL projective texture mapping, automated generation of texture co-ordinates.

Vertices further from the S and T planes correspond to texture coordinates further from the focus of expansion in the texture. Increased distance from the R plane moves points towards the FOE, (small far away). The texture matrix, M, is configured to put the focus of expansion, FOE, at the origin of the texture. The texture matrix is also used to scale, rotate, and apply a perspective transform to the texture, see equation 3.6.

$$M = T \cdot S \cdot R \cdot P \tag{3.6}$$

where T is the translation matrix, S is the scale matrix, R is the rotation matrix and P is perspective projection matrix. Multiplication of the texture co-ordinates by the texture matrix and then normalizing provides the 2D (u,v) required to access the texture.

$$\begin{bmatrix} s' \\ t' \\ r' \\ q' \end{bmatrix} = M \cdot \begin{bmatrix} s \\ t \\ r \\ q \end{bmatrix} \tag{3.7}$$

With the correct values of scale, aspect ratio (same as input texture), rotation and field of view (FOV), the projector can be made to line up the projected photograph with the structure of the model, see Figure 3. In practice this was done using slider bar controls on a Windows application until alignment was achieved.

 ISBN 978-0-9934207-2-6

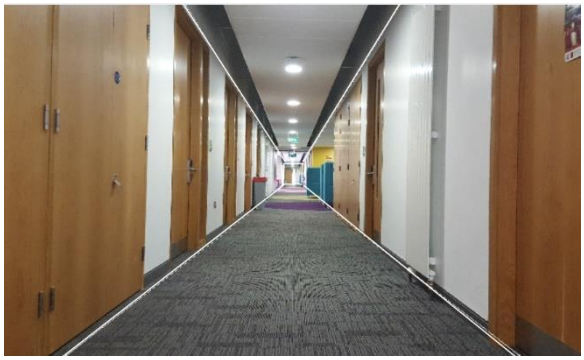


Figure 3 Photograph projected onto the open-ended box with model edges shown as white lines superimposed. These edges represent the boundaries between walls, ceiling and floor used in the model and were used to align the photograph with the model.

The position and orientation of the vehicle can be described using two parameters to set the displacement, *d*, which is the position on the road and a parameter to set the direction, θ. These parameters are used to build the MODEL_VIEW matrix for the camera, see Figure 4.

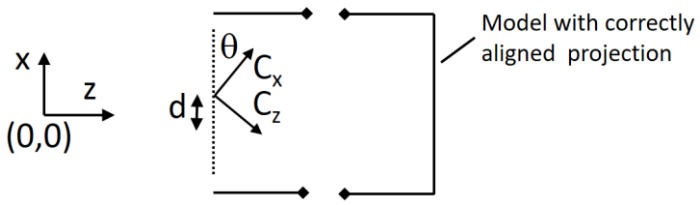


Figure 4 Moving the camera relative to the model to simulate vehicle motion (plan view)

4 Experiment

Experiment 1 Comparison of synthesised images with real photographs of the same scene.

6 images were collected of a straight corridor (75 meters in length, 1.8 meters wide, and 2.5 meters high). The images collected included, a central straight view image (placed 90 centimetres from the right and left wall, and approximately 1 metre from the ground leaving the camera front and parallel to the central axis of the corridor) from the end of the hallway, central positioned with left and right yaw rotation, and left and right translation of the camera (placed 45 centimetres from the right and then the left wall). The central straight view image was used as the input to the projective texture mapping program and using this software all the corresponding views were simulated.

The simulated images and real images were cropped so as to limit the field of view to that presented to the driver in the simulator. The original real images contain additional information at the edges that the forward looking image does not capture and so cannot be recreated. The main effect is to reduce the field of view from that available on the camera used to capture the original texture image.

Experiment 2 Creation of synthesised views from an image of the Dublin port tunnel

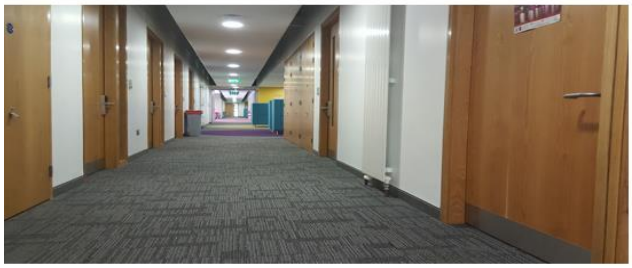
An image of the Dublin port tunnel was tested under similar conditions to account for the feasibility of moving the point of view around this 3D textured driving environment.

 ISBN 978-0-9934207-2-6

5 Results

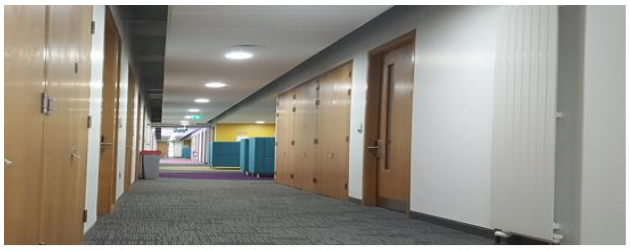
Real Straight View

Simulated Straight View

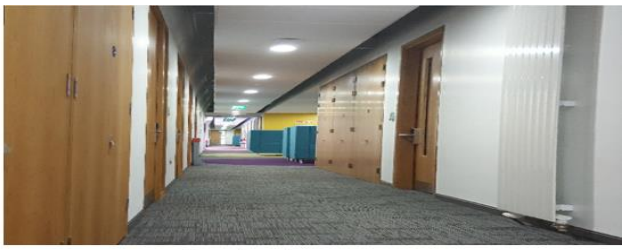


Real Central Position Looking Right

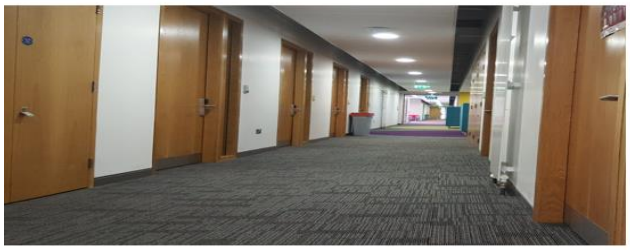
Simulated Central Position Looking Right



Real Left Position Central

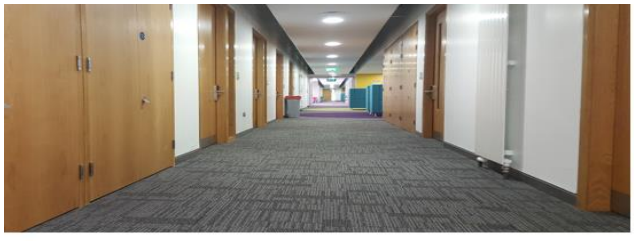


Simulated Left Position Central

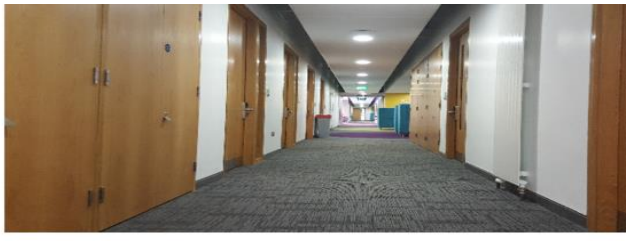


Real Right Position Central

Simulated Right Position Central



Real Central Position Looking Left



Simulated Central Position Looking Left

Figure 5 Images left-aligned are the real images of a hallway. Images right-aligned are the simulated images which used the "real straight view" image as an input to implement projective texture mapping.

 ISBN 978-0-9934207-2-6

Figure 5 presents the real world (left-aligned) and simulated (right-aligned) versions of the hallway environment that was recreated. Qualitatively, the simulated images are quite close to the real-world representations. However, there are some minor effects that become apparent to the attentive viewer such as sampling and aliasing effects in the rendering of fine lines of the flooring. There is also stretching of perspectives of objects that are contained within the hallway rather than on the walls, floor or ceiling. A clear example of this can be found in the simulated right position central view versus the real right position central view (Figure 5).

A quantitative measure of how this simulator performs in terms of reconstruction time can be found in the frames per second which operates at 31.25 frames per second on a laptop with an Intel core i5 processor and a GeForce GTX 950M graphics card. The original texture images of the hallway had a resolution of 5312x2988 pixels, and the simulated image acquired 2916x2360 of those pixels. The reduction in resolution is 57%-caused by the reduced field required to accommodate lane movement and steering.

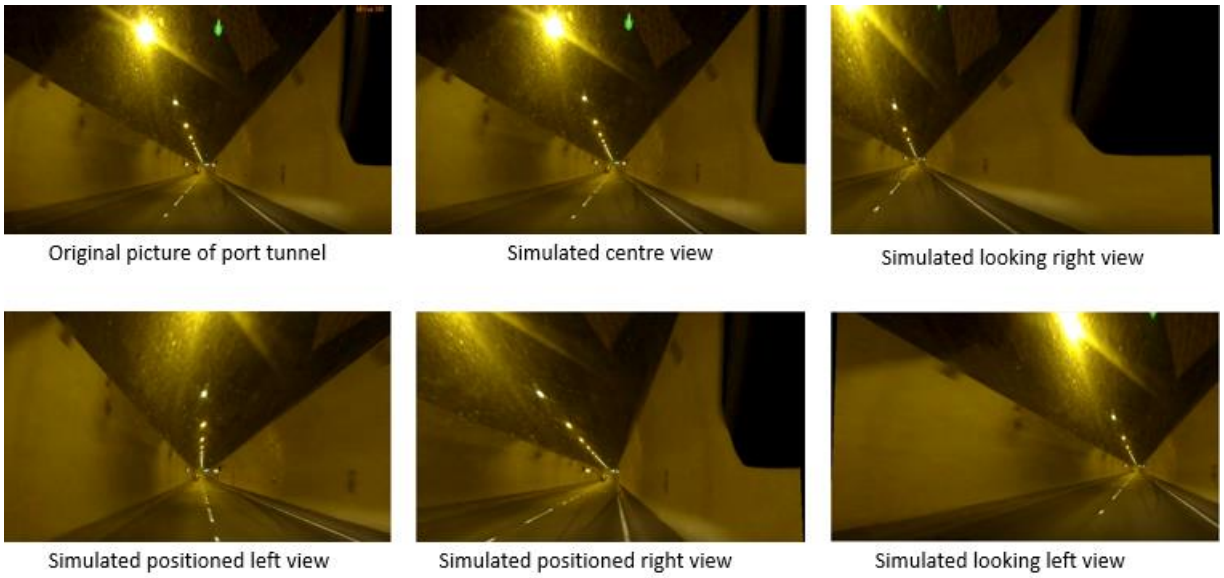


Figure 6 First image of the Dublin port tunnel as labelled and all other images are the simulated orientation made from that initial image.

Figure 6 presents the application of projective texture mapping tool on an image of the Dublin port tunnel. Although there are no real-world images to compare different positions and orientations, it shows that it is possible to make a set of simulated images based on what they would be expected to look like from a driver's perspective.

6 Conclusion/Discussion

This work has demonstrated that it is possible to reconstruct an image from a different point of view from a single frame of video. These techniques are likely to be effective where the path chosen by the viewer is similar to that recorded to produce the video. The simulator being developed is planned to be used for driver distraction studies where complicated navigation is not required. The main distractions will be real or augmented features on the side of the road. The results are sufficient for a person driving/walking through an environment such as this.

The technique developed requires a camera with a wider field of view than the view being reconstructed. An extension to this technique will be to use machine vision methods to detect cars passing on the right and add a simplified model to the scene to improve the projective view [Kamat & Ganesan, 1995].

Figure 6 provides the simulated and original image of the Dublin port tunnel thus illustrating how projective texture mapping can be applied to real world driving simulators. Given the Figure 5 success of the simulation of the hallway, it is a fair assumption that Figure 6 illustrates a well simulated attempt of estimating the Dublin port tunnel driver perspectives while driving and varying position and orientation through steering.

7 References

[Brogan et al., 2013] Brogan, M., Daly, N., Kaneswaren, D. A., Commins, S., Markham, C., Deegan, C. (2013) Layering Reality: Realistic Driving Simulation. IT&T Conference, AIT, Athlone.

[Engström et al., 2005] Engström, J., Johansson, E., & Östlund, J. (2005). Effects of visual and cognitive load in real and simulated motorway driving. *Transportation Research Part F: Traffic Psychology and Behaviour*, *8*(2), 97-120.

[Everitt, 2001] Everitt, C. (2001). Projective texture mapping. *White paper, NVidia Corporation*, *4*.

[Kamat & Ganesan, 1995] Kamat, V., & Ganesan, S. (1995, May). An efficient implementation of the Hough transform for detecting vehicle license plates using DSP'S. In *Real-Time Technology and Applications Symposium, 1995. Proceedings* (pp. 58-59). IEEE.

[Lee, 2011] Lee, J. D. (Ed.). (2011). *Handbook of driving simulation for engineering, medicine, and psychology*: CRC Press.

[Newcombe et al., 2011] Newcombe, R. A., Izadi S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., Kohi, P., Shotton, J., Hodges, S., Fitzgibbon, A. (2011) *KinectFusion: Real-time dense surface mapping and tracking*. Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on, IEEE.

[Segal et al., 1992] Segal, M., Korobkin, C., Van Widenfelt, R., Foran, J., & Haeberli, P. (1992, July). *Fast shadows and lighting effects using texture mapping.* In ACM SIGGRAPH Computer Graphics (Vol. 26, No. 2, pp. 249-252). ACM.

[Strayer et al., 2003] Strayer, D. L., Drews, F. A., & Johnston, W. A. (2003). Cell phone-induced failures of visual attention during simulated driving. *Journal of experimental psychology*: Applied, 9(1), 23.

[Whelan et al., 2012] Whelan, T., Kaess, M., Fallon, M., Johannsson, H., Leonard, J., & McDonald, J. (2012). Kintinuous: Spatially extended kinectfusion.

[Wright & Sweet, 1999] Wright, R. S., & Sweet, M. (1999). *OpenGL SuperBible* with Cdrom. Sams.

[Wu, 2011] Wu, C. (2011). VisualSFM: A visual structure from motion system.

Improving The Viola-Jones Face Detection Performance by Using The Brightness Channel in HSV and HLS Colour Spaces

Inas Al-Taie, Adrian Clark and Nassr Azeez

Computer Science and Electronic Engineering
University of Essex, UK.

Abstract

The algorithm due to Viola and Jones is the *de facto* standard way for locating faces in images. Although in widespread use, it does have shortcomings: it is effective on only frontal images, performance decreases with changes to the head size in an image, and it is sensitive to illumination. Generally, the algorithm is applied to grey-scale images in which the value of each pixel represents the intensity of light at this pixel. Illumination plays a surprisingly significant role in the effectiveness of Viola-Jones performance. This paper investigates whether using the V and L channels in HSV and HLS colour spaces respectively has an effect on the performance of Viola-Jones detection. The presence of statistically significant performance differences is assessed using McNemar's Test . It is found that using the V-channel of HSV colour space yields significantly fewer incorrect classifications than when the algorithm is applied to grey-scale images or to the L channel.

Keywords: Adaboost, HSV, Colour space, Face Detection.

1 Introduction

Face recognition has become one of the most important applications of computer vision, widely used in security systems [Zhao et al., 2003, Chao, 2007]. Face recognition is a biometric application that allows an individual to be identified or have their identity verified in a digital image. In general, a face recognition system involves three steps: face detection, feature extraction and face recognition [Schroff et al., 2015, Klare et al., 2015].

Accurate face detection (sometimes called face location) is important because knowing where faces are located makes the recognition phase less complicated [Zhao et al., 2003]. Viola and Jones proposed an object detection framework characterized by high detection accuracy and low computation time, and applied it to the problem of face detection. Their technique involves collecting a large set of face and non-face images and training a classifier to discriminate them [Viola and Jones, 2004].

Although the most successful face location algorithm in widespread use — it is used within almost every camera and smartphone — it is known that the Viola-Jones technique both misses faces (false negatives) and identifies non-face regions as being faces (false positives). It requires only grey-scale images, and it generally used with colour cameras by first converting images to intensity. However, there are colour spaces, such as HSV and HLS, which arguably separate the image intensity (luma) from the colour information (chroma) in a better way. This paper explores whether the failure rates can be reduced by the simple expedient of manipulating the colour space of an image before presenting it to Viola-Jones.

The remainder of this paper is structured as follows. Section 2 presents the Viola-Jones face recognition algorithm, describing its major stages in some detail. Section 3 outlines the most common failure modes of the technique. This is followed by a discussion of colour spaces in section 4. McNemar's test is discussed in section 5, then section 6 presents the results of experiments that demonstrate how the effectiveness of the algorithm can be improved by first manipulating the colour space of an image. Section 7 draws conclusions.

 ISBN 978-0-9934207-2-6

2 Face detection using the Viola-Jones approach

Before the advent of the Viola-Jones algorithm, the principal way of identifying faces was by colour, but that approach was easily confused by other regions of skin or by other similarly-coloured image features. Unlike previous approaches, Viola-Jones operates on grey-scale images. The Viola-Jones approach [Viola and Jones, 2004] involves the reduction of an input image to a fixed size of 24×24 pixels. After that, three kinds of Haar features are calculated, as shown in Figure 1.

Making use of the so-called *integral image* representation allow the computation of the sums of rectangular regions in constant time, so the computation of the Haar features is rapid. (In computer graphics, an integral image is known as a *summed area table* and is widely used in texture mapping.) The value of a point (x, y) within an integral image is computed as the sum of all pixels in the above-left region of the original image, including the value at the point (x, y) itself. This allows to calculation of the sum of a rectangular region at any position or size in the image to be achieved using only four values of the integral image and hence can be computer in constant time. The pixel value at the point (x, y) in the integral image is simply computed by:

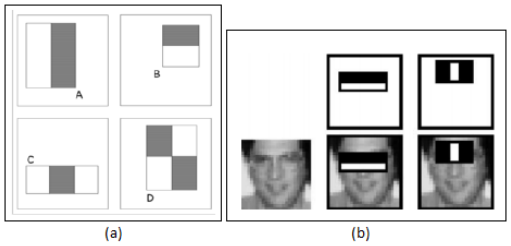


Figure 1: (a) Feature types used by Viola and Jones (b) Haar features that looks similar to the eye region and the bridge upper nose region is applied on a face [Viola and Jones, 2004].

$$s(x, y) = i(x, y) + s(x-1, y) + s(x, y-1) - s(x-1, y-1) \quad (1)$$

The pixel value $i(x, y)$ in the original image is added to the values directly above and left to this pixel at $s(x-1, y)$ and $s(x, y-1)$ from the integral image. Then, the value top-left of $i(x, y)$ in the integral image, $s(x-1, y-1)$, is subtracted. Figure 2 illustrates the process.

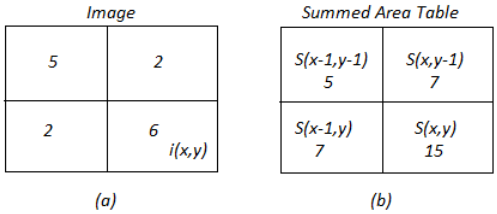


Figure 2: (a) Original image (b) Integral image

Most Haar features are irrelevant but a few will help to detect the face. The learning process tries to find which set of features reduces the error rate, the number of mis-classifications, in the identification of face and non-face image regions. Individually, each feature is considered as a weak classifier; all the weak classifiers are combined in such away as to gain a strong classifier, as discussed below [Jones and Viola, 2006].

2.1 Adaboost Training and Feature Selection

The technique known as *adaptive boosting* or *AdaBoost* is can be used to combine multiple weak classifiers into a single strong classifier, thereby improving performance. With the 24×24-pixel windows used in Viola-Jones, there are 162,336 possible rectangle features, and computing them all is time-consuming. Adaboost was used to establish those that contribute to distinguishing face and non-face regions.

A training set of labelled images is prepared by scaling all images to 24×24 pixels. So, each image has index $l, l = 1 \ldots L$. A corresponding value y_l is assigned for each image [Viola and Jones, 2004]: $y_l = 1$ for faces and $y_l = 0$ for non-faces. Then, some weights are initialized:

$$w_{1,l} = \frac{1}{2p_-}, \frac{1}{2p_+} \quad (2)$$

where p_- and P_+ are the number of negative (images without faces) and positive (images with faces) images in the images set.

 ISBN 978-0-9934207-2-6

For $i = 1 \dots L$, the following steps are performed.

1. Normalize the weights using
$$\frac{w_{i,l}}{\sum_{j=1}^{n} w_{i,j}} \rightarrow w_{i,l} \tag{3}$$

2. For each feature j, train a classifier h_j which is restricted to use a single feature. The classifier's error rate ε_j is evaluated with respect to $w_{i,j}$ as
$$\varepsilon_j = \sum_{l=0}^{L-1} w_{i,l} |h_j(x_l) - y_l| \tag{4}$$

3. Choose the classifier, h_i with the lowest error ε_i. Update the weights using
$$\begin{aligned} w_{i+1,l} &= w_{i,l} \beta_i^{1-\varepsilon_i} \\ \beta_i &= \frac{\varepsilon_i}{1-\varepsilon} \end{aligned} \tag{5}$$

The final strong classifier is then

$$h(x) = \begin{cases} 1, & \text{if } \sum_{i=0}^{I-1} \alpha_i h_i(x_l) \geq \frac{1}{2} \sum_{i=0}^{I-1} \alpha_i \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $\alpha_i = \log \frac{1}{\beta_i}$.

2.2 Cascaded Classifiers

The strong classifiers generated by AdaBoost can be implemented quickly but not fast enough to execute in real time. To increase overall computation speed for an entire image, Viola and Jones [Viola and Jones, 2001] devised a *cascade* of classifiers: the first of these execute quickly and reject obvious non-faces; subsequent stages make use of more Haar features, and are consequently slower to compute.

3 Failure Modes of the Viola-Jones Algorithm

As alluded to in section 1, the Viola-Jones algorithm does not find all faces in all images and can identify non-face regions as being faces. The following paragraphs identify the most commonly-reported failure modes.

Head pose. The Viola-Jones algorithm works only for subject looking at the camera, However, humans do not always face the camera — indeed, good portrait photographs rarely have the subject posed full face. This is arguably the most common failure mode of the algorithm.

Illumination. Illumination plays a surprisingly significant role in deciding the image quality and the evaluation time to detect a face and thus on the effectiveness of Viola-Jones. For example, a back-lit subject photographed on a bright day means that their face is largely in shadow and hence impedes the detection of certain features of the face [Viola and Jones, 2002]. Similarly, a dark image with low contrast causes difficulty in detecting variations the intensity level of the face as shown in Figure 3.

 ISBN 978-0-9934207-2-6

Occlusion of the face. The Haar features used to classify a detected object as a face or not depend on the variation of intensity of different parts of a face. The presence of an occluding object in front of a face prevents the determination of the features needed to detect it. Spectacles with obtrusive frames that are much lighter or darker then skin also can cause a similar effect.

Facial expression. In face recognition, a discrepancy may occur between a recorded face and the facial expression of a subject [Valstar et al., 2015]. Although Viola-Jones is locating rather than recognising faces, extreme facial expressions also cause faces to be missed.

Figure 3: Face images demonstrating illumination variation [Chaudhari et al., 2015]

4 Colour spaces

A *colour space* is a specific way of representing a set of colours. The most common colour model, the one used by most computer displays and digital cameras, is red–green–blue (RGB). However, other colour spaces abound, with acronyms such as YIQ, YUV, YC_bC_r and CMYK [Sural et al., 2002, Ganesan et al., 2015]. This work has investigated two specific colour spaces related to an artist's notion of hue, saturation and brightness, namely HSV and HLS [Setiawan et al., 2006], both of which can be obtained by a simple transformation of RGB. The principal difference between HLS and HSV is the calculation of the brightness component (L or V), which determines the distribution and the range of the brightness and the saturation (S) [Sural et al., 2002, Setiawan et al., 2006].

5 McNemar's test

McNemar's test is a statistical test that can be applied to a pair of algorithms to explore where one is more effective than the other [Kanwal, 2013]. For each region under consideration, one identifies the outcomes, success or failure, reported by the two algorithms. This allows one to build up a 2×2 'truth table' as shown in Table 1, where N_{ss} represents the number of times that both algorithms succeeded, and so on. Of particular interest are the number of cases in which the first algorithm succeeded and the second failed, N_{sf}, and *vice versa*, N_{fs} — it is only those cases in which the algorithms performed differently that are of interest.

More precisely, McNemar's test involves computation the so-called Z-score using:

$$Z = \frac{|N_{sf} - N_{fs}| - 1}{\sqrt{N_{sf} + N_{fs}}} \tag{7}$$

where the -1 is a continuity correction. Clearly, if the performances of the two algorithms being considered are identical, $Z = 0$ when $N_{sf} = N_{fs}$, and its value increases as the number of discrepancies in performance increase. Confidence limits can be associated with the value of Z, and the one most commonly used is 1.96, which indicates that there is a probability of 0.05 (*i.e.*, one in twenty) that the results obtained could be an artefact of the data used [Yimyam and Clark, 2016, Clark and Clark, 2003].

		algorithm A	
		failure	success
algorithm B	failure	N_{ff}	N_{sf}
	success	N_{fs}	N_{ss}

Table 1: 'Truth table' for McNemar's test

 ISBN 978-0-9934207-2-6

6 Experimental work

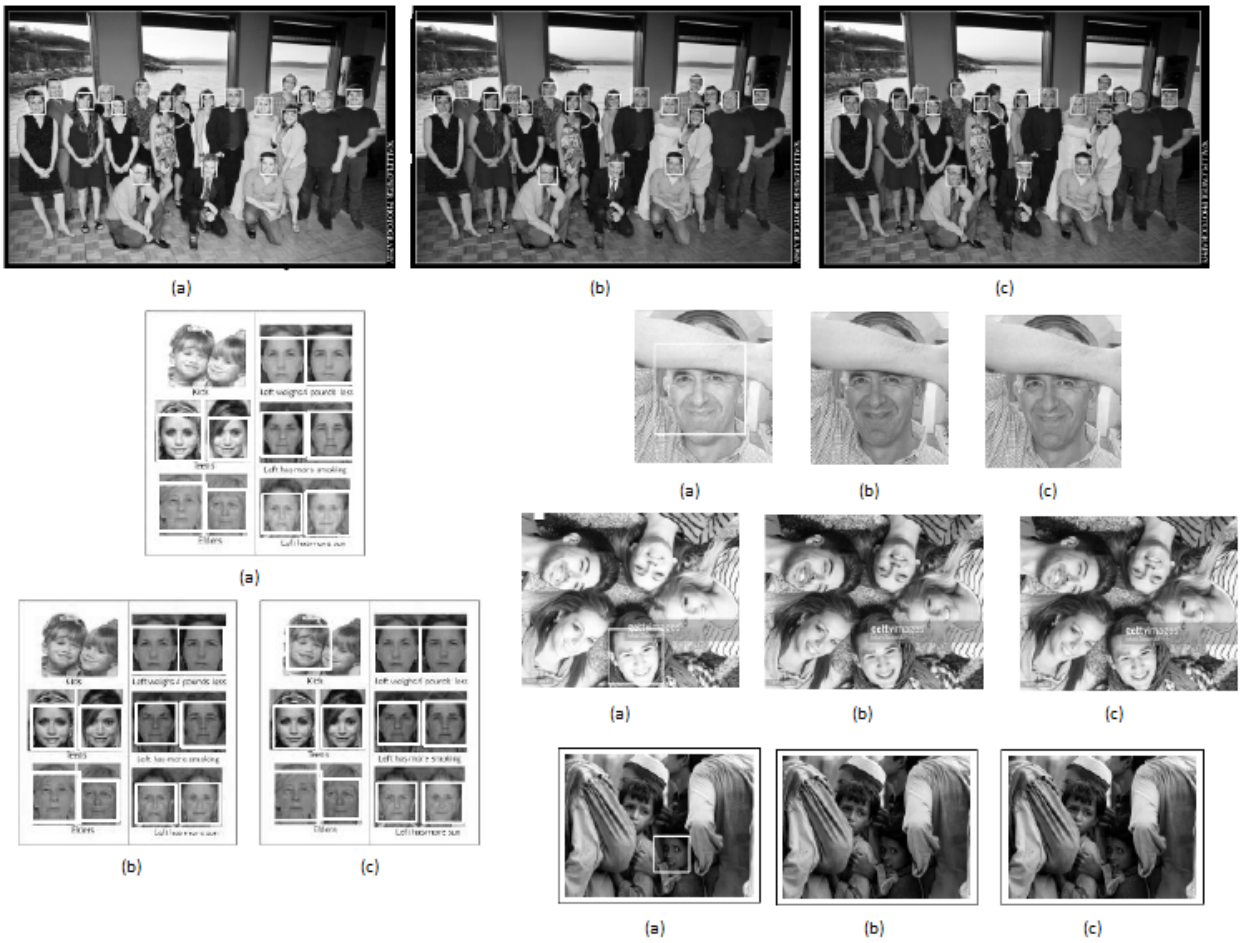


Figure 4: The Viola-Jones approach applied to different colour-space channels. (a) output when applied to V-channel image. (b) output when applied to L-channel image. (c) output when applied to grey-scale image

Databases used for face recognition typically contain a single face, so there are no scene components that could potentially confuse Viola-Jones. Hence, this work has assimilated a database of group photographs, each of which contains more than one face; the total number of faces in all the images is 2,250. Different lightings have been cast on each image. Each of these images has been transformed to three different single-channel representations: grey-scale, lightness (L of HLS) and value (V of HSV). In each case, all of the images were transformed to the relevant colour space before being passed to the Viola-Jones algorithm, and all the detected faces were recorded. These detected faces were then compared with the known locations of all the faces in all the images, and used to determine whether the face had been found or not, and to establish whether non-face regions had been identified as faces (false positives). Figures 4 and 5 show some of the images with detected faces.

In experiments of these kind, it is common to quote the number of true positives *etc*; but despite their widespread use these numbers actually have little meaning. In establishing whether there is a genuine, statistically-significant performance difference in applying Viola-Jones to images obtained from different colour spaces, a more meaningful approach is to employ a statistical test. There is no good reason to suppose that the statistics of the errors are Gaussian, so it would be inappropriate to use analysis of variance (ANOVA); similarly, there is

 ISBN 978-0-9934207-2-6

Figure 5: Viola-Jones applied to different images with different lightness $(+10, +20, -10, -20\%)$. (a) Viola-Jones applied on V-channel image. (b) output when applied to L-channel image. (c) output when applied to grey-scale image

no reason for the error distribution to be symmetric, ruling out the Wilcoxon test. Under these circumstances, the best available test is McNemar's test, which forces the underlying distribution to be binomial by way of doing a series of pairwise comparisons.

Results obtained from the comparisons using McNemar's test of the original Viola-Jones algorithm with

 ISBN 978-0-9934207-2-6

colour spaces	N_{sf}	N_{fs}	Z
V and grey	146	57	6.176
L and grey	82	44	3.296
V and L	45	19	4.238

Table 2: Z-values of Viola-Jones applied on grey-scale, H, and L images

the V-channel of HSV space and the L-channel of HLS space are shown in table 2. Both of these are well above $z_{crit} = 1.96$, so one can conclude that, with a probability > 0.95, the Viola-Jones algorithm is more effective if the image is first transformed to the HSV or HLS colour space and the appropriate brightness channel of that space (H, L channels) is used. The figures also indicate that the V-channel of HSV is more effective than the L-channel of HLS by a statistically-significant amount.

7 Conclusions

This work has shown that the well-established Viola-Jones algorithm can be made more effective by applying it to the brightness channel of a transformed colour image rather than the conventional grey-scale image obtained from a colour one. Moreover, the V-channel of HSV is more effective then the L-channel of HLS. This means that the simple, fast transformation from RGB pixels to HSV as a precursor to using conventional Viola-Jones face detection should yield significantly fewer false positives and negatives, a significant improvement in its effectiveness.

References

[Chao, 2007] Chao, W.-L. (2007). Face recognition. *GICE, National Taiwan University*.

[Chaudhari et al., 2015] Chaudhari, M., Sondur, S., and Vanjarel, G. (2015). A review on face detection and study of viola jones method. *International Journal of Computer Trends and Technology (IJCTT)*, 25(1):54–61.

[Clark and Clark, 2003] Clark, A. F. and Clark, C. (2003). Performance characterization in computer vision: A tutorial. Technical report, University of Essex, UK.

[Ganesan et al., 2015] Ganesan, P., Rajini, V., Sathish, B., Kalist, V., and Basha, S. K. (2015). Satellite image segmentation based on ycbcr color space. *Indian Journal of Science and Technology*, 8(1):35–41.

[Jones and Viola, 2006] Jones, M. J. and Viola, P. (2006). Method and system for object detection in digital images. US Patent 7,099,510.

[Kanwal, 2013] Kanwal, N. (2013). *Low-level image features and navigation systems for visually impaired people*. PhD thesis, University of Essex.

[Klare et al., 2015] Klare, B. F., Klein, B., Taborsky, E., Blanton, A., Cheney, J., Allen, K., Grother, P., Mah, A., and Jain, A. K. (2015). Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1931–1939.

[Schroff et al., 2015] Schroff, F., Kalenichenko, D., and Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823.

 ISBN 978-0-9934207-2-6

[Setiawan et al., 2006] Setiawan, N., Seok-Ju, H., Jang-Woon, K., and Chil-Woo, L. (2006). Gaussian mixture model in improved hls color space for human silhouette extraction. *Advances in Artificial Reality and Tele-Existence*, pages 732–741.

[Sural et al., 2002] Sural, S., Qian, G., and Pramanik, S. (2002). Segmentation and histogram generation using the hsv color space for image retrieval. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 2, pages II–II. IEEE.

[Valstar et al., 2015] Valstar, M. F., Almaev, T., Girard, J. M., McKeown, G., Mehu, M., Yin, L., Pantic, M., and Cohn, J. F. (2015). FERA 2015 — second facial expression recognition and analysis challenge. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 6, pages 1–8. IEEE.

[Viola and Jones, 2001] Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE.

[Viola and Jones, 2002] Viola, P. and Jones, M. (2002). Fast and robust classification using asymmetric adaboost and a detector cascade. *Advances in neural information processing systems*, 2:1311–1318.

[Viola and Jones, 2004] Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *International journal of computer vision*, 57(2):137–154.

[Yimyam and Clark, 2016] Yimyam, P. and Clark, A. F. (2016). 3D reconstruction and feature extraction for agricultural produce grading. In *Knowledge and Smart Technology (KST), 2016 8th International Conference on*, pages 136–141. IEEE.

[Zhao et al., 2003] Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM computing surveys (CSUR)*, 35(4):399–458.

Tahitian Pearls' Lustre Assessment

G. Mondonneix, S. Chabrier, J.M. Mari, A. Gabillon, J.P. Barriot

Université de Polynésie française, Tahiti
Laboratoire d'excellence CORAIL
Géopôle du Pacifique sud EA4238

Abstract

This paper represents a preliminary work toward a machine learning process which could be used to automatically assess Tahitian pearls' lustre. In particular, it investigates the different aspects of lustre which could be used to design feature vectors for machine learning algorithms.

Keywords: Pearl, Lustre, Features, Perception

1 Introduction

Tahitian pearls represent the first exportations of French Polynesia in terms of income[1]; this income is currently growing[2]. Millions of pearls are assessed each year by experts. On such a highly competitive market, an automated assistance could bring a crucial advantage to French Polynesia: under the RAPA[3] project, we undertook research on automatic measures of the thickness of nacre [3] and on characterizing pearls' colour [4], [5]. This paper presents our preliminary work on lustre automated assessment.

In section 2, we review the physical aspects of pearls, pointing out peculiarities of Tahitian pearls from an optical point of view, like the tendency to a stronger iridescence or darker colours than other pearls. In section 3, we address the notion of lustre and the distinction between its physical and perceptual dimensions, which do not necessarily correlate. The former consists of the optical phenomenon of specular reflection on nacre, while the latter consists of the perception of this phenomenon. In section 4, we review perceptual aspects of lustre. These aspects can serve to design feature vectors which could be used in a machine learning algorithm to reproduce the way about how experts assess lustre.

2 Physical and optical properties of nacre

A pearl is made of a nuclei coated with mother of pearl, or nacre, so that the physical aspects of its lustre borrow to the optical properties of its nacre. A complete view of nacre's structure can be found in [6]. Nacre is composed of aragonite plates (crystals about 0.5 μm thick) bound together by conchiolin (organic matter secreted by molluscs). These plates are structured in parallel layers; a quick computation shows that nacre contains about 2000 layers per millimetre thick. Aragonite is transparent and birefringent, whereas conchiolin contains pigments. Their structuration in parallel layers determines some optical properties of pearls [7]. As such, iridescence[4], sometimes called 'orient' in the specific case of pearls, is created by multiple reflections and refractions of light on the parallel interfaces formed by the interleaved layers of aragonite and conchiolin: the more the layers, the stronger

[1] 8,8 Billion F.CFP in 2014, representing 69% of the exportations of French Polynesia [1].
[2] More than 12% of growth between 2013 and 2014 [1].
[3] RAPA [2] stands for "Reconnaissance Automatique de la qualité des Perles de TAhiti".
[4] Iridescence (or 'goniochromism', or even sometimes referred to as 'perlescence') is the "interference of light either at the surface or in the interior of a material that produces a series of colours as the angle of incidence changes" [8]. This can be seen on thin layers of oil for example.

 ISBN 978-0-9934207-2-6

the 'orient'. As well, aragonite plates are interleaved with conchiolin, yet conchiolin contains pigments, thus, the more the layers, the more the pigments, and darker are the pearls. Aragonite is a birefringent material; nevertheless, a camera fitted with a polarized filter does not allow detecting polarization due to birefringence on a pearl. The only polarization it detects stands on pearls' boundaries when light is reflecting at grazing angle; however, this observation is explained through Fresnel's equations and is not specific to pearls: the very same observation is done on artificial pearls. It is worth noting that nacre thickness is not necessarily uniform over the nucleus, thus lustre is not necessarily homogeneous over the surface of the pearl.
Tahitian pearls come from a mollusc called *pinctada margaritifera*. The conchiolin secreted by this mollusc contains black pigmentation [9]. Moreover, its nacre has the highest texture index [10] among other molluscs, resulting in more compact layers, that is, more layers per unit of nacre thickness. As a consequence, Tahitian pearls are darker and, for an equal thickness of nacre, exhibit a stronger orient than other pearls.

3 Notion of lustre

Definition: Lustre commonly refers either to an objective notion, as reflected high-lights on a surface[5], or to a subjective notion, as the perception we have of these reflected high-lights[6]. For the purpose of the present paper, let the former definition be qualified as 'physical' and the latter one as 'perceptual'. Although synonym of gloss[7], 'lustre' is preferred to 'gloss' when one wants to stress the material out of which the reflecting surface is made. This usage is well illustrated in cases of minerals or fabrics. In the specific case of pearls, lustre is related to the layered structure of nacre[8]. In this regard, some pearls' experts state that lustre is the ability to deeply reflect light, through these layers. The Gemological Institute of America (GIA) gives an incident definition of pearls' lustre [16] by providing a list of adjectives that can qualify pearls' reflections in order to assess it[9]. More specifically to Tahitian pearls, the *Assemblée de la Polynésie française*[10] (APF), in a normative text [17], states that the term 'lustre' can be replaced by 'gloss', and defines it as the "more or less perfect" reflection of light on the surface of the pearl. An excellent lustre is said to correspond to the total reflection of the light on the pearl's surface and the ability to reflect images like a mirror, whereas no lustre would correspond to a matt finish of the surface. According to both the GIA and the APF, the quality of lustre depends on the physical parameters of the nacre: the GIA relates the poor quality of lustre to an insufficient thickness of the nacre, while the APF states that the quality of lustre depends on both the thickness and the structure of the nacre.
The notion of lustre may seem to be complex if not ambiguous (physical/perceptual duality), and the presence of a normative definition of Tahitian pearls' lustre adds to the difficulty of deciding what definition to use. Nevertheless, *the goal when investigating Tahitian pearls' lustre assessment is not to decide what the word 'lustre' should mean but to capture what is actually done when a Tahitian pearl is being assessed regarding a characteristic called 'lustre'*. From this point of view, the definition allowing capturing the most information about this process should be the broadest one, i.e., lustre is the *appearance* of *reflected high-lights on the surface of a pearl*. From this point of view, lustre is a particular realization of gloss, as defined by the *Compagnie Internationale de l'Eclairage* (CIE) [18].
Duality of lustre: Scientific insights on lustre can be found in the research literature about gloss. Even though this literature is not specific to lustre, and *a fortiori* to Tahitian pearls' lustre, it applies to it since lustre is an instance of gloss. Gloss is commonly reduced to specular reflectance, and measuring instruments like glossmeters simply

[5] E.g., "the brightness that a shiny surface has" (Cambridge dictionary) [11], "the manner in which the surface … reflects light" (Oxford dictionary) [12].
[6] E.g., "the appearance of a … surface in terms of its light-reflective qualities" (Encyclopaedia Britannica) [13].
[7] E.g. [14], [15]
[8] As such, according to the Encyclopaedia Britannica, lustre "results from the repeated reflections from minute cleavage cracks" [13].
[9] 'bright', 'sharp' or 'distinct' for high lustre quality; 'weak', 'hazy', 'blurred', 'dim' or 'diffused' for low lustre quality
[10] Assembly of French Polynesia

 ISBN 978-0-9934207-2-6

measure specular reflectance at various angles[11]. Nonetheless, literature shows that physical and perceptual aspects of gloss evolve in distinct spaces that do not necessarily correlate. Indeed, despite common sense, gloss is not equivalent to specular reflectance. In other terms, specular reflectance is neither a sufficient nor even a necessary condition of gloss. First, the presence of specular reflectance is not sufficient to determine the presence of gloss. Indeed, materials exist for which specular reflectance is not perceived as gloss: snow for example, when seen at grazing angle, is not glossy, yet has a high specular reflectance[12]. Second, the presence of specular reflectance is not necessary to determine the presence of gloss. Indeed, it is possible to generate gloss using only diffuse light modulation [22]. To sum up, it is possible that specular reflectance be not perceived as gloss as well as that gloss be perceived despite the absence of specular reflectance. This demonstrates that gloss, hence lustre, cannot *a priori* be reduced to specular reflectance. Lustre has then to be studied as a perceptual phenomenon, and it is worth identifying the visual aspects coming into play in lustre assessment.

4 Aspects of lustre

Pearls exist as physical objects and, as such, exhibit optical properties. Yet they exist for an expert through its perceptual system and we have seen in the previous section that physical and perceptual properties do not necessarily coincide. The perceptual attributes of lustre have thus to be reviewed. These attributes, once extracted into feature vectors, can be used to automatically discover the underlying mechanisms of lustre perception. Hunter[23] provides a list of types of gloss: *specular gloss*, *sheen*, *contrast gloss*, *distinctness-of-reflected-image gloss*, *absence-of-bloom gloss* and *absence-of-surface-texture gloss*. We illustrate how these aspects are likely to allow grasping lustre as a perceptual phenomenon and emphasize the independence between them. In addition, we identify two aspects more specific to pearls; we call them *iridescence* and *deep reflectance*.

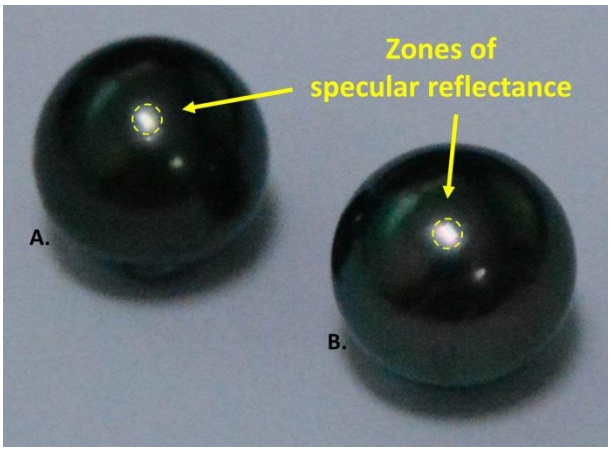


Figure 1: Two pearls of different *specular gloss* (maximal intensity values of the image in an HSV colour space: 93.33% for pearl A; 100% for pearl B).

Specular gloss: *Specular gloss* is the perceived brightness of a surface due to specular reflectance at non grazing angles with respect to incident light. Figure 1 shows two pearls of different *specular gloss*.

Numerical values corresponding to *specular gloss* could be the ratio of the incident intensity to the specular reflected intensity at non grazing angles or, if the incident intensity is constant, only the specular reflected intensity.

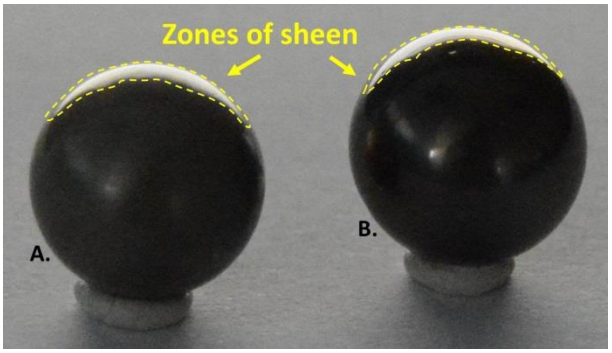


Figure 2: Two pearls of different *sheen* (maximal intensity values of the image in an HSV colour space: 96.47% for pearl A; 99.22% for pearl B).

Sheen: *Sheen* is the perceived brightness of a surface due to specular reflectance at grazing angles with respect to incident light. Figure 2 shows two pearls of different *sheen*. *Specular gloss* and *sheen*, though both expressing the strength of specular reflectance, are not necessarily dependent. Cases where there are both *specular gloss* and *sheen*, as well as cases where there is neither of them,

[11] For an example of glossmeter, see [19]

[12] [20] as cited by [21].

 ISBN 978-0-9934207-2-6

are commonly found. The case where there is *sheen* but no *specular gloss* is justified by Fresnel equations[13] and can be seen in flat wall paints. Eventually, the more counter intuitive case (because apparently contradicting Fresnel equations), where there *is specular gloss* but no *sheen*, has been found in "a number of yarn and paper samples which possessed a fuzziness that caused them to appear matt if viewed at near grazing angles" [23].

Numerical values corresponding to *sheen* can be obtained the same way as for *specular gloss*, but at grazing angles.

Contrast gloss: *Contrast gloss* is the perceived brightness of a surface due to the contrast between specular reflectance and diffuse reflectance. Figure 3 emphasizes the difference between *contrast gloss* and *specular gloss*: pearl B has a higher *contrast gloss* than pearl A[14], yet pearl A has a higher *specular gloss* than pearl B[15]. *Contrast gloss* and *specular gloss* both refer to the strength of reflected light at specular angle. However, they have different definitions: the former is defined with respect to incident light while the latter is defined with respect to diffuse reflectance. These two definitions express two different conceptions of gloss as a perception: *specular gloss* implies the capability of assessing specular reflectance given a reference that is not necessarily in the proximity of the specular angle, or even not directly accessible in the scene, while *contrast gloss* consists of assessing specular reflectance given a reference that is in the immediate proximity of the specular angle. In other words, unless considering *specular gloss* as mere specular reflectance[16], in which case its assessment would be based on references independent of the scene, *specular gloss* assessment should be based on actual or even estimated references from a global view of the scene, while *contrast gloss* assessment could be based only on actual references from a local view of the scene[17].

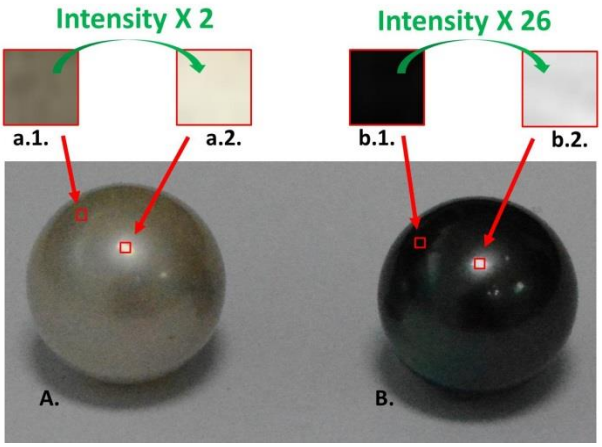


Figure 3: Two pearls of different *contrast gloss.* Pearl A has a lower contrast gloss than pearl B.

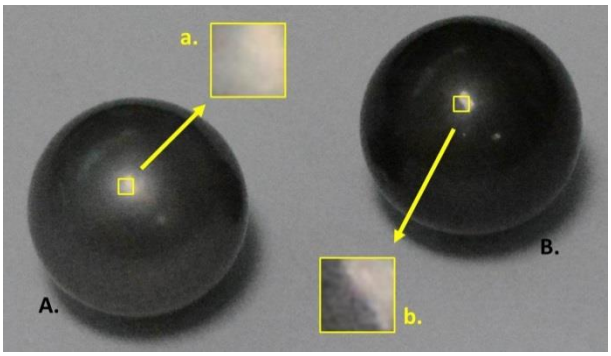


Figure 4: Two pearls of different *distinctness-of-reflected-image gloss.* Pearl A has a lower *distinctness-of-reflected-image gloss* than pearl B.

Numerical values corresponding to contrast gloss could be the ratio of specular reflected intensity to diffuse reflected intensity, or the ratio of the difference between specular and diffuse reflected intensities to the incident intensity.

Distinctness-of-reflected-image gloss: *Distinctness-of-reflected-image gloss* (DOI) is the perceived brightness of a surface due to the sharpness of the specular reflected light. The sharpest the specular reflected light, the more the surface behaves like a mirror. This aspect of gloss is sometimes called *mirror-like effect*. Figure 4 shows two pearls with different *distinctness-of-reflected-image gloss* qualities. The contour of the specular reflectance zone is

[13] *Sheen* is related to the reflected light at grazing angle (large angle of incidence) while *specular gloss* is related to reflected light at non grazing angle (low angle of incidence); yet, according to Fresnel equations, the fraction of incident light that reflects from the surface is higher when the angle of incidence is larger [24].

[14] In an HSV colour space, the mean intensity value of a.2. is twice the one of a.1., while the one of b.2. is 26 times the one b.1.

[15] Maximal intensity values of the image in an HSV colour space: 93.73% for pearl A; 93.33% for pearl B.

[16] I.e., *specular gloss* would be simply defined as reflected light whose intensity is greater than a given threshold, no matter the intensity of incident light. This basic conception of gloss is sometimes adopted, although such an approach is very much open to criticism (see (Chadwick) on (Shimomura)).

[17] It can be noticed that these conditions of assessment make *contrast gloss* prone to gloss consistency.

 ISBN 978-0-9934207-2-6

sharper on the pearl B, exhibiting a better *distinctness-of-reflected-image gloss*[18]. As well, figure 4 emphasizes the difference between d*istinctness-of-reflected-image gloss* and *specular gloss*: pearl B has a higher d*istinctness-of-reflected-image gloss* than pearl A, yet pearl A has a higher *specular gloss* than pearl B[19].

A numerical value corresponding to DOI could be the first derivative of the specular reflected intensity with respect to the angle of reflected light. On an image, it can be related to the magnitude of the gradient of the intensities around the specular reflectance zones.

Absence-of-bloom gloss: *Absence-of-bloom gloss* is the perceived brightness of a surface due to the absence of haze around specular reflected high-lights. As an illustration, in figure 5, pearl B has almost no haze compared to the two other pearls; as such, it can be said to have a better *absence-of-bloom gloss*. *Absence-of-bloom gloss* and *distinctness-of-reflected-image gloss* both relate to quality of reflected image. However, they account for two different aspects of it. The former accounts for how well the reflected image preserves intensity amplitudes around the specular angle, while the latter accounts for how well the reflected image preserves edge sharpness. On figure 5, pearl A and B both exhibit a high *distinctness-of-image gloss* compared to pearl C[20], yet pearl B has better *absence-of-bloom gloss* than pearl A[21].

Numerical values corresponding to *absence-of-bloom gloss* could be the ratio of the surface of haze to the surface of specular reflectance it surrounds. The difficulty is however to determine the exact surface of haze automatically. This feature could be learned in a supervised mode.

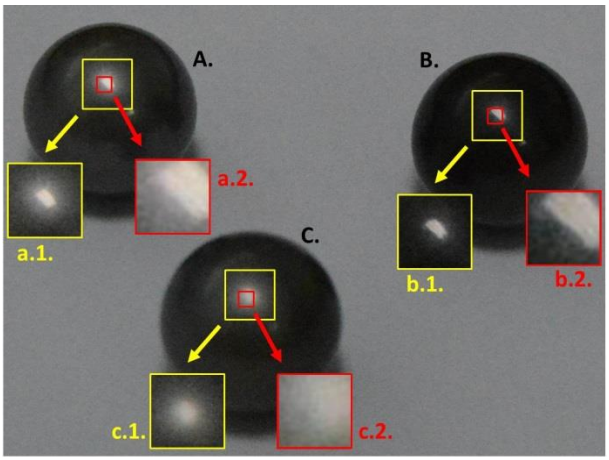


Figure 5: Pearls of different *absence-of-bloom gloss*. Pearl A and C have a lower *absence-of-bloom gloss* than pearl B.

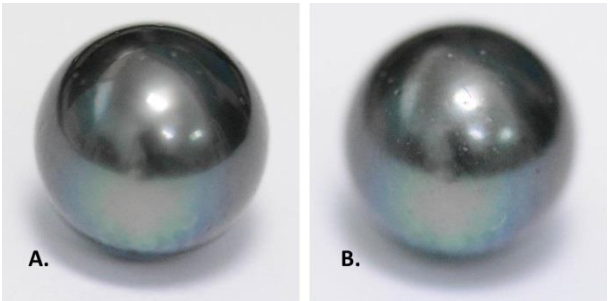


Figure 6: Pictures of a same pearl taken with two different focal distances (A focuses on the reflected image; B focuses on the surface of the pearl).

Absence-of-surface-texture gloss: *Absence-of-surface-texture gloss* is the perceived brightness of a surface due to the absence of interference between the reflected image and the image of the surface itself. Figure 6 illustrates the two images that can be formed out of a single pearl when its surface exhibits irregularities. Figure 6A focuses on reflected image while figure 6B focuses on the surface texture. Because the eyes can switch from one view to another, it creates an effect lessening the glossy appearance of the pearl.

Absence-of-surface-texture gloss seems related to the quality of the surface, which is another criterion used by the experts to assess the quality of pearls, so, numerical values corresponding to *absence-of-surface-texture gloss* may be derived directly from a measure of the quality of the surface.

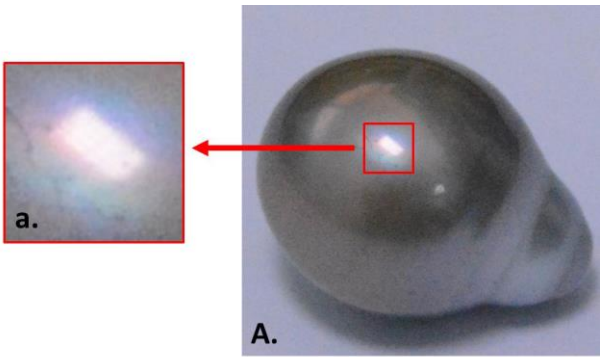


Figure 7: Some pearls exhibit colour changes around the specular angle, due to iridescence.

[18] An edge is visible between the left and the right parts of b., but not of a.

[19] Maximal intensity values of the image in an HSV colour space: 87.84% for the left pearl; 82.75% for the right pearl.

[20] An edge is visible between the left and the right parts of a.2. and b.2., but not of c.2.

[21] The difference of intensity between the left and the right parts of a.2. is lower than of b.2., but the edge between these parts is not sharper in b.2. than in a.2.

 ISBN 978-0-9934207-2-6

Iridescence: Since changes in wavelength due to specular reflection are negligible for dielectric materials, specular reflectance is usually studied as being achromatic, while colour is related to diffuse reflectance only. However, it has been seen that nacre exhibits iridescence [7], yet iridescence is a peculiar case since it takes form through colour changes, but is due to specular reflection. Figure 7 illustrates this aspect, with a zoom on the specular reflectance area, where parallel elongated coloured zones are visible, like superimposed. In some cases, rotating the pearl makes these zones rotate in the same direction. This anisotropic behaviour is observable on ringed pearls, like the one on figure 7.

Numerical values corresponding to *iridescence* could be the variance of chromaticity on the zone of the surface covered by the specular reflectance and surrounding haze.

Deep reflectance: Figure 8 shows the difference of reflectance between a fake pearl made of plastic (A) and a cultured pearl (B). Both pictures are taken in the exact same conditions. The fake pearl exhibits only a white reflectance zone while the cultured one exhibits both white and yellow reflectance zones. The reflectance split on the cultured pearl is not clearly visible to the naked eye, which only perceives an effect of 'deep' reflection when the pearl is rotated. Taking a picture of the pearl using a very short exposure time helps making it visible. Indeed, if exposure time is too long, the two reflectance zones merge in a single, larger reflectance zone, corresponding to the observation made with naked eye. Contrary to iridescence, where colours vary with the illumination and observation angles, the secondary zone here stays yellow regardless of the illumination or observation angles. A probable explanation is that, since conchiolin contains pigments, the different layers constituting nacre act like an absorbing filter: only part of the incident spectrum is specularly reflected from the deeper layers, the rest being absorbed by the successive layers of conchiolin. Figure 9 shows the same pearl as the one shown figure 7. The picture however is taken with a very short exposure time. One can observe both iridescence (b.3 to b.5) and deep reflectance (b.1).

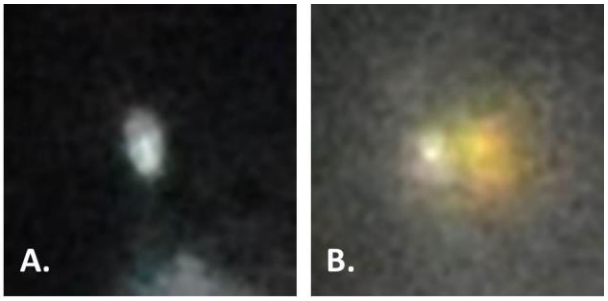


Figure 8: Detail of reflectance on two pearls (on the left, an artificial pearl; on the right, a real pearl).

Numerical values corresponding to *deep reflectance* could be computed using pictures taken using a very short exposure time. The ratio of the intensity of the deep reflectance point to the intensity of surface reflectance point could be a candidate. Furthermore, it might be that the distance between these two points has some impact.

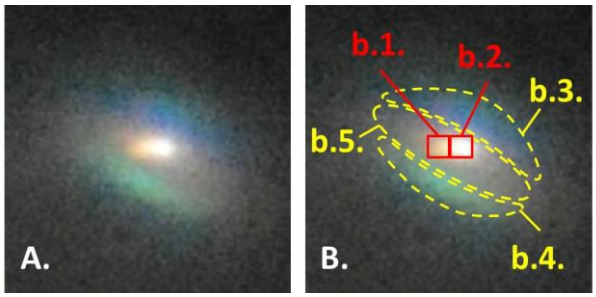


Figure 9: Difference between iridescence and deep reflectance.

5 Conclusion

In this paper, we present eight aspects of lustre which may be used to design feature vectors in a machine learning perspective for automatically assessing pearls' lustre. Each of these aspects is explained and illustrated and the distinctions between aspects are emphasized. Furthermore, with the goal of designing logical descriptors of lustre extractable from images of pearls, ways to obtain numerical values corresponding to these aspects are discussed. We have now to develop a protocol for images acquisition that ensures reproducibility and estimate the minimal number of samples needed for a machine learning process.

In addition to automatizing pearl's lustre assessment, machine learning could allow, by looking at the weights distribution yielded by the fitting process, to investigate the actual impact of each aspect of lustre on pearl's lustre assessment, to better understand how human assessment of lustre works.

 ISBN 978-0-9934207-2-6

6 Acknowledgements

The present paper could not have been done without the experts who accepted to talk about their experience or take part to experiments: Aline and Jean-Luc Baldassari, Philippe Chenne from *Tahiti Rava Rava Pearl*, Heinarii Haoatai, Heipua Lu Look from *L'atelier de la perle*, Eric Sichoix, Richard Wan from the *Groupe Robert Wan*, Loïc Wiart from *Poe Black Pearl*, Marie-Laure Tang, and the experts from the *Direction des ressources marines et minières*, *Blue Pearl*, *Mihiarii Pearls*, and *Vaima perles*.
The RAPA project is funded by the French ministry for Overseas Territories.

7 References

[1] Institut de la statistique de Polynésie française, "Bilan de la perle 2014."
[2] "RAPA project." [Online]. Available: https://sites.google.com/site/rapaproject/. [Accessed: 06-Dec-2016].
[3] M. Loesdau, S. Chabrier, and A. Gabillon, "Automatic Nacre Thickness Measurement of Tahitian Pearls," in *Image Analysis and Recognition*, vol. 9164, M. Kamel and A. Campilho, Eds. Cham: Springer International Publishing, 2015, pp. 446–455.
[4] M. Loesdau, S. Chabrier, and A. Gabillon, "Hue and Saturation in the RGB Color Space," in *Image and Signal Processing*, vol. 8509, A. Elmoataz, O. Lezoray, F. Nouboud, and D. Mammass, Eds. Cham: Springer International Publishing, 2014, pp. 203–212.
[5] M. Loesdau, S. Chabrier, and A. Gabillon, "Automatic Classification of Tahitian Pearls," in *Image Processing & Communications Challenges 6*, vol. 313, R. S. Choraś, Ed. Cham: Springer International Publishing, 2015, pp. 95–101.
[6] F. Marin, G. Luquet, B. Marie, and D. Medakovic, "Molluscan Shell Proteins: Primary Structure, Origin, and Evolution," in *Current Topics in Developmental Biology*, vol. 80, Elsevier, 2007, pp. 209–276.
[7] M. R. Snow, A. Pring, P. Self, D. Losic, and J. Shapter, "The origin of the color of pearls in iridescence from nano-composite structures of the nacre," *Am. Mineral.*, vol. 89, no. 10, pp. 1353–1358, Oct. 2004.
[8] "iridescence | mineralogy | Britannica.com." [Online]. Available: https://www.britannica.com/science/iridescence-mineralogy. [Accessed: 15-Oct-2016].
[9] S. Elen, "Identification of Yellow Cultured Pearls from The Black-Lipped Oyster <i>Pinctada Margaritifera</i>," *Gems Gemol.*, vol. 38, no. 1, pp. 66–72, Apr. 2002.
[10] D. Chateigner, C. Hedegaard, and H.-R. Wenk, "Mollusc shell microstructures and crystallographic textures," *J. Struct. Geol.*, vol. 22, no. 11–12, pp. 1723–1735, Nov. 2000.
[11] "lustre Meaning in the Cambridge English Dictionary." [Online]. Available: http://dictionary.cambridge.org/dictionary/english/lustre. [Accessed: 13-Sep-2016].
[12] "lustre - definition of lustre in English from the Oxford dictionary." [Online]. Available: http://www.oxforddictionaries.com/definition/english/lustre. [Accessed: 13-Sep-2016].
[13] "lustre | mineralogy | Britannica.com." [Online]. Available: https://www.britannica.com/science/lustre. [Accessed: 13-Sep-2016].
[14] "Luster Synonyms, Luster Antonyms | Thesaurus.com." [Online]. Available: http://www.thesaurus.com/browse/luster. [Accessed: 26-Sep-2016].
[15] "Synonyms of lustre | Oxford Dictionaries Thesaurus." [Online]. Available: https://en.oxforddictionaries.com/thesaurus/lustre. [Accessed: 26-Sep-2016].
[16] "Pearl Quality Factors." [Online]. Available: http://www.gia.edu/pearl-quality-factor. [Accessed: 13-Sep-2016].
[17] *Délibération APF n°2005-42*. 2005.
[18] International Electrotechnical Commission, *Vocabulaire electrotechnique international. Eclairage: Vocabulaire international de l'éclairage = International electrotechnical vocabulary. Chapter 845, Lighting : International lighting vocabulary.* Genéve: Bureau Central de la Commission Electrotechnique Internationale, 1987.
[19] E. Rapaport, A. Nussinovitsch, and E. Mey-Tal, "Glossmeter," 25-Jan-2000.
[20] W. E. K. Middleton and A. G. Mungall, "The Luminous Directional Reflectance of Snow*," *J. Opt. Soc. Am.*, vol. 42, no. 8, p. 572, Aug. 1952.
[21] A. C. Chadwick and R. W. Kentridge, "The perception of gloss: A review," *Vision Res.*, vol. 109, pp. 221–235, Apr. 2015.

[22] R. Sève, "Problems connected with the concept of gloss," *Color Res. Appl.*, vol. 18, no. 4, pp. 241–252, Aug. 1993.
[23] R. Hunter S., "Methods of determining gloss," *Res. Pap. RP958*, vol. 18, Jan. 1937.
[24] J. Peatross and M. Ware, "Physics of Light and Optics: A Free Online Textbook," 2010, pp. 65–68.

 ISBN 978-0-9934207-2-6

Fast Video Processing Using a Spiral Coordinate System and an Eye Tremor Sampling Scheme

J. Fegan,[1] S.A., Coleman,[1] D. Kerr,[1] B.W., Scotney[2]

[1]*School of Computing and Intelligent Systems,*
[2]*School of Computing and Information Engineering,*
Ulster University, Northern Ireland

Abstract

In the advent of autonomous machines, the need for real time video processing is becoming an increasingly important issue. Although technological advances have brought us closer to achieving this goal, they are often based on expensive and uniquely designed hardware solutions. It can be argued that as the complexity of image processing increases, it becomes more desirable to focus on portability and cost effective processing strategies. In this paper, we present a biologically inspired processing strategy that can be integrated with common, cost-effective image hardware. The results demonstrate that this approach can achieve a six-fold speedup, against a traditional image processing strategy, without any hardware modifications and a ten-fold speedup on adapted hardware. Alongside this, we present a novel type of processing that is used to detect video features in a space-time continuum. The results of this also demonstrate real-time processing potential and appear promising for motion focused tasks such as robot navigation.

Keywords: Fast Video Processing, Spiral Coordinate System, Eye Tremor, Edge Detection, Space-time Processing

1 Introduction

Efficient video processing is essential in many important machine vision tasks where computer hardware is expected to operate on a stream of consecutive image frames under strict time constraints. The prevalent way to accomplish these tasks, where runtime performance is important, often relies on long-standing principles that do not reflect our current understanding of biological vision. For example, in Traditional Image Processing (TIP) a digital image is sampled on a rectangular lattice and stored as a matrix of picture elements (pixels) according to a two-dimensional (2D) Cartesian or raster coordinate system. By contrast, the Human Visual System (HVS) senses stimuli on a hexagonal lattice of light sensitive cells [1]. This observation has inspired a one-dimensional (1D) spiral coordinate system that is effective for storing images sampled on a hexagonal lattice [2]. Unfortunately, the benefits of this scheme, including fast image processing performance, are currently undermined by a lack of hardware that can capture hexagonal images, and the subsequent computational cost needed to map an image to a hexagonal pixel structure [3]. To circumvent these issues, the spiral coordinate system was adapted for traditional, rectangular hardware [3].

Building on the work in [4], this paper presents the application of a spiral coordinate system and a biologically inspired sampling procedure to conduct fast video processing. Here the implementation has been optimised to ensure high speed performance whilst maintaining accuracy. Furthermore, we extend the implementation for temporal processing based on a sparse ‘space-time neighbourhood’ operation which demonstrates promising initial results for future work. An overview of the spiral framework is presented in Section 2, with spatial processing being presented in Section 3. Section 4 introduces the concept of a ‘space-time neighbourhood’ and provides a set of preliminary results with the work being concluded in Section 5.

 ISBN 978-0-9934207-2-6

2 Image Representation

The traditional way to store an image using a 2D Cartesian coordinate system is intuitive but the pixels must be stored as a sequence of rows or columns. Consequently, the pixels are not kept in proximity with all of their nearest neighbours and this means that the pixels cannot be processed with their nearest neighbours in a linear sequence. By comparison, a spiral coordinate system allows some pixels to be stored beside their nearest neighbours in a contiguous vector and this formation can be exploited to improve the runtime performance of image processing algorithms.

2.1 Square Spiral Coordinate System

In the square spiral (Squiral) coordinate system a single coordinate is used to locate a point in 2D space. In this system, the origin (numbered 0) is at the centre of a region being sampled. The eight points surrounding the origin are numbered in an outward spiral, thus each number represents a cardinal or intermediate direction, for example:

{C, W, NW, N, NE, E, SE, S, SW} = {0, 1, 2, 3, 4, 5, 6, 7, 8}

In a similar way, a base 9 number is assigned to each point, such that a point's distance from the origin is determined by the position of the digits in its coordinate. For example, a pixel at coordinate 315 is nine (3^2) pixels north, three (3^1) pixels west, and (3^0) one pixel east from the origin. Further information on the Squiral Coordinate system can be found in [3], [5] and [6]. In accordance with this system, each pixel at a coordinate 0 mod 9 is stored beside its eight nearest neighbours in a contiguous vector. This simplifies and facilitates fast spatial processing because these pixels and their eight nearest neighbours can be traversed sequentially. However, it is difficult to process pixels with neighbours contained in multiple spiral regions because these pixels are not stored contiguously, for example pixels 1 and 15 in Figure 1. To overcome this difficulty an approach based on the simulation of involuntary eye movements called tremors was proposed in [7] where a series of images which incorporate small pixel shifts are used to facilitate processing across multiple spiral regions. We extend that approach here by adapting it for video sequences.

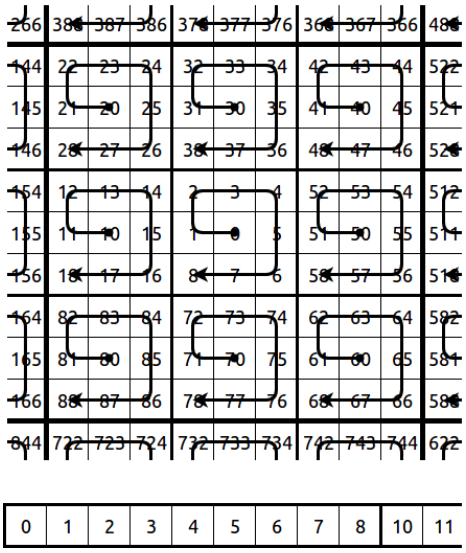


Figure 1: Squiral Coordinate System

2.2 Eye Tremor Frame Sampling

In video processing, the biological behaviour of eye tremor can be simulated by shifting the origin of the Squiral coordinate system, by one pixel, on each new frame. For example, Figure 2 shows a static 5x5 region where nine 3x3 frames are captured using the Squiral coordinate system. In the first frame (F_0) the origin of the Squiral coordinate system is located at the centre of the sampling lattice, and thus each pixel sampled at a coordinate 0 mod 9 is stored adjacent beside its eight nearest neighbours in a contiguous vector. In the next frame (F_1) the origin of the Squiral coordinate system is shifted left by one pixel. By doing this, the points that were previously coordinated 1 mod 9 assume the coordinates 0 mod 9 and are sampled in sequence with their surrounding neighbours. This process is repeated until a set of nine 'eye tremor' images are captured, one for each set of mod 9 pixels.

 ISBN 978-0-9934207-2-6

3 Spatial Processing

Spatial processing, commonly referred to as neighbourhood operations, describes an image operation where an output is computed by considering the properties of a pixel in relation to those of its surrounding neighbours. In TIP, all the pixels in an image are processed by considering the local neighbourhood area and a complete feature map output is obtained. This is not representative of the HVS, which sparsely interprets the visual stimuli it receives. In this section, a similar approach facilitated by the Squiral coordinate system and eye tremor sampling scheme is discussed.

3.1 Methodology

In this approach, only the central pixel in each spiral region is processed using its contiguous neighbours. As a result, the output pixels sparsely occupy one-ninth of a complete feature map. Therefore, the eye tremor-sampling scheme is used to 'focus' on a different mod 9 pixel in each spiral region allowing them to be sparsely processed in the same way. The outputs can be combined to produce a full-sized feature map. For example, Figure 2 demonstrates how a 3x3 image region can be sparsely processed across nine eye tremor frames (F_0 – F_8). In this situation, it takes nine initial frames to achieve a complete representation of the 2D scene. Thereafter each subsequent frame can be sparsely processed and combined with the output of the previous eight frames to construct a single, approximate feature map.

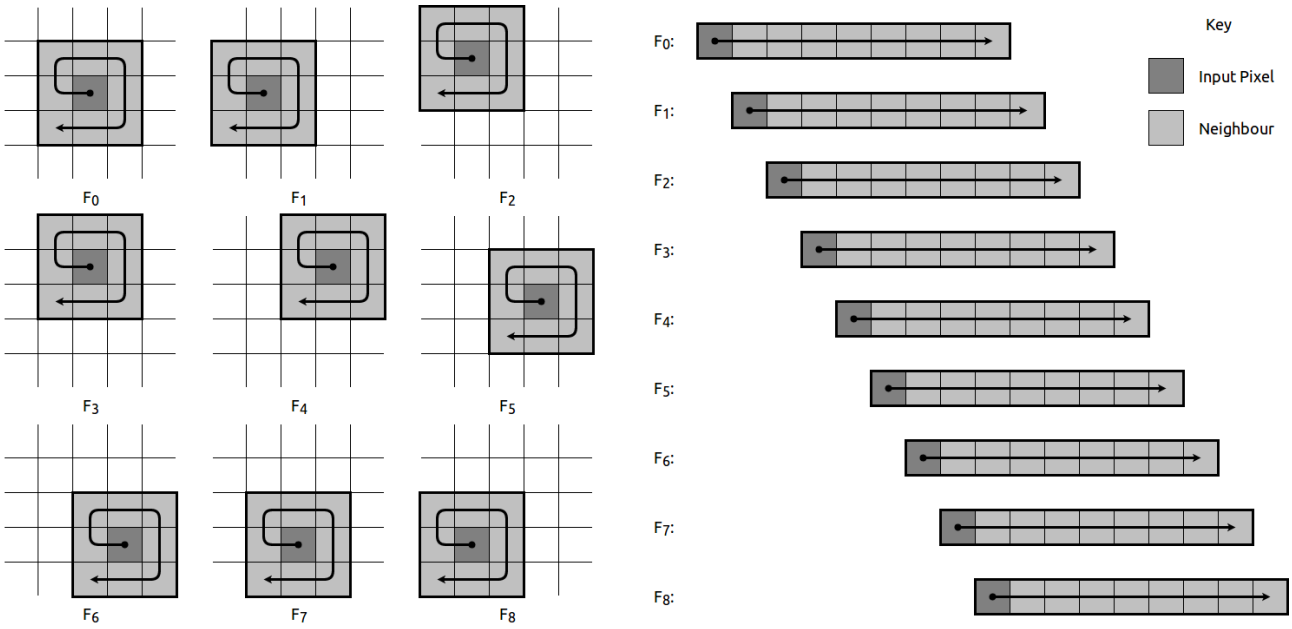


Figure 2: Spatial Processing Using Eye Tremor

3.2 Visual Evaluation

An implementation of the Sobel operator was tested on several videos and used to obtain features maps for a TIP approach and Squiral, eye tremor image processing approach. In this paper, ten frames (234 – 243) are shown from a video [8] which depicts a woman raising her right arm. The video was chosen for inclusion in this paper because it contains static and dynamic features that clearly illustrate the similarities and differences of the two processing approaches. The unaltered frames of the video are presented in Figure 3.1: the feature maps obtained using the TIP approach are shown in Figure 3.2; and the feature maps obtained using the Squiral, eye tremor approach are shown in Figure 3.3.

 ISBN 978-0-9934207-2-6

Figure 3.1: Arm Gesture

Figure 3.2: Traditional Feature Maps

Figure 3.3: Eye Tremor Feature Maps

In the eye tremor feature maps, the more relaxed features of the woman such as the face and torso closely resemble their counterparts in the traditional feature maps. However, there are some discrepancies around the lower arm and hand where there is fast movement. This is consistent with the feature maps from other test videos and indicates that the strength of a detected feature is affected by the rate of its spatial change. Based on this observation it is thought that a sparse eye tremor processing strategy will detect features more clearly if a higher framerate is used or if the spatial changes within the video sequence are small between frames. Regardless of these anomalies, the features detected in all of the test videos appear complete enough to support machine vision tasks.

3.3 Runtime Evaluation

The system used to measure the runtime performance had an Intel Core i7-4790 CPU @ 3.60GHz x 8, 16GB RAM and Ubuntu Linux 16.04 LTS 64-bit. The time taken to apply the Sobel operator to each and every frame of the video [8] at different resolutions are shown in Table 1. The times are measured in frames per second (fps) and correspond with a traditional and eye tremor implementation. The times given for the eye tremor implementation also list the time taken to map the 2D Cartesian frames to 1D Squiral frames; the time taken to map the 1D Squiral feature maps to 2D Cartesian feature maps (for display purposes); and the total time taken for all three actions. The timings show that a Squiral coordinate system and eye tremor sampling scheme can increase runtime performance significantly compared to a traditional implementation. At larger scales, such as layer 6, the speedup is almost 6 times faster than its traditional counterpart. If the overhead needed to map a traditional image to and from a Squiral image is removed the speedup is almost ten times faster.

Image Size		Traditional	Eye Tremor			
Layer	Pixels	Total	2D -> 1D	Edge Detection	1D -> 2D	Total
1	3x3	401,929fps	577,367fps	2,283,110fps	5,025,130fps	422,119fps
2	9x9	168,265fps	494,805fps	1,148,110fps	2,881,840fps	308,737fps
3	27x27	27,462fps	246,305fps	260,824fps	1,058,200fps	113,135fps
4	81x81	3,285fps	48,377fps	32,621fps	197,472fps	17,733fps
5	243x243	371fps	6,772fps	3,605fps	22,281fps	2,128fps
6	729x729	42fps	775fps	406fps	2,918fps	244fps

Table 1: Spatial Processing Runtimes

4 Space-time Processing

Previous research on the Squiral coordinate system combined with eye tremor sampling was primarily conducted on individual, static images, and it was only considered in the spatial domain. In video processing the influence of time is also considered. In most instances, temporal image processing examines how a pixel at a given location changes over time. In other words, a pixel is compared with a pixel at the same location in a previous or succeeding frame. In this section, we discuss a novel processing approach that incorporates spatial and temporal characteristics by using the vertical eye tremor processing strategy discussed in [6].

4.1 Methodology

In this space-time processing approach, the pixels in one frame are spatially processed using their neighbours in succeeding (future) frames. This idea is illustrated in Figure 4 where a pixel in the frame F_0 is processed using its eight spatial neighbours in the succeeding frames (F_1 – F_8). In practice, the Squiral, eye tremor frames are stacked to form a matrix and the pixels in the top row (F_0) are processed using the pixels that are parallel in the other rows. In this situation, a parallel pixel represents a spatial neighbour at a different point in time, a ‘space-time

neighbourhood' (Figure 4). In the implemented approach, a set of nine frames is needed before a single frame can be processed. Thereafter, each time a new frame is loaded, one-ninth of the top frame is processed. In other words, a set of mod 9 pixels are processed every time a new frame is loaded. The new frames are mapped to a second matrix. A limitation of this approach is that only one-ninth of the video frames can be processed. An alternative approach is to drop the oldest frame and append a new frame to the matrix. However, this presents a problem because only the first frame in a set of nine is vertically adjacent with its spatial neighbours. For example, the central pixel in F_1 does not have an immediate neighbour at the centre of $F_4 – F_6$. A possible solution to this problem is to use expensive base 9 computation or a lookup table similar to the one in [5] to locate a pixel's neighbours. This is considered a subject for further work.

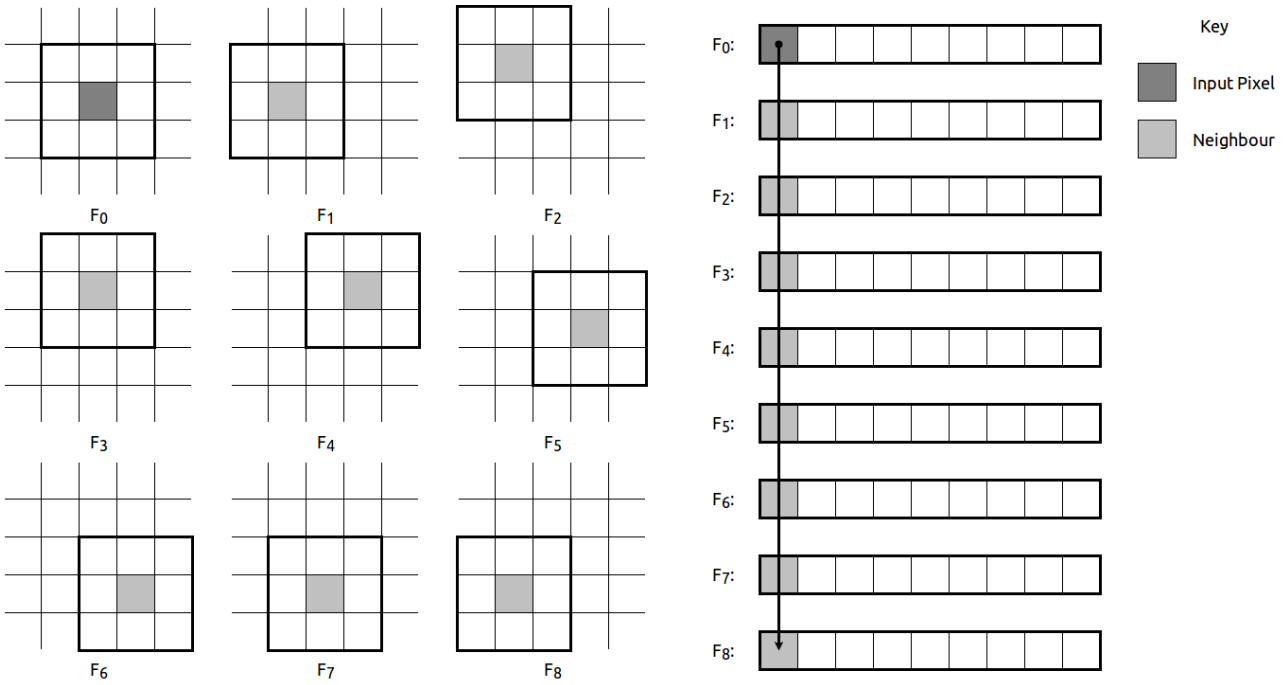


Figure 4: Space-time Processing Using Eye Tremor

4.2 Visual Evaluation

A set of features maps obtained from 'space-time neighbourhood' processing is shown in Figure 5. The Sobel operator was used with the system setup described in Section 3. In this example, it is visually apparent that the outline of the woman's right arm is the strongest detected feature. Therefore, it is hypothesized that feature detection in a space-time continuum will place more emphasize on features that change the most in space over time. Incidentally, in the complete video it is noticeable that some features such as the woman's shadow are identified more clearly in space-time. The usefulness of these unique space-time features is yet to be determined and will be considered in future research. Overall, it is thought that 'space-time neighbourhood' processing could be very useful in tasks where motion and run-time are key considerations.

 ISBN 978-0-9934207-2-6

Figure 5: Space-time Feature Maps

4.3 Runtime Evaluation

The runtime results for space-time image processing are shown in Table 2. The results indicate that this approach is slower than the eye tremor spatial processing results in Section 3, but they are still significantly faster than the TIP results at larger resolutions. This was expected, because a pixel's temporal neighbours are not contiguous with it in the conceptualised space-time matrix. It is envisioned that the performance of space-time processing could be further improved by sparsely storing pixels in each frame beside their space-time neighbours in a contiguous sequence.

Image Size		Runtimes			
Layer	Pixels	Cartesian -> Squiral	Edge Detection	Squiral -> Cartesian	Total
1	3x3	524,109fps	1,404,490fps	4,464,290fps	351,617fps
2	9x9	425,532fps	579,374fps	2,680,970fps	224,770fps
3	27x27	230,840fps	115,808fps	977,517fps	71,479fps
4	81x81	46,531fps	22,592fps	192,604fps	14,070fps
5	243x243	6,347fps	2,769fps	23,361fps	1,781fps
6	729x729	744fps	322fps	3,314fps	210fps

Table 2: Space-time Processing Runtimes

5 Conclusion

The effectiveness of a spiral coordinate system and eye tremor sampling scheme was evaluated by processing video footage at different resolutions. The results have shown that a spiral, eye tremor approach can be used to extract image features significantly faster than a traditional approach using a Cartesian coordinate system. In addition, it has been shown that a spiral, eye tremor scheme can facilitate processing in a space-time continuum and this approach could be more meaningful for motion focussed tasks. Future work will consider how multi-stage operators such as corner detectors can be applied to a spiral vector. Furthermore, we will continue to explore the characteristics of space-time processing and develop unique space-time operators.

References

[1] A. Róka, Á. Csapó, B. Reskó and P. Baranyi, "Edge Detection Model Based on Involuntary Eye Movements of the Eye Retina System," *Acta Polytechnica Hungarica,* vol. 4, no. 1, pp. 31 - 46, 2007.

[2] L. Middleton and J. Sivawamy, Hexagonal Image Processing: A Practical Approach, Springer, 2005.

[3] M. Jing, B. Scotney, S. Coleman and M. McGinnity, "A Novel Spiral Addressing Scheme for Rectangular Images," in *International Conference on Machine Vision Applications*, Tokyo, 2015.

[4] J. Ming, S. Coleman, B. Scotney and M. M., "Biologically Motivated Spiral Architecture for Fast Video Processing," in *IEEE International Conference on Image Processing*, Quebec, 2015.

[5] J. Fegan, S. Coleman, D. Kerr and B. Scotney, "Fast Corner Detection Using a Squiral Architecture," in *Irish Machine Vision and Image Processing*, Galway, 2016.

[6] J. Fegan, S. Coleman, D. Kerr and B. Scotney, "An Implementation Framework for Fast Image Processing," in *International Conference on Robotics and Vision*, Wuhan, 2017.

[7] S. Coleman, B. Scotney and B. Gardiner, "Biologically Motivated Feature Extraction," in *International Conference on Image Analysis and Processing*, 2011.

[8] T. U. o. Tokyo, "Services for High-speed Image Processing - Videos (SHIP-v)," Ishikawa Watanabe Laboratory, [Online]. Available: www.k2.t.u-tokyo.ac.jp/ship-v.

 ISBN 978-0-9934207-2-6

Study of imperfect keys to characterise the security of optical encryption

Lingfei Zhang, Thomas J. Naughton

Department of Computer Science, Maynooth University–National University of Ireland Maynooth, Maynooth, County Kildare, Ireland

Abstract

In conventional symmetric encryption, it is common for the encryption/decryption key to be reused for multiple plaintexts. This gives rise to the concept of a known-plaintext attack. In optical image encryption systems, such as double random phase encoding (DRPE), this is also the case; if one knows a plaintext-ciphertext pair, one can carry out a known-plaintext attack more efficiently than a brute-force attack, using heuristics based on phase retrieval or simulated annealing. However, we demonstrate that it is likely that an attacker will find an imperfect decryption key using such heuristics. Such an imperfect key will work for the known plaintext-ciphertext pair, but not an arbitrary unseen plaintext-ciphertext pair encrypted using the original key. In this paper, we illustrate the problem and attempt to characterise the increase in security it affords optical encryption.

Keywords: Optical information processing, Optical image processing, Optical image encryption

1 Introduction

Optical encryption has received much attention in recent years; the reason can be primarily attributed to some of its distinct advantages over conventional digital electronic hardware and software encryption. Double random phase encoding (DRPE), proposed by Réfrégier and Javidi in 1995 [Refregier and Javidi, 1995], is one of the most studied and extended technologies in optical encryption to date. A security concern about optical encryption was first reported by Carnicer et al. in 2005, where a chosen-ciphertext attack (CCA) was introduced to find the exact decryption key [Carnicer et al., 2005]. Subsequently, Peng et al. proposed a chosen-plaintext attack (CPA) to extract the exact key [Peng et al., 2006a], as well as a proposal that the original key could be obtained by solving a linear system of equations from Frauel et al. [Frauel et al., 2007]. In a more practical circumstance, if an attacker only has one plaintext-ciphertext pair, a phase-retrieval algorithm [Peng et al., 2006b] or an heuristic algorithm [Gopinathan et al., 2006] can obtain an approximation of the decryption key. A multiplicity of known pairs could be used to significantly reduce the error in the output image [Situ et al., 2007]. To respond to the above attacks, some DRPE-based security enhancement approaches have introduced additional parameters in the Fresnel domain [Situ and Zhang, 2004] and in the fractional Fourier domain [Unnikrishnan et al., 2000], and have added an extra amplitude mask directly behind the Fourier domain mask [Cheng et al., 2008] as extra keys to force attackers to find improvements from their side. However, the two phase masks are still the main concern of a number of optimized attacks [Kumar et al., 2012, Wang and Zhao, 2012, Zhang et al., 2013, Wang et al., 2015, Li and Shi, 2016].

DRPE is a symmetric encryption algorithm, which means the encryption and decryption steps share the same key. All symmetric encryption keys can only be shared over a secure channel. (It is different from asymmetric key encryption system where the public key used for encryption can be openly shared.) It is inconvenient to apply new key to each subsequent image in symmetric optical encryption because the size of the key is relatively large (routinely hundreds of times that in conventional cryptography). One possible solution is to apply modes of operation to optical encryption [Naughton et al., 2008].

 ISBN 978-0-9934207-2-6

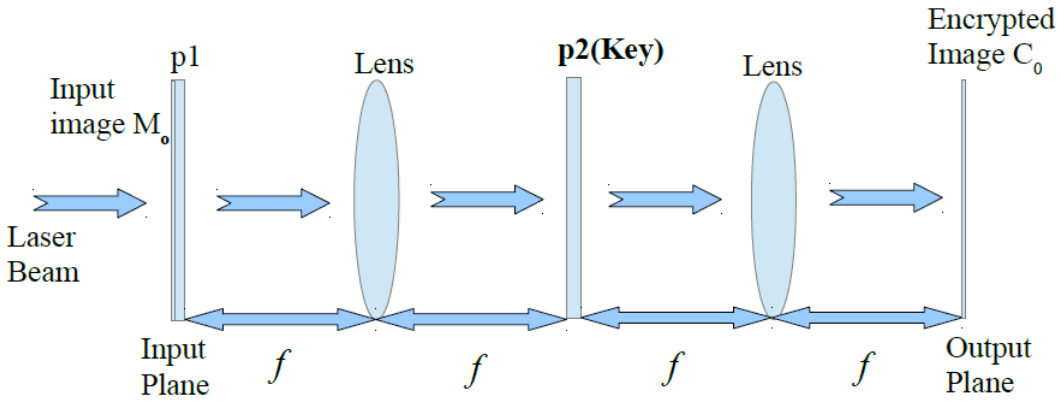


Figure 1: An illustration of symmetric DRPE, showing the location of the two phase-encoded image masks (p^1 and p^2) that constitute the encryption key (and the decryption key). Image mask p^1 is located immediately after the input image (which effects a pointwise multiplication between the input and p^1). Image mask p^2 is located in the Fourier plane, where it will be pointwise multiplied by the Fourier transform of the product of the input and p^1. A second Fourier transform returns the encrypted image from the spatial frequency domain to the space domain.

2 Keyspace analysis

The numerical implementation of the encryption process in DRPE (see illustration in Fig.1) can be described as

$$\Psi(x,y) = \mathscr{F}\left\{\mathscr{F}\left[f(x,y)p_1(x,y)\right]p_2(u,v)\right\}, \tag{1}$$

where the $p_1(x,y)$ and $p_2(u,v)$ are two statistically independent phase masks representing the encryption key of the system, and $\mathscr{F}$ is a Fourier transform. Each phase key is of the form $\exp(\mathrm{i}m(x,y))$, where m is a discrete phase mask with height M pixels and width N pixels, and with values randomly taken from the range $[0,2\pi)$. The relevant keyspace has size $K = Q^{2(M\times N)}$, where Q is the number of quantization levels in the phase distribution. A study of the keyspace has been proposed by Monaghan et al. [Monaghan et al., 2007], where a small 3×3 pixels mask with 4 quantisation levels was used. They showed that $4^{3\times3}$ possible keys in the keyspace have to be examined, in the worst case, to decrypt a known plaintext-ciphertext pair. In this discussion, they selected a tolerable decryption error threshold. Multiple keys in the keyspace were found which could decrypt the known ciphertext with an error lower than or equal to the threshold, exactly Q of which (as is well known) were equivalent to the correct key. These Q perfect keys differ from each other only by a constant additive phase $2\pi/Q$.

We determined which of the keys would decrypt subsequently unseen plaintext-ciphertext pairs encrypted using the same original key. We summarise our results using one of the grayscale and one of the binary plaintext images from our experiment. The size for each image continues the use of 3×3 pixels as chosen by Monaghan et al. [Monaghan et al., 2007] and we choose to have 8 quantisation levels in the encryption key. Normalized root mean square (NRMS) error was used to determine the quality of the decrypted output, calculated as

$$E_{\mathrm{NRMS}} = \left\{ \frac{\sum_x \sum_y |I_{\mathrm{d}}(x,y) - I(x,y)|^2}{\sum_x \sum_y |I(x,y)|^2} \right\}^{1/2}, \tag{2}$$

where I_{d} denotes the intensity of the decryption output and I is the expected intensity. A NRMS error threshold of 0.1 was chosen to decide whether decryption was successful or not. The first mask is not required in the decryption process and therefore the second mask can be regarded as the only decryption key in this system. There are $8^{3\times3} = 1.3\times10^9$ possible keys in this keyspace.

Our specific experimental platform runs on a Dell Optiplex 780 desktop PC with an Intel Core$^{\mathrm{TM}}$2 Duo E7500 CPU and 4 GB of RAM, running Python 3.5.2 in Linux. For this experiment with binary-valued plaintext

 ISBN 978-0-9934207-2-6

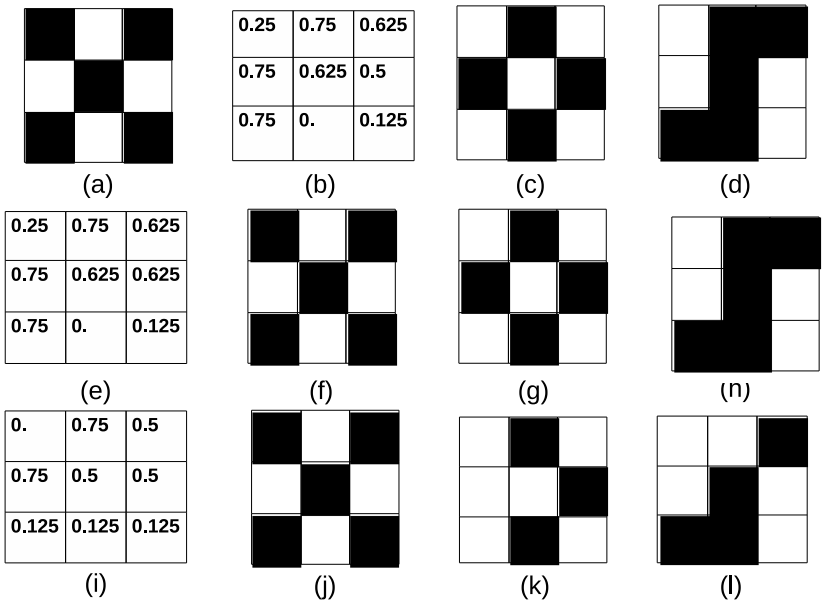


Figure 2: Binary image experiment (all subimages are explained in detail in the main text).

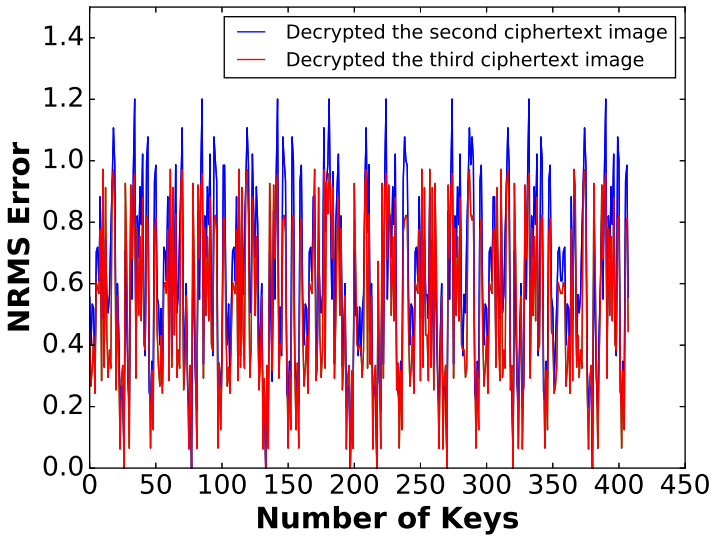


Figure 3: Binary image experiment: for all 408 keys that decrypted the known pair with NRMS error of 0.1 or less, this figure shows how well they decrypted the two unseen images.

 ISBN 978-0-9934207-2-6

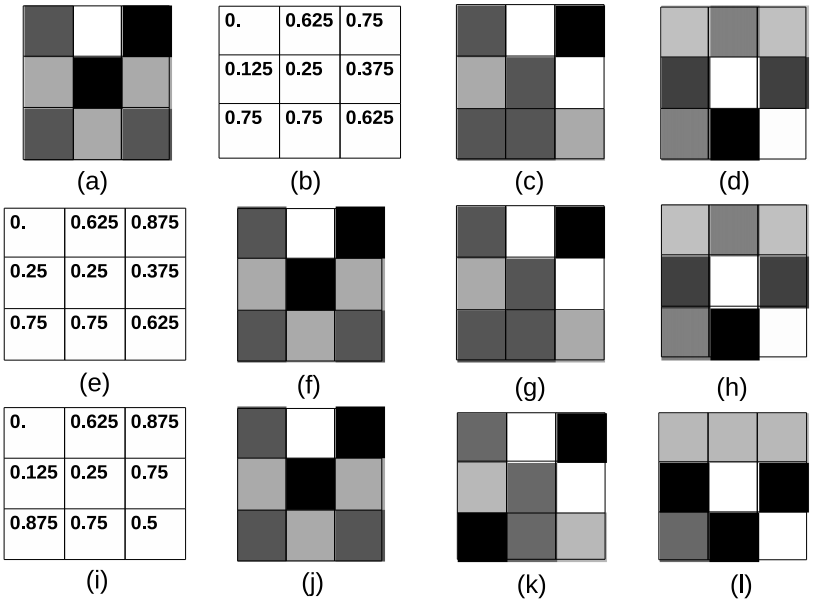


Figure 4: Greyscale image experiment (all subimages have the same meaning as those in Fig. 2).

images, it took approximately 10 hours to investigate all 1.3×10^9 possible keys. Figures 2(a) and (b) show the plaintext part of the known pair and the second random phase mask, respectively, the latter being the phase distribution before being multiplied by 2π. Figures 2(c)-(d) are the second and third binary-valued plaintext images, for which their encrypted versions only will be known to the attacker. Of the 1.3×10^9 possible keys, 408 keys decrypted the encrypted version of the known plaintext image in Fig. 2(a) with a NRMS error of 0.1 or less. Then each of these keys were used to decode encrypted versions of Figs. 2(c) and (d) that were encrypted with the same key Fig. 2(b). The corresponding NRMS errors are plotted in Fig.3, which shows that in general they yield much higher errors with unseen images than the 0.1 error yielded with the known pair. For this second stage, a NRMS error threshold of 0.2 was introduced to discriminate correct decryption (following Monaghan et al. [Monaghan et al., 2007]), followed by the application of a threshold of 0.5 to determine if the binary pixel is white (1) or black (0). From the 408 keys that decrypted the known pair with error up to 0.1, only 24 of them could correctly decrypt both unseen encrypted images, including the Q ($Q = 8$) perfect keys. Another 16 keys produced one correct decryption, with the remainder resulting in errors consistently over 0.2.

Example decryption keys and corresponding decrypted outputs are shown in Figs. 2(e)-(l). Fig. 2(e) is a decryption key with one incorrect pixel – it decrypts encrypted versions of the above three plaintext images with NRMS errors of 0.1, 0.13 and 0.17, respectively, and Figs. 2(f)-(h) are the corresponding outputs. Fig. 2(i) is a decryption key with half of the pixels incorrect – it decrypts the same images with NRMS errors of 0.1, 0.23 and 0.26, respectively, and Figs. 2(j)-(l) are the corresponding outputs. It can seen that although Fig. 2(i) decrypts the known pair with low error, it decrypts the unseen images with higher error.

The experiment was repeated for grayscale 3×3 pixel images. The results are shown in Figs. 4 and 5 and each subimage has the same explanation as those in Figs. 2 and 3. In this test, 120 keys were able to decrypt the known pair with NRMS error of 0.1 or less. Of these, only 32 could successfully decrypt both subsequent unseen images (using our choice of a threshold of 0.3 being reasonable based on visual inspection). As with the binary image case, only a minority of keys that successfully decrypt the known pair can successfully decrypt the unseen images. The key in Fig. 4(e) decrypts encrypted versions of the three plaintext images with NRMS errors of 0.1, 0.21 and 0.22, respectively. The key in Fig. 4(i) decrypts encrypted versions of the three plaintext images with NRMS errors of 0.1, 0.33 and 0.35, respectively. It can seen that although Fig. 4(i) decrypts the known pair with low error, it decrypts the unseen images with higher error.

We define "imperfect keys" as those keys that decrypt the known pair successfully, but do not consistently decrypt unseen images successfully. Of keys that decrypt the known pair, these imperfect keys are in the majority. They disrupt the job of the attacker and their presence increases the security of optical encryption

 ISBN 978-0-9934207-2-6

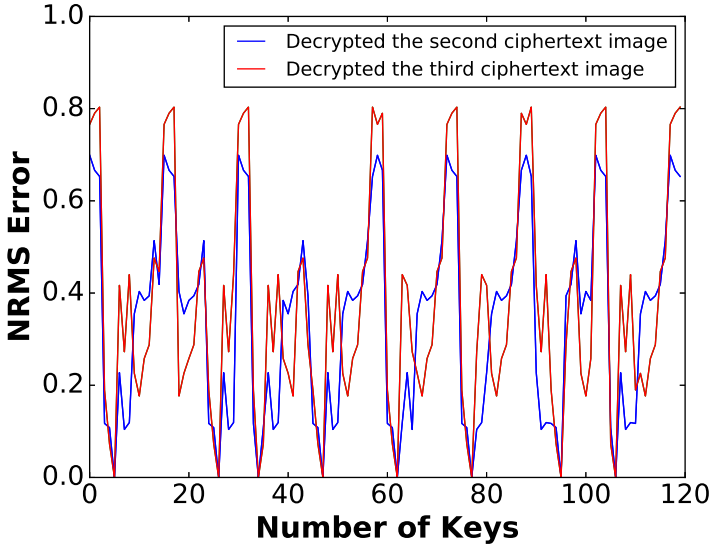


Figure 5: Greyscale image experiment: for all 120 keys that decrypted the known pair with NRMS error of 0.1 or less, this figure shows how well they decrypted the two unseen images.

because an attacker cannot know from one known plaintext pair if they have deduced an approximation of a perfect key or a (relatively useless) imperfect key.

3 Investigation of large keyspaces

The examined keyspaces in the previous section were of small sized phase keys. In practice, the plaintext image would have at least two orders of magnitude more pixels. Relatively, the cryptanalysis of the relevant keyspace becomes computationally difficult. In order to prove the existence of the imperfect key in the large size of the keyspace, such as 64×64 pixels, we intentionally selected the keys found using a simulated annealing algorithm. This heuristic approach designed to the DRPE has been proposed by Gopinathan et al. [Gopinathan et al., 2006]. The prerequisite of the SA algorithm is one known plaintext-ciphertext pair and more ciphertexts all encrypted with the same key, that is well fitted with our analysis. In that paper, the keys found in the SA algorithm based on a binary image pair have been examined to decrypt the second unseen image, the decrypting NRMS errors for 32×32 pixels and 64×64 pixels plaintexts were both close to an average of 0.4. Some examples are shown in Fig. 6(a)-(d). Moreover, we have complemented the test adopting grayscale image, the errors to decrypt the second unseen image found that sometime reached to the NRMS of 0.8, which is consistent with the worst cases shown in the Fig. 5. The threshold for the known pair remains the NRMS of 0.1.

As considered binary image is more immune to the noise than grayscale image, the peak error of 0.4 that still provides a tolerable visibility, referring to the Fig. 6(d). Referring to the binary plaintext image can not support details as many as the grayscale image, it is not an ideal choice for hiding text or graph, we do not further discuss it. On the other side, when the noise in the grayscale output exceeds NRMS of 0.8 the attacker can not recover any useful information. It is worth to mention the collision problem in DRPE that also arouse high noise in result, as shown in Fig. 6(j). Collision is commonly inevitable phenomenon in a linear system, such as DRPE. More details of the collision problem can be referred to [Situ et al., 2008]. That suggests to use grayscale image rather than binary image in DRPE to resist the known-plaintext attack.

4 Regions in the keyspace

There is no doubt that all correct but imperfect keys can be presented as adding a small amount of noise to the perfect keys. For example, the key in Fig. 2(e) and Fig. 4(e). That implies a small region around the perfect

 ISBN 978-0-9934207-2-6

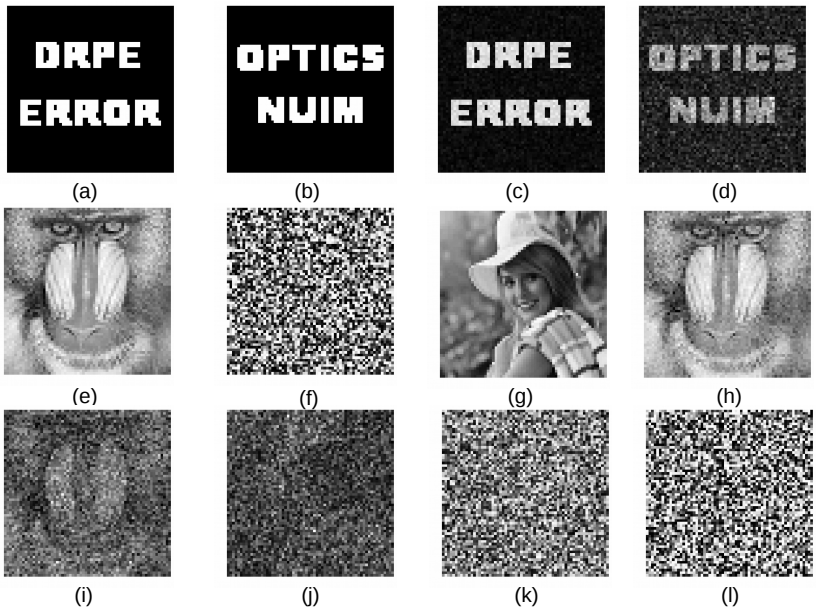


Figure 6: Original and decrypted 64×64 pixel images using the SA algorithm. (a) and (b) are the plaintext part of the known binary pair and a second pair, respectively; (c) and (d) are the decrypted versions using the SA algorithm yielding NRMS errors of 0.1 and 0.4, respectively; (e) is the input part of the known grayscale pair; (f) is the encryption key with 256 Q-levels; (g) is the second grayscale plaintext; (h) is the decrypted version of the original input image, with NRMS error of 0.09; (i) and (j) are decrypted versions of the second unseen image; (k) is the imperfect key found using the SA algorithm; (l) is a highly approximated version of the key in (f).

key in which contains the correct keys (decryption error in 0.1) in keyspace. The theory of region is previously introduced in [Situ et al., 2010]. We also believe that there is the region of imperfect keys in the keyspace.

Table.1 illustrates the increased amount of noise added to a imperfect and an approximate key to explore the region of each that guarantees an average of NRMS of 0.1, the experimental keys were found using the SA algorithm and produced by adding a slight noise to the original encryption key, shown in Fig. 6(k) and (l), respectively, the two keys decrypts the known pair yielding identically NRMS of 0.09. The known plaintext was chosen from the Fig. 6(e), and the same encryption key (f) was reused as well. In this trial, the additional noise was presented as a matrix with the equal size of the keys, and to be randomly produced based on the normal(Gaussian) distribution algorithm which simulates equivalent possibility for all pixels of the trial keys to receive a random phase error. We used one phase-level in phase distribution ($2\pi/Q$) as the unit of the adding noise, $Q = 256$ in the encryption key. The mean parameter of the normal distribution algorithm in this trial was fixed as 0, and its standard deviation(STD) was initialized as 1.0 that indicates noise added to pixels are mainly centralized at one or two phase-levels at the beginning. For the same level of STD, noise was randomly generated for 10 thousand times and each simultaneously added to both experimental keys to generate new keys. If the average error to decrypt the known pair using the new keys is under NRMS of 0.1, the STD of the algorithm increases 0.1 to perform higher amount of noise added to both keys.

Table 1: The table shows the result of additive normal (Gaussian) noise with a standard deviation (STD) of 1.4. Each column shows an average of ten thousand simulations, the mean and standard deviation of decryption errors are listed, followed by the maximum and the minimum error in the decrypted output. The last two columns show the NRMS errors of using the newly found keys to decrypt the second unseen image.

Key category	STD(noise)	Mean(NRMS)	STD(NRMS)	Max(min)	Mean(NRMS)_{2nd}	Max(min)_{2nd}
Imperfect	1.4	0.101	0.001	0.103(0.099)	0.823	0.827(0.823)
Approx	1.4	0.102	0.001	0.106(0.099)	0.094	0.101(0.094)

 ISBN 978-0-9934207-2-6

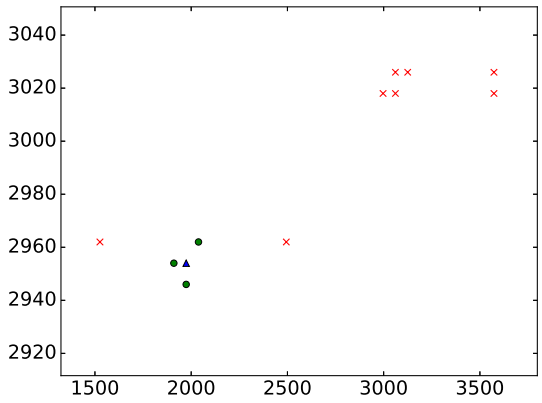


Figure 7: A portion of the keyspace (explained in the text). The total size of the keyspace is $(3 \times 3)^8$.

The two trial keys showed almost identically capability to accept noise to estimate more keys, that yields errors in a close range (see in Table. 1), the STD was both stopped at 1.4. Furthermore, the new keys derived from the imperfect key are exactly consistent style, the evidence is shown in the last column of Table.1. It is clear that how many of the correct regions in the keyspace that is equivalent to the number of perfect keys in the keyspace. We presume that the size of correct region to be larger than any of the imperfect regions in the keyspace, because the perfect key would provide zero error in decryption but no imperfect could do, that implies the perfect key would undertake higher noise to still satisfy the error threshold.

5 Classification of the keyspace

In this context, the keyspace of the DRPE can be classified as,
1. Incorrect keys, this kind of keys occupied the main part of the keyspace, which is unable to decrypt the known pair yielding NRMS errors in a preset threshold.
2. Perfect keys, they provide zero noise in the output intensity, the exact number is equivalent to the Q-levels in the encryption key.
3. Approximations of perfect keys, the high approximation of the perfect keys that decrypt the known pair with a tolerable noise, these keys can be expected to decrypt all ciphertext with consistently low errors.
4. Imperfect keys, the keys can ideally decrypt the known image pair but which unable to decrypt all ciphertexts within a reasonable error range.

Figure 7 is an illustration of a significant part of the keyspace. The exact data is originated from a previous trial (see in Fig. 4). Each of the possible keys has a unique index according to the sequence of it being examined, the corresponding coordinate is calculated and drawn on a 2d map. We use several remarks to represent different keys, such as, the triangle means the perfect key, the cross stands for the imperfect keys and the circle denotes the approximate key. Fig. 4 typically reflects the characteristic of the entire keyspace, also strongly supports our analysis of the keyspace, the perfect keys (triangle) appear always closely attached with a few approximate keys (circle), the evidence of the correct regions. Theoretically, there is only one correct region for each of the perfect keys. Besides, we notice that one imperfect key separately located at the right and left side of the correct region, that mean the flexible choice of different error threshold would inappropriately determine some keys actually close to the perfect key. Meanwhile, the imperfect keys in Fig. 7 display as clusters or lines, that is regarded as the imperfect region. Fig. 7 can be mapped into the keyspace by continuously adding a constant phase of $1/Q$.

6 Conclusion

In optical encryption, many previous studies have considered a known-plaintext attack. A commonality among these studies has been them seeking a highly approximated decryption key using an efficient heuristic algorithm

 ISBN 978-0-9934207-2-6

rather than an exhaustive attack. In this paper, we show that these attacks are not robust, and that while the key found will decrypt the known encryption/decryption pair, it is not likely to decrypt unseen images encrypted with the same key. This implies that optical encryption may not be as susceptable to plaintext attacks as previously reported.

Acknowledgements. This publication has emanated from research conducted with the financial support of an Irish Research Council (IRC) Postgraduate Scholarship and of Science Foundation Ireland (SFI) under grant no. 13/CDA/2224.

References

[Carnicer et al., 2005] Carnicer, A., Montes-Usategui, M., Arcos, S., and Juvells, I. (2005). Vulnerability to chosen-cyphertext attacks of optical encryption schemes based on double random phase keys. *Opt. Lett.*, 30(13):1644–1646.

[Cheng et al., 2008] Cheng, X. C., Cai, L. Z., Wang, Y. R., Meng, X. F., Zhang, H., Xu, X. F., Shen, X. X., and Dong, G. Y. (2008). Security enhancement of double-random phase encryption by amplitude modulation. *Opt. Lett.*, 33(14):1575–1577.

[Frauel et al., 2007] Frauel, Y., Castro, A., Naughton, T. J., and Javidi, B. (2007). Resistance of the double random phase encryption against various attacks. *Opt. Express*, 15(16):10253–10265.

[Gopinathan et al., 2006] Gopinathan, U., Monaghan, D. S., Naughton, T. J., and Sheridan, J. T. (2006). A known-plaintext heuristic attack on the fourier plane encryption algorithm. *Opt. Express*, 14(8):3181–3186.

[Kumar et al., 2012] Kumar, P., Kumar, A., Joseph, J., and Singh, K. (2012). Vulnerability of the security enhanced double random phase-amplitude encryption scheme to point spread function attack. *Optics and Lasers in Engineering*, 50(9):1196 – 1201.

[Li and Shi, 2016] Li, T. and Shi, Y. (2016). Vulnerability of impulse attack-free four random phase mask cryptosystems to chosen-plaintext attack. *Journal of Optics*, 18(3):035702.

[Monaghan et al., 2007] Monaghan, D. S., Gopinathan, U., Naughton, T. J., and Sheridan, J. T. (2007). Key-space analysis of double random phase encryption technique. *Appl. Opt.*, 46(26):6641–6647.

[Naughton et al., 2008] Naughton, T. J., Hennelly, B. M., and Dowling, T. (2008). Introducing secure modes of operation for optical encryption. *J. Opt. Soc. Am. A*, 25(10):2608–2617.

[Peng et al., 2006a] Peng, X., Wei, H., and Zhang, P. (2006a). Chosen-plaintext attack on lensless double-random phase encoding in the fresnel domain. *Opt. Lett.*, 31(22):3261–3263.

[Peng et al., 2006b] Peng, X., Zhang, P., Wei, H., and Yu, B. (2006b). Known-plaintext attack on optical encryption based on double random phase keys. *Opt. Lett.*, 31(8):1044–1046.

[Refregier and Javidi, 1995] Refregier, P. and Javidi, B. (1995). Optical image encryption based on input plane and fourier planerandom encoding. *Opt. Lett.*, 20(7):767–769.

[Situ et al., 2007] Situ, G., Gopinathan, U., Monaghan, D. S., and Sheridan, J. T. (2007). Cryptanalysis of optical security systems with significant output images. *Appl. Opt.*, 46(22):5257–5262.

[Situ et al., 2008] Situ, G., Monaghan, D. S., Naughton, T. J., Sheridan, J. T., Pedrini, G., and Osten, W. (2008). Collision in double random phase encoding. *Optics Communications*, 281(20):5122 – 5125.

[Situ et al., 2010] Situ, G., Pedrini, G., and Osten, W. (2010). Strategy for cryptanalysis of optical encryption in the fresnel domain. *Appl. Opt.*, 49(3):457–462.

[Situ and Zhang, 2004] Situ, G. and Zhang, J. (2004). Double random-phase encoding in the fresnel domain. *Opt. Lett.*, 29(14):1584–1586.

[Unnikrishnan et al., 2000] Unnikrishnan, G., Joseph, J., and Singh, K. (2000). Optical encryption by double-random phase encoding in the fractional fourier domain. *Opt. Lett.*, 25(12):887–889.

[Wang and Zhao, 2012] Wang, X. and Zhao, D. (2012). A special attack on the asymmetric cryptosystem based on phase-truncated fourier transforms. *Optics Communications*, 285(6):1078 – 1081.

[Wang et al., 2015] Wang, Y., Quan, C., and Tay, C. J. (2015). Improved method of attack on an asymmetric cryptosystem based on phase-truncated fourier transform. *Appl. Opt.*, 54(22):6874–6881.

[Zhang et al., 2013] Zhang, Y., Xiao, D., Wen, W., and Liu, H. (2013). Vulnerability to chosen-plaintext attack of a general optical encryption model with the architecture of scrambling-then-double random phase encoding. *Opt. Lett.*, 38(21):4506–4509.

 ISBN 978-0-9934207-2-6

Gaussian Random Vector Fields in Trajectory Modelling

Miguel Barão*, Jorge S. Marques**

** University of Évora and INESC-ID Lisboa, Portugal*
*** Institute for Systems and Robotics (ISR/IST), LARSyS,*
Instituto Superior Técnico, University of Lisboa, Portugal

Abstract

This paper proposes the use of Gaussian random vector fields as a generative model to describe a set of observed trajectories in a 2-dimensional space. The observed trajectories are sequences of points in space sampled from continuous trajectories that are assumed to have been generated by an underlying velocity field. Given the observed velocities connecting the trajectory points, a vector field is obtained by conditioning a Gaussian random vector field. Some results obtained in simulation are presented.

Keywords: Random fields, trajectory modelling, Pedestrian surveillance.

1 Introduction

This paper deals with the estimation of a 2-dimensional vector field describing a set of observed trajectories. One of the possible applications is to estimate models for moving people, cars, animals, etc. The models can then be used in surveillance problems to detect abnormal behavior when new observations (trajectories) do not fit well into the previously estimated models, considered as "normal".

This kind of problems has been tackled before using a: 1) a parametric approach where a model with a small number of parameters, e.g. linear dynamical system, is fit to the data; 2) a nonparametric approach where a grid with a large number of nodes is defined and vectors estimated at those nodes, then the vector field is obtaind by interpolation of those nodes [Nascimento et al., 2014, Nascimento et al., 2015, Ferreira et al., 2013]; 3) Using gaussian process regression flow [Kim et al., 2011].

Here we propose the use of random vector fields to model and estimate the underlying vector field generating the observed trajectory data. The use of the random vector fields provides some advantages over the nonparametric approach. The random vector field approach replaces the interpolation by conditioning the random field by the available data. The random vector field works as a prior and by working uniquely under a probabilistic setting, all uncertainties are taken into account automatically which is not the case when using interpolation.

The main contributions are the random vector field proposal and the issues related with the computational complexity of the algorithm, particularly the replacement of the data by a fixed size statistic that may allow online application of the framework.

The paper is organized as follows: section 2 provides some background on random fields, section 3 formulates the problem, 3.1 provides a simulation example, section 3.2 proposes ways to deal with complexity and finally section 4 draws conclusions.

2 Background

A *random field* is a generalization of a stochastic process where the 1-dimensional "time" parameter is replaced by n-dimensional space. In the most general setup a random field is defined as a measurable function

$$T : \mathscr{M} \times \Omega \to \mathscr{N} \tag{1}$$

 ISBN 978-0-9934207-2-6

where $\mathscr{M}$ and $\mathscr{N}$ are manifolds and Ω is a realization space. In this work, we will be dealing with vector valued random fields in a 2-dimensional image space $T : \mathbb{R}^2 \times \Omega \to \mathbb{R}^2$. In this case, for every realization $\omega \in \Omega$ of the random vector field we get a real vector field $T^\omega : \mathbb{R}^2 \to \mathbb{R}^2$ where $T^\omega \triangleq T(\cdot, \omega)$. Similarly, for every point $x \in \mathbb{R}^2$ in the image space we get a random vector $T_x : \Omega \to \mathbb{R}^2$ where $T_x \triangleq T(x, \cdot)$.

A particular example of a random vector field is one where a Gaussian assumption is made. In this case, the random vector field is completely specified by its mean and covariance functions $\mu(\cdot)$ and $K(\cdot,\cdot)$. The mean function $\mu : \mathbb{R}^2 \to \mathbb{R}^2$ assigns a 2D vector to each point in the 2D image space, while the covariance function (kernel) is a function $K : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}^{2\times 2}$ such that, given two points in the image space, returns the covariance matrix that relates the two random vectors at those two points. For example, for any pair of points $x_1, x_2 \in \mathbb{R}^2$, the random vectors $T(x_1)$ and $T(x_2)$ are jointly characterized by a multivariable Gaussian distribution

$$\begin{bmatrix} T(x_1) \\ T(x_2) \end{bmatrix} \sim \mathscr{N}\left(\begin{bmatrix} \mu(x_1) \\ \mu(x_2) \end{bmatrix}, \begin{bmatrix} K(x_1,x_1) & K(x_1,x_2) \\ K(x_2,x_1) & K(x_2,x_2) \end{bmatrix} \right), \tag{2}$$

where the mean vector and covariance matrix have dimensions 4×1 and 4×4 respectively.

The previous example generalizes to any finite number of points $(x_1, \ldots, x_n)$. The multivariable Gaussian distributions obtained this way can be though of as the marginal distributions from an underlying Gaussian random vector field, provided the conditions of the Kolmogorov extension theorem are satisfied [Billingsley, 1995].

Considering again the joint distribution (2), if the vector $T(x_1) = V$ is observed then the conditional distribution $p(T(x_2) \mid T(x_1) = V)$ characterizes the prediction $T(x_2)$ of the field at the point x_2, which is again Gaussian distributed $\mathscr{N}(\mu^*, K^*)$ with mean and covariance given by

$$\mu^* = \mu_2 + K_{21} K_{11}^{-1} (V - \mu_1), \tag{3}$$

$$K^* = K_{22} - K_{21} K_{11}^{-1} K_{12}, \tag{4}$$

where $\mu_i \triangleq \mu(x_i)$ and $K_{ij} \triangleq K(x_i, x_j)$. Again, this generalizes to any finite number of points partitioned into two sets containing observed and unknown vectors. The prediction of the unknown vectors can be performed using the same equations (3)-(4), where the subscript 1 refers to the observed data and the subscript 2 refers to the predictive part of the mean and covariance matrix.

For a more in depth introduction to gaussian processes refer to [Rasmussen and Williams, 2006].

3 Problem Formulation

In this paper, a set of observed trajectories is used to estimate a generative model that best fits the data. The observed trajectories are represented by sequences of points in a 2-dimensional Euclidean space sampled at regular time intervals.

It is assumed that the trajectories $\{x_t\}$ were generated by flowing along an unknown vector field $T(x)$. The additive variable w_t represents unknown additive perturbations affecting the velocity. Using a normalized time interval $\Delta t = 1$ between samples, gives the generative model

$$x_t = x_{t-1} + T(x_{t-1}) + w_t. \tag{5}$$

Given an observed trajectory $(x_0, x_1, \ldots, x_L)$, the computed velocities are calculated by the difference $v_t = x_{t+1} - x_t$, yielding a set of L position/velocity pairs $\{(x_0, v_0), \ldots, (x_{L-1}, v_{L-1})\}$. In what follows, the collection of positions and velocities are represented in matrix form as $\mathbf{X}$ and $\mathbf{T}$ of size $L \times 2$, where each row represents a particular time instant and the two columns represent the horizontal and vertical axis of the 2-dimensional image space.

To predict the velocities at arbitrary points using the random vector field technique, the desired coordinates $\mathbf{G}$ are appended to the trajectory points $\mathbf{X}$, and the velocities $\mathbf{T}^*$ to be predicted are appended to $\mathbf{T}$ to get the augmented matrices

$$\begin{bmatrix} \mathbf{X} \\ \mathbf{G} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \mathbf{T} \\ \mathbf{T}^* \end{bmatrix}. \tag{6}$$

 ISBN 978-0-9934207-2-6

Then, given a joint probability distribution $p(\mathbf{T}, \mathbf{T}^*)$, the predicted velocities are then obtained by taking the conditional distribution $p(\mathbf{T}^* \mid \mathbf{T})$, as described in section 2. The following assumptions are made to the joint distribution $p(\mathbf{T}, \mathbf{T}^*)$:

1. The random vector field has zero mean everywhere, $\mu(x) = \begin{bmatrix} 0 & 0 \end{bmatrix}$.
2. Given any two points x_i and x_j, the covariance matrix between their respective velocity vectors is isotropic in $\mathbb{R}^2$ and therefore the covariance matrix is given by $k_{ij}\mathbf{I}_{2\times 2}$, where $k_{ij} \triangleq k(x_i, x_j)$ is a scalar function that depends only on the chosen points.

   This allows covariances to be greatly simplified by using the reduced covariance matrix

$$\mathbf{K} = \begin{bmatrix} k_{11} & \cdots & k_{1n} \\ \vdots & & \vdots \\ k_{n1} & \cdots & k_{nn} \end{bmatrix} \tag{7}$$

   instead of the full matrix, which is given by the Kronecker product $\mathbf{K} \otimes \mathbf{I}_{2\times 2}$.
3. The kernel function $k(\cdot, \cdot)$ used to define the covariance is a positive decreasing function depending on the distance between the two points.

The previous three assumptions impose a stationarity condition in space. Examples of such functions are the Ornstein-Uhlenbeck, squared exponential and triangular functions

$$k_{ou}(x_1, x_2) \triangleq \exp(-\alpha \| x_1 - x_2 \|), \tag{8}$$

$$k_{se}(x_1, x_2) \triangleq \exp(-\alpha \| x_1 - x_2 \|^2), \tag{9}$$

$$k_{tri}(x_1, x_2) \triangleq \max(1 - \alpha \| x_1 - x_2 \|, 0), \tag{10}$$

with parameter α adjusting the spatial dependency between the points.

Given the observed trajectories $\mathbf{X}$ and the grid coordinates $\mathbf{G}$, the covariance matrix is computed in partitioned form as

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{xx} & \mathbf{K}_{xg} \\ \mathbf{K}_{gx} & \mathbf{K}_{gg} \end{bmatrix} \tag{11}$$

where the subscripts x and g denote respectively the part of the observed data and the points of the grid where prediction is to take place.

The velocity vectors can now be predicted using (3)-(4):

$$\mu^* = \mathbf{K}_{gx}\mathbf{K}_{xx}^{-1}\mathbf{V} \tag{12}$$

$$\mathbf{K}_{gg}^* = \mathbf{K}_{gg} - \mathbf{K}_{gx}\mathbf{K}_{xx}^{-1}\mathbf{K}_{xg} \tag{13}$$

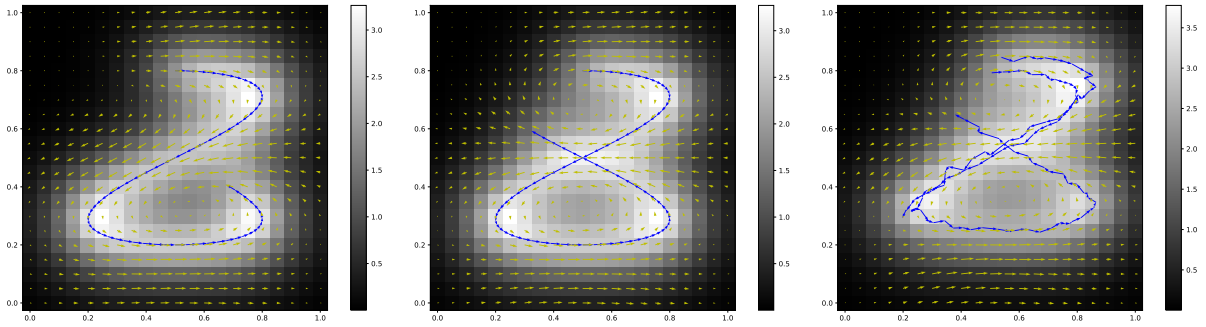
where the zero mean was dropped from the equations.

3.1 Example

To illustrate the algorithm a trajectory was generated and the prediction was performed on a regularly spaced 21×21 grid using a squared-exponential kernel. Figure 1 shows the observed trajectory in blue. The predicted velocities at the grid are jointly gaussian with mean μ^* and covariance matrix $\mathbf{K}_{gg}^*$. The figure shows the marginals distributions for individual points, with the mean represented by green arrows and the variances obtained from the diagonal of $\mathbf{K}_{gg}^*$ represented by the background gray level in log-scale, darker meaning higher variance/uncertainty in the prediction.

It can be seen that the nodes near the trajectory have much lower uncertainty in the predicted velocity than nodes in areas where no nearby data exists. The darker areas tend to give a result close to the prior, assigning near zero velocity.

 ISBN 978-0-9934207-2-6



(a) Single trajectory of length 70. (b) Single trajectory of length 100. (c) Two noisy trajectories with different lengths.

Figure 1: Observed trajectories (blue) and random field prediction at a regularly space grid (yellow). The gray background represents the uncertainty associated with the prediction.

Although a crossing exists in the trajectory, which is impossible in a deterministic dynamical system, the solution found can be interpreted as two separate upper and lower regions with circular motion. The crossing is then explained by the stochastic nature of the problem where a perturbation can produce the jump from one region to the other.

3.2 Dealing with complexity

A practical problem of directly applying the equations (12)-(13) is dealing with variable size and always increasing amount of data. A suboptimal solution to this problem is to use the predicted vectors at the grid as a fixed size statistic that describes the past observed data. As new observations are obtained the statistic can then be updated and the data discarded.

To implement a fixed size statistic, a fixed size grid is used. The algorithm now works in two steps: in the first step the grid vectors are estimated and in the second step the vector field is predicted from the grid, which now acts as a new "virtual" data, instead of the original trajectory.

As a further reduction in complexity, the grid nodes with high uncertainty can be omited and the prediction can be performed using only smaller but relevant information. Figure 2 shows three fields generated from subsets of nodes from solution in figure 1.

This solution is clearly suboptimal since information is being retained in areas where an already good description exists and new contributions are small. A possibly better approach would be to keep the nodes that lead to the largest information gain. This line of research is still ongoing.

4 Conclusions

This paper deals with the use Gaussian vector random fields to build models describing a set of trajectories observed in 2-dimensional space. The use of the random vector field framework has the main advantage that all the uncertainties are being taken into account. The random vector field can be seen as providing a prior, that when constrained on the data provides prediction for the rest of the space. Regions where no data is observed are then closer to the prior and have higher associated uncertainty.

Constraining on all available data has the drawback that, in an online setting, the complexity is always increasing. Here we propose to replace the data by a fixed size statistic that is then updated online. The size of this statistic is experimented with by selecting only the nodes with lower uncertainty. While this is not an optimal solution, it provides a first step in the pursuit of a sparse solution that keeps the complexity of the algorithm sufficiently low to be able to run it online.

 ISBN 978-0-9934207-2-6

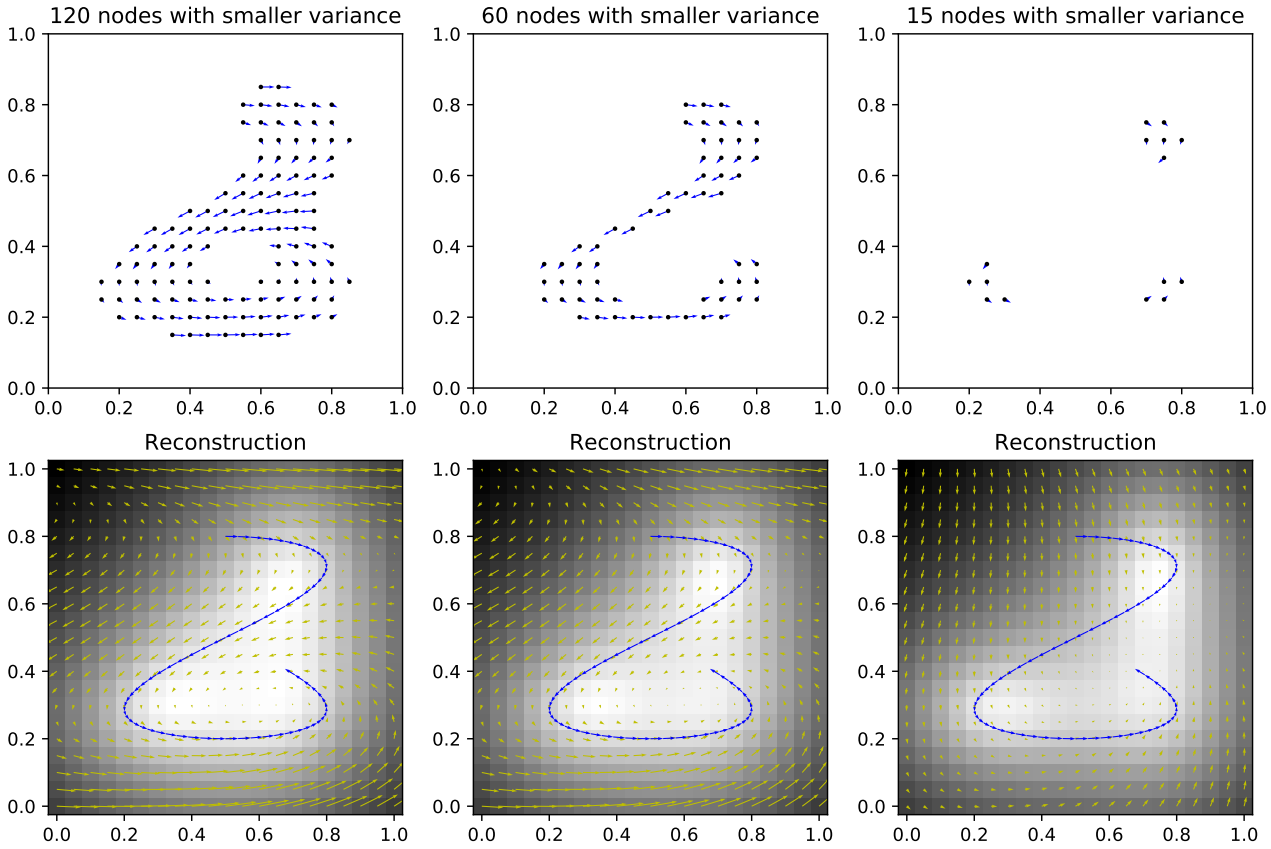


Figure 2: Observed trajectory (blue) and random field prediction at a regularly space grid (yellow). The gray background represents the uncertainty associated with the prediction.

 ISBN 978-0-9934207-2-6

Acknowledgments

This work was partially supported by FCT (Fundação para a Ciência e Tecnologia), grants PTDC/EEIPRO/0426/2014, UID/CEC/50021/2013 and UID/EEA/50009/2013.

References

[Billingsley, 1995] Billingsley, P. (1995). *Probability and Measure*. John Wiley and Sons.

[Ferreira et al., 2013] Ferreira, N., Klosowski, J. T., Scheidegger, C. E., and Silva, C. T. (2013). Vector fields k-means: Clustering trajectories by fitting multiple vector fields. In B. Preim, P. R. and Theisel, H., editors, *Eurographics Conference on Visualization*, volume 32.

[Kim et al., 2011] Kim, K., Lee, D., and Essa, I. (2011). Gaussian process regression flow for analysis of motion trajectories. In *IEEE International Conference on Computer Vision*.

[Nascimento et al., 2014] Nascimento, J. C., Barão, M., Marques, J. S., and Lemos, J. M. (2014). Information geometric algorithm for estimating switching probabilities in space-varying HMM. *IEEE Transactions on Image Processing*, 23(12):5263–5273.

[Nascimento et al., 2015] Nascimento, J. C., Barão, M., Marques, J. S., and Lemos, J. M. (2015). An information geometric framework for the optimization on discrete probability spaces: Application to human trajectory classification. *Neurocomputing*, 150:155–162.

[Rasmussen and Williams, 2006] Rasmussen, C. E. and Williams, C. (2006). *Gaussian Processes for Machine Learning*. MIT Press.

Automatic Book Finding on Bookshelves.

Jason Hogan & Kenneth Dawson-Howe

School of Computer Science and Statistics, Trinity College, University of Dublin

Abstract

A novel application of Computer Vision is described for locating books on shelves based on images of their spines. Using simple template matching and colour comparison techniques the system can reliably locate books. The system was built into a mobile phone based app and validated in different settings (bookstore, university library and home bookshelf).

Keywords: Computer Vision Application, Recognition

1 Introduction & State of the Art

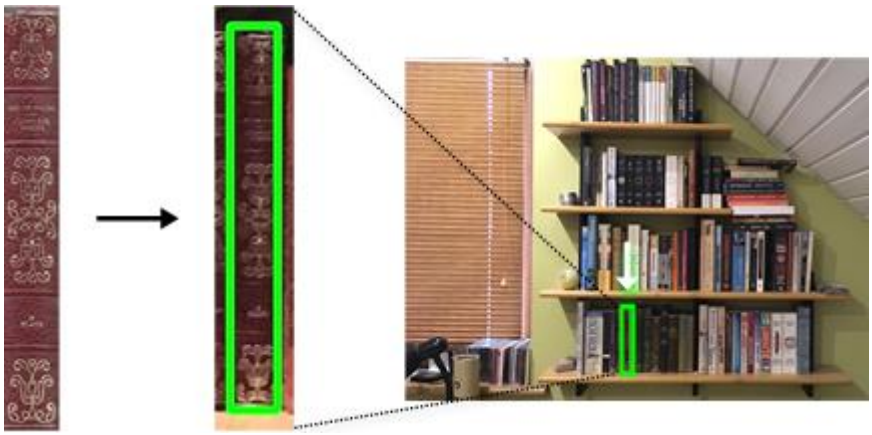


Figure 1. Book recognition based on an image of the spine of the book. Scores were 63.7% for template matching and 37.9% for colour comparison.

This paper describes a novel but simple application of computer vision working with collections of physical books (See Figure 1). While packaged as an application to locate books on bookshelves it could equally be used as part of a cataloguing system.

Existing systems in this field are very limited. [Chen et al., 2010] present an application that uses SURF feature detection to analyse images of bookshelves against a database of book spines and identifies each book in the given image. This method starts by extracting individual book spines from an image, then works backwards to find which spine from the database each book matches most closely. Once it has segmented out a single book spine, it creates a shortlist of the 50 most similar spines from the database using SURF (Speeded Up Robust Features) feature detection, then uses RANSAC (Random Sample Consensus) to narrow down this list to find the best match. This method relies on extracting comprehensive edge data from the image, and in the test images provided in the paper every book spine is a different colour, there are no shadows, and the image is taken at close range, all of which results in strong edges between books.

[Evernote] has a built-in functionality that automatically performs optical character recognition on any images saved in notes in the application, and catalogues any data that it can find, making the images searchable. It has been demonstrated that this feature could be used to catalogue shelves of books. This relies on a clear, high resolution

 ISBN 978-0-9934207-2-6

photo of the shelf taken at close range.
The application presented here works from an image taken on a mobile device (such as a phone) and is intended to work whether or not the text (if any) on the spine of the book is visible. It is ultimately intended to be part of a larger application which directs users to the correct bookshelf and then book.

2 Locating Shelves and Books

The location of individual books in a particular image is done in two stages. First the shelves are located and then each shelf is searched in parallel for the relevant book. There is no attempt to segment individual books prior to locating them as this was found to be impractical (as often there are no/little edge features between the books).

2.1 Locating Shelves

To locate the shelves in an image we compute an edge image and look for rows where there are few (or no) vertical edge pixels (as computed using the vertical Sobel partial derivative). See Figure 2. We are assuming that the image is taken so that the shelves are horizontal, but this requirement could easily be relaxed.

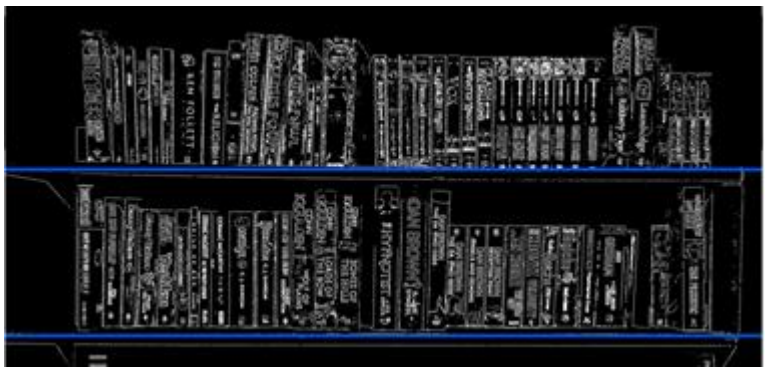


Figure 2. Locating shelves based on searching for rows with few or no vertical edge pixels.

2.2 Recognising Books

Having located shelves of potential books we search for the requested books using a previously acquired image of the book spine, using a combination of CLAHE histogram equalisation, standard template matching and colour matching.

2.2.1 Preprocessing - Histogram Equalisation

Contrast Limited Adaptive Histogram Equalisation (CLAHE) [Sasi and Jayasree, 2013] is used as a preprocessing step to lessen any shadow effects on the greyscale image of each bookshelf. See Figure 3.

Figure 3. Reduction of shadow effects using CLAHE. Original greyscale image (left) and the same image after CLAHE (right). The highlighted areas are significant shadows caused by the shelf above.

 ISBN 978-0-9934207-2-6

2.2.2 Template Matching

Template matching is used to compare a greyscale version of the book spine image with each bookshelf in each possible position at a number of scales (starting with the height of the shelf and reducing down by 5% until the height is half the height of the shelf). Normalised correlation coefficient is used as the matching metric, and the highest confidence match is considered further.

2.2.3 Colour Comparison

As an additional verification stage we compare a colour histograms in RGB space with 8 bits per channel using a colour correlation metric $D_{Correlation}(h_1, h_2)$ (1). This gives a score between -1.0 and +1.0 with +1.0 being a perfect match. CLAHE equalisation is applied to the luminance channel before the comparison.

$$D_{Correlation}(h_1, h_2) = \frac{\sum_i (h_1(i) - \overline{h_1})(h_2(i) - \overline{h_2})}{\sqrt{\sum_i (h_1(i) - \overline{h_1})^2 \sum_i (h_2(i) - \overline{h_2})^2}} \quad (1)$$

3 Testing & Results

There was no available database of bookshelf images, and of book spine images. As a result, we scanned 106 book spines and searched for each of these books in an image of one of three bookcases. The ground truth (success or failure) of each search was done manually based on whether the book was successfully located or not by the application in the search. Each book was only searched for once in an image of a bookshelf which did contain the book in question.

3.1 Results

We plotted the template matching correlation coefficient scores against the colour comparison correlation scores (See Figure 4) for all 106 test cases. We found that we could create a liner discrimination function which separated all but one of the correct matches from the incorrect matches.

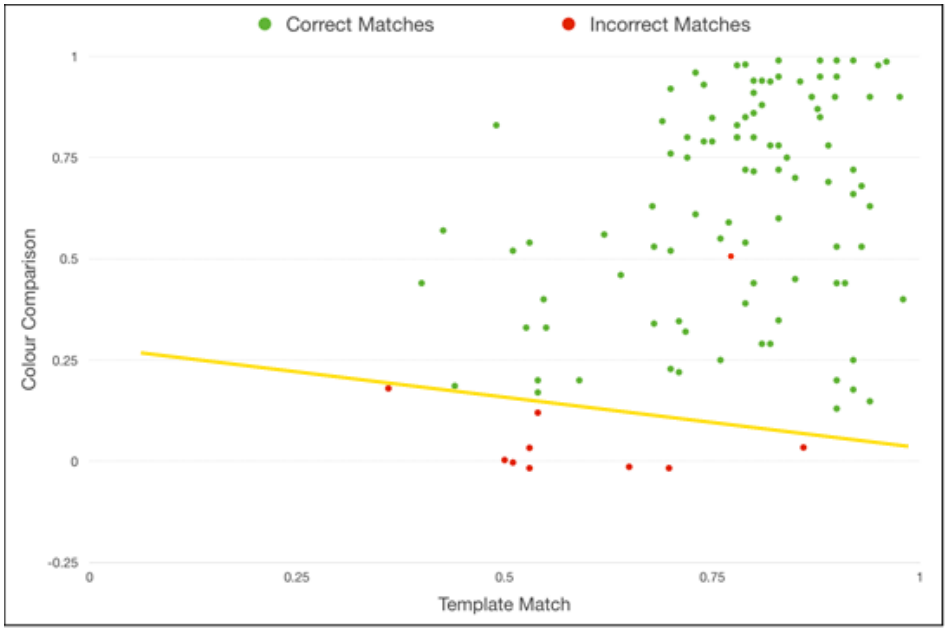


Figure 4. Graph showing the scores for all 106 tests with a discrimination function correctly separating all but one test. Books were located successfully in 90.5% of the tests, were not found (incorrectly) in 8.6% of the tests and were located incorrectly in 0.9% of the tests.

 ISBN 978-0-9934207-2-6

Sample successful results are shown in Figures 1 & 5, while sample incorrect results are shown in Figure 6.

Figure 5. Sample images from the application showing books located in a bookstore (left; Hodges Figgis in Dublin; taken with permission) and in a university library (right). Note that the successful location of the book in the library may have been influenced by the sticker on the spine indicating the location of the book.

Figure 6. Incorrect matches. The incorrect match on the left (template matching 77.1%, colour comparison 50.7%) produced is the one matching error which was not removed by the classification function. The match on the right was missed due to the damage to the spine of the book.

4 Conclusions

This paper represents a proof of concept for an application to locate books on bookshelves without needing to have sufficient resolution to read the titles. The potential for incorporating this into a larger system for use in bookshops/libraries is significant.

References

[Chen et al., 2010] Chen, David M. et al. *Building Book Inventories using Smartphones*, Proceedings of the 18th ACM International Conference on Multimedia, 2010.

[Sasi and Jayasree, 2013] Neethu M. Sasi, and V. K. Jayasree, *Contrast Limited Adaptive Histogram Equalization for Qualitative Enhancement of Myocardial Perfusion Images*, Engineering, 2013, 5, 326-331

[Evernote] Evernote Note Taking Application available at https://evernote.com

Digital holographic sensor network and image analyses for distributed potable water monitoring

Tomi Pitkäaho,[1,2] Ville Pitkäkangas,[2], Mikko Niemelä[2], Sudheesh K. Rajput[3,4], Naveen K. Nishchal[3] and Thomas J. Naughton[1]

[1] *Department of Computer Science, Maynooth University–National University of Ireland Maynooth, Maynooth, County Kildare, Ireland*

[2] *University of Oulu, Oulu Southern Institute, Pajatie 5, 85500 Nivala, Finland*

[3] *Department of Physics, Indian Institute of Technology Patna Bihta, Patna-801 103, India*

[4] *Department of Systems Science Graduate School of System Informatics Kobe University, Rokkodai 1-1 Nada, Kobe 657-8501, Japan*

Abstract

Water-related diseases affect societies in all parts of the world. On-line sensors are considered as a solution to the problems of low sampling density and time-consuming culturing methods associated with laboratory testing for microbiological content in potable water. Digital holographic microscopy (DHM) has been shown to be well suited to image microscopic objects, especially in laboratory environments, and has the potential to rival state-of-the-art techniques such as advanced turbidity measurement. In this paper, we provide a solution that permits DHM to be applied to a whole class of on-line remote sensor networks, of which potable water analysis is one example.

Keywords: digital holographic microscopy, water quality, compression

1 Introduction

Water-related diseases (WRDs), such as diarrhea, typhoid fever, and hepatitis A, remain one class of major global health concerns [World Health Organization and others, 2010]. Nearly ninety percent of diarrheal diseases are caused by bad quality drinking and bathing water [World Health Organization and others, 2004]. To increase safety and to ensure high microbiological quality of potable water, the use of on-line sensors has been suggested [Lopez-Roldan et al., 2013]. Digital holographic microscopy (DHM) is an imaging technique that is well suited for imaging three-dimensional (3D) objects [Javidi et al., 2005, Garcia-Sucerquia et al., 2006, Mudanyali et al., 2010]. Digital holography can be regarded as an enhancement of light scattering approaches [Wyatt, 1968] with the following desirable properties: (i) the scattering from the object is captured holographically so that the scattering can be reversed in software thus generating an in-focus image of the object at any distance from the camera, (ii) a relatively large volume can be imaged so that the object does not have to be in any special location, and (iii) multiple objects can be sensed and distinguished simultaneously.

2 Design choices

We identify four major design choices for an on-line DHM sensor: I) optical hardware and architecture, II) location of data processing and analyses, III) processing and analysis algorithms, and IV) hologram video compression.

 ISBN 978-0-9934207-2-6

2.1 Optical hardware and architecture

The trade-offs between various interferometer architectures and illumination choices have been well-studied. For example, a free-space propagation DHM avoids the need for an expensive microscope objective, but suffers from a depth-dependent spatial resolution, and vibration-sensitive alignment of a pinhole, to produce the spherical wave.

2.2 Location of data processing and analyses

Due to the large volume of data in holographic video of real-world objects, networked holographic video applications have an ever-present problem of how to optimally partition the data processing between the capture side (before network transmission) and the display side [Kujawinska et al., 2014].

2.3 Processing and analyses algorithms

As the system is required to be near-real-time, algorithms need to be optimized and chosen on the basis of the specific application. In the literature, objects have been found in hologram reconstruction volumes using amplitude analysis [Restrepo and Garcia-Sucerquia, 2012], edge detection [Kempkes et al., 2009] and contrast analysis [Pitkäaho et al., 2014]. However, a different set of methods is appropriate for each application.

2.4 Hologram video compression

Hologram video compression is necessary because in practice the limiting factor on the sampling density of the system is the data throughput over the network. The principle employed in this compression strategy is to partition (temporally and spatially) the regions of pixels in the hologram video sequence according to how much information they contain about the sensed particles, and represent those regions with a number of bits per pixel proportional to how much information they contain. We include pixels with varying numbers of bits of representation (including the possibility of zero bits. Starting from the second hologram in the video sequence, and for each hologram, we apply the steps as shown in Fig. 1. Holograms are subtracted from their predecessor to generate a subtraction hologram.

3 Results

To verify the effectiveness of the design, a physical implementation using inexpensive off-the-shelf components was designed, built, and evaluated in an active potable water facility. The imaging sensor was an in-line DHM, illustrated in Fig. 2, whose principal components were a 405 nm laser module (CNI PGL-D8-405-50), a flow-through channel (Ibidi, 81121 μ-Slide 0.1 Luer), a 40X microscope objective (Olympus PLN 40X), and a 1280×1024 pixel CMOS camera with a 5.3 μm pixel pitch (IDS Imaging UI-1242LE-M). The sensor was evaluated in a laboratory environment with test objects such as a static resolution chart, 1 μm latex beads and living *E. coli* in water flow.

For tests in an active potable water facility, a portable version of the sensor was assembled in a commercially available aluminum case that contained a low-calculating-power computer unit (Thinclient Zotac Zbox), the imaging and sample circulation components as described above, and a 3G modem (Huawei E367). The flow speed was controlled with a variable-area flow meter (Kytola instruments LH-).

The Finnish wholesale potable water company Vesikolmio Oy (Nivala, Finland, www.vesikolmio.fi), which serves water to 50,000 people and annually delivers 3,7 million m^3 of water, provided access to one of their ground water pumping stations. The system was installed in this station before the ultraviolet water purification system for a testing period of two months. During the two-month testing period the system was capable of capturing multiple holograms that contained microparticles. An example result is shown in Fig. 3. The 3D locations of all of the particles in the field of view can be obtained through automated means [Kempkes et al., 2009, Pitkäaho et al., 2014].

 ISBN 978-0-9934207-2-6

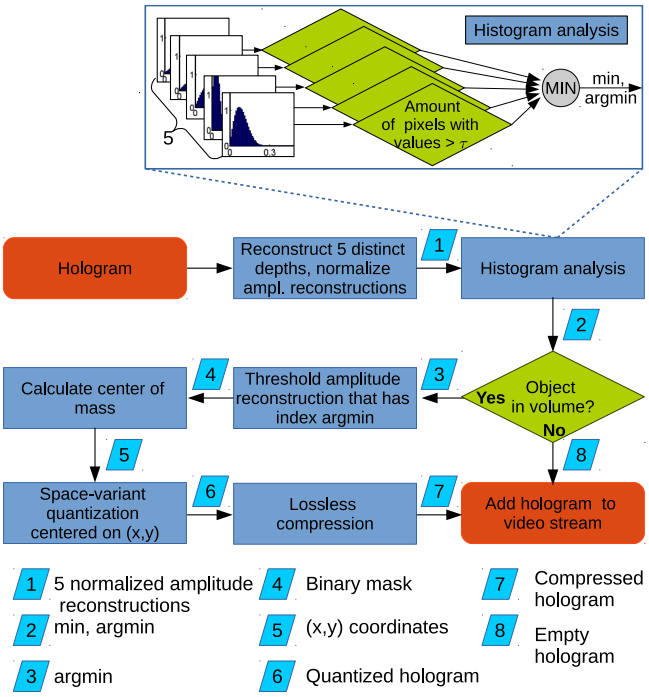


Figure 1: Compression principle. The input for the compression algorithm is a subtraction hologram and the output is a compressed hologram. The inset shows how the histogram analysis is executed.

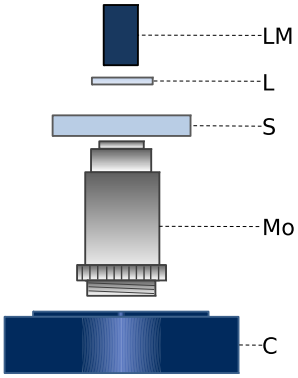


Figure 2: Imaging sensor components. Light from the laser module (LM) is collimated by the lens (L) and transmitted through an aperture containing the sample (S). Magnification is realized with the microscope objective (MO) and the hologram is captured with the digital camera (C).

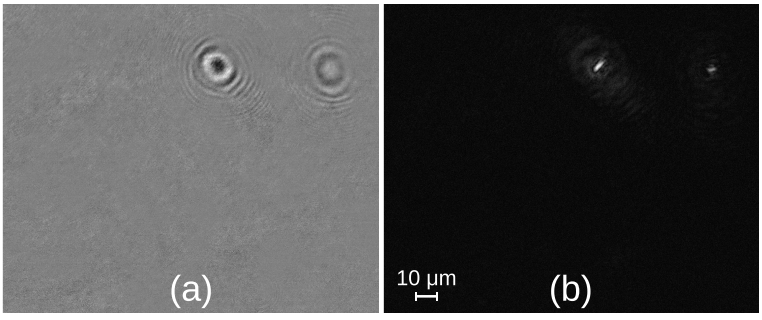


Figure 3: (a) subtraction of two temporally different holograms, (b) intensity reconstruction at 159 mm from the hologram plane where a single microscopic object is in focus.

 ISBN 978-0-9934207-2-6

4 Conclusions

In this paper, we described a system that satisfies the requirements of an on-line DHM sensor system that can be used in distributed water quality monitoring. An example implementation of the system was described and results from an active water potable water facility were shown.

Acknowledgments

This publication has emanated from research conducted with the financial support of an Irish Research Council (IRC) Postgraduate Scholarship, of Science Foundation Ireland (SFI) under grant no. 13/CDA/2224, and Kerttu Saalasti Foundation.

References

[Garcia-Sucerquia et al., 2006] Garcia-Sucerquia, J., Xu, W., Jericho, S. K., Klages, P., Jericho, M. H., and Kreuzer, H. J. (2006). Digital in-line holographic microscopy. *Applied optics*, 45(5):836–850.

[Javidi et al., 2005] Javidi, B., Moon, I., Yeom, S., and Carapezza, E. (2005). Three-dimensional imaging and recognition of microorganism using single-exposure on-line (seol) digital holography. *Optics Express*, 13(12):4492–4506.

[Kempkes et al., 2009] Kempkes, M., Darakis, E., Khanam, T., Rajendran, A., Kariwala, V., Mazzotti, M., Naughton, T. J., and Asundi, A. K. (2009). Three dimensional digital holographic profiling of micro-fibers. *Optics Express*, 17(4):2938–2943.

[Kujawinska et al., 2014] Kujawinska, M., Kozacki, T., Falldorf, C., Meeser, T., Hennelly, B. M., Garbat, P., Zaperty, W., Niemelä, M., Finke, G., Kowiel, M., and Naughton, T. (2014). Multiwavefront digital holographic television. *Optics express*, 22(3):2324–2336.

[Lopez-Roldan et al., 2013] Lopez-Roldan, R., Tusell, P., Cortina, J. L., and Courtois, S. (2013). On-line bacteriological detection in water. *TrAC Trends in Analytical Chemistry*, 44:46–57.

[Mudanyali et al., 2010] Mudanyali, O., Oztoprak, C., Tseng, D., Erlinger, A., and Ozcan, A. (2010). Detection of waterborne parasites using field-portable and cost-effective lensfree microscopy. *Lab on a Chip*, 10(18):2419–2423.

[Pitkäaho et al., 2014] Pitkäaho, T., Niemelä, M., and Pitkäkangas, V. (2014). Partially coherent digital in-line holographic microscopy in characterization of a microscopic target. *Applied optics*, 53(15):3233–3240.

[Restrepo and Garcia-Sucerquia, 2012] Restrepo, J. F. and Garcia-Sucerquia, J. (2012). Automatic three-dimensional tracking of particles with high-numerical-aperture digital lensless holographic microscopy. *Opt. Lett.*, 37(4):752–754.

[World Health Organization and others, 2004] World Health Organization and others (2004). Water, sanitation and hygiene links to health: facts and figures.

[World Health Organization and others, 2010] World Health Organization and others (2010). Progress and challenges on water and health: the role of the protocol on water and health. In *Proceedings of the 5th Ministerial Conference on Environment and Health Parma*, pages 10–12.

[Wyatt, 1968] Wyatt, P. J. (1968). Differential light scattering: a physical method for identifying living bacterial cells. *Applied optics*, 7(10):1879–1896.

Towards Dense Collaborative Mapping using RGBD Sensors

Louis Gallagher [*†] and John B. McDonald

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

Abstract

Development of collaborative, perception driven autonomous systems requires the ability for collaborators to compute a rich, shared representation of the environment, and their place in it, in real-time. Using this shared representation, collaborators can communicate geometric, semantic and dynamic information about the environment across frames of reference to one another. Existing state-of-the art dense mapping systems provide a good starting point for developing a collaborative mapping system, however, no system currently covers collaborative mapping directly. In this paper, we introduce our approach to dense collaborative mapping, offering an introduction to the problem, a discussion of the key challenges involved in developing such a system and an analysis of preliminary results.

Keywords: Dense, SLAM, Reconstruction, Mapping, Collaborative.

1 Introduction

The aim of dense visual simultaneous localisation and mapping (VSLAM) is to recover a dense reconstruction of a scene from a freely moving visual sensor. This is achieved through the continued fusion of measurements into a single representation, whilst simultaneously tracking the motion of the sensor, all in real-time. The state-of-the-art in the field includes a multitude of systems offering large-scale, high-precision mapping and tracking capabilities [Dai et al., 2017, Whelan et al., 2015, Kerl et al., 2013, Whelan et al., 2014, Newcombe et al., 2011, Engel et al., 2014]. To date all of these dense SLAM systems have focused on single sensor mapping. In this paper we report on initial work to address the wider problem of collaborative, multi-sensor mapping and tracking which is essential to many robotics and augmented reality applications such as human robot interaction (HRI), cooperative robotics and multi-session mapping. Historically, collaborative mapping has been a recurring theme in the SLAM literature [Saeedi et al., 2016], though the concept has yet to be extended to the more contemporary setting of dense visual SLAM.

This paper reports on first results of in-progress research to extend the ElasticFusion (EF) single sensor dense mapping system to allow for multi-sensor collaborative mapping and tracking using RGBD sensors. The contributions of the paper are: (i) a discussion of the challenges involved in extending EF to allow for full multi-sensor collaborative mapping, (ii) a description of our proposed multi-sensor EF framework and the components implemented to date; and (iii) a qualitative comparison between multi and single sensor EF.

The remainder of the paper is structured as follows. Section 2 provides a brief summary of the elements of the EF algorithm pertinent to understanding our proposed extensions. Section 3 discusses the challenges and proposed solutions to allow this extension, and provides details of the subset of the solutions that we have implemented to date. Section 4 presents the multi-sensor dense mapping capabilities of the current system, and provdes an initial comparison of its outputs to single-sensor EF. Finally, in Section 5 we give concluding remarks and discuss future research directions.

[*]This research is funded by the Irish Research Council, Project ID: GOIPG/2016/1320

[†]louis.gallagher.2013@mumail.ie

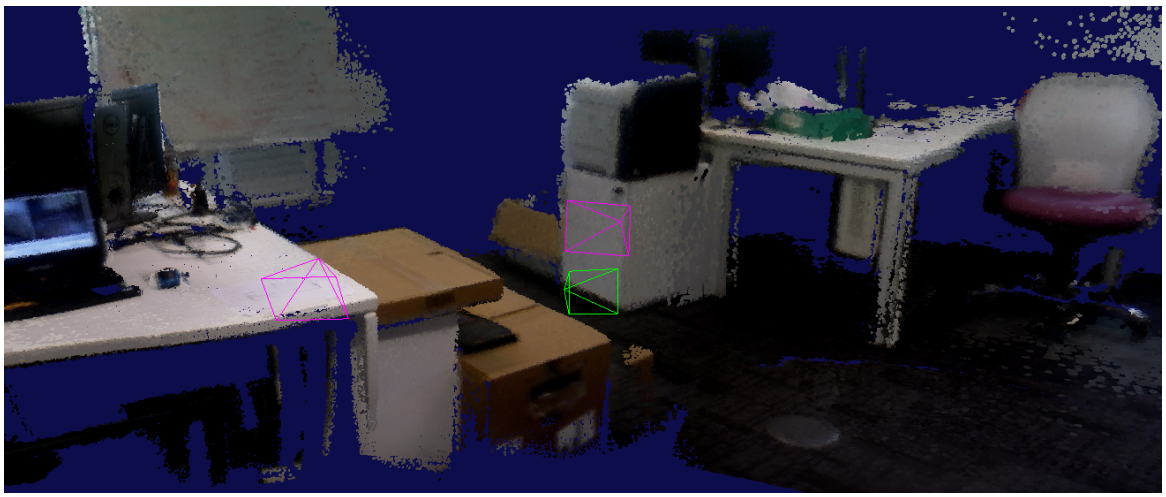


Figure 1: A three sensor collaborative mapping session in our system. All sensors started with the same initial pose but proceeded along independent trajectories

2 Background

In this section we give an overview of the EF dense mapping system, although the reader will find a more comprehensive treatment in [Whelan et al., 2015]. EF takes a point-based fusion approach to dense mapping. As an RGBD sensor traverses an environment its measurements are fused into a single model, internally represented by an unstructured list of surface elements (*surfels*), $\mathscr{M}$. Each surfel in $\mathscr{M}$ is an estimate of a discrete point on the underlying continuous surface contained in the environment being mapped. $\mathscr{M}$ is split along temporal lines into two distinct sublists; Θ, containing active surfels that have been observed recently, and Φ, containing inactive surfels that have not been observed in a period of time δ_t.

At each time step, the global pose of the sensor, $P_t \in \mathbb{SE}_3$, at time t is estimated by aligning the *active* model-predicted surface, derived from Θ using P_{t-1}, to the surface contained in the latest sensor frame at time t. This alignment yields a transformation that is applied to P_{t-1} to give P_t. Once P_t is resolved the latest frame can be fused into Θ, and thus into $\mathscr{M}$.

Assuming that a global loop closure has not occured at the current time step, a local loop closure between Θ and Φ is sought, reactivating inactive surfels and maintaining local surface consistency. Local loops are closed by performing a surface registration between the portion of Θ in view of P_t to the portion of Φ in the same view.

Occassionally, the sensor drifts too far for the estimated model to be corrected by the local loop closure mechanism. To solve this problem a randomised fern-encoding database containing discriminative views of the scene is maintained. The database is searched for a view matching the current active model-predicted view. If a matching view is found then a global loop closure is determined by solving a *model-to-model* surface registration between the surfaces underlaid within the matching views, akin to local loop closure.

In the case that either a local or a global loop closure is achieved the model is non-rigidly deformed by applying a space deforming graph to $\mathscr{M}$ [Whelan et al., 2015].

Surface registrations in EF are performed using a local alignment technique. An objective function is defined over both geometric and photometric constraints between the two surfaces, parameterised by T^k, the k^{th} estimate of the transformation $T \in \mathbb{SE}_3$ that aligns them. This objective defines an error surface, E, with respect to T. Through iterative non-linear least-squares, using Gauss-Newton optimisation and a three-level coarse-to-fine pyramid scheme, the objective is minimized by updating T^k in a direction of descent along E, yielding increasingly refined approximations of T.

 ISBN 978-0-9934207-2-6

3 Extending ElasticFusion for Multi-Sensor SLAM

In extending EF to permit collaborative mapping we identify two distinct phases of processing for any given input stream. This distinction arises from the fact that in collaborative mapping there is, in general, no common frame of reference to begin with, and hence each sensor's initial pose is unknown relative to the other sensors. Therefore, at the outset, the system assumes that each sensor is positioned at the origin of a frame of reference that is independent to that of other sensors. During this initial phase each sensor input is essentially processed via an independent EF mapping pipeline. As mapping progresses the aim is for global inter-sensor loop closures to occur, thereby providing the necessary transformations between the sensor submaps. These transformations permit alignment of submaps into a common frame of reference which makes it possible to perform multi-sensor fusion into a single global map. In order to concentrate on the development required for this second phase of processing, in this paper we constrain our multi-sensor datasets such that each sensor starts with the same initial pose. Thus from the outset we assume a single global map with multiple independently moving sensors. Therefore, our system to date, deals with the post-alignment stage of collaborative mapping.

Hence, we take a phased approach to extending EF, where in the first phase of the extension we assume datasets that are constrained in the manner described above. Multiple sensors can then fuse measurements into, and track against, a common global surfel map. Our representation for multiple sensor mapping is a tuple of the form $< \mathscr{M}, \{P_t^i\} >$ for surfel map $\mathscr{M}$ and sensor poses $\{P_t^i\}$, where each $P_t^i \in \mathbb{SE}_3$ represents the pose of sensor i at time t. Local surface consistency is maintained in the same manner as discussed in Section 2 with the exception that the active region of $\mathscr{M}$ contains the surfels in $\mathscr{M}$ that are active with respect to at least one sensor. For detecting global loop closures all sensors keep a common fern encoding database, loops can then be detected and closed in the same way as was described in Section 2.

Following this, we will concentrate on seperating out the sensor's temporal windows such that each sensor defines its own active and inactive map regions. Having one active region for all sensors leads to inefficient view predictions and temporal incoherence, preventing sensors from closing local loops in certain situations. For example when two sensors following independent trajectories intersect, with each sensor transitioning to a portion of the map that the other has kept active, no local loop closures will occur. Once the temporal windows have been separated multi-sensor mapping and tracking can then continue as before. To maintain separate temporal windows for each sensor we introduce the concept of a *context*. A context is a triple of the form $< \Theta_t^i, \Phi_t^i, P_t^i >$, where $\Theta_t^i \subset \mathscr{M}$ and $\Phi_t^i \subset \mathscr{M}$ represent the regions of $\mathscr{M}$ that are active and inactive w.r.t sensor i at time t. As before $P_t^i \in \mathbb{SE}_3$ represents the pose of sensor i at time t. Thus collaborative mapping can be represented as a tuple of the form $< \mathscr{M}, \{\zeta_i\} >$, for surfel map $\mathscr{M}$ and contexts $\{\zeta_i\}$.

In the final phase of extension we will focus on removing the constraint that each sensor must start with the same initial pose. Thus, initially each sensor will be associated with its own frame of reference denoted by $\mathscr{F}_i =< \mathscr{M}_i, \{\zeta_j^i\} >$ for surfel map $\mathscr{M}_i$ and contexts $\{\zeta_j^i\}$. By solving the global alignment problem between maps, frames of reference can be merged, and so, as mapping proceedes and inter-map loop closures occur, our representation will tend towards a single global map.

4 Experiments

In this section we report the first results of our approach. In the absence of ground-truth multi-sensor VSLAM datasets we use single-sensor EF to generate scene reconstructions and sensor poses. We then measure the deviation between these reconstructions and pose estimates and those outputted by our system as we incrementally increase the number of sensors collaboratively mapping. All experiments were run on a machine with an NVidia GeForce GTX 1080 ti GPU, 11GB of GDDR5 VRAM, an 8 core Intel i7-7700K CPU running at 4.20GHz and 16GB of DDR4 system memory.

We give a qualitative comparison between our system and EF using a custom lab dataset. The dataset consists of four sequences through the same scene. Each sequence was captured with an ASUS Xtion pro live depth sensor running at $30hz$. The initial pose of the sensor is the same across all sequences. The first sequence is used in conjunction with single-sensor EF to compute a reconstruction of the scene. We also use single-sensor EF to compute sensor poses for the other three sequences. Then, we compare the single-sensor reconstruction

 ISBN 978-0-9934207-2-6

and sensor poses to those computed by our system under the different permutations of the other three sequences collaboratively mapping and tracking. Table 1 summarises this data. The reader is encouraged to watch the accompanying video for a clearer visualisation of both this dataset and dense collaborative mapping in general (`https://youtu.be/qYNpP_5Vp7I`).

	2/3	2/4	3/4	2/3/4
surface reconstruction deviation	$0.1310m$	$0.0810m$	$0.1114m$	$0.1113m$
ATE RMSE	$0.012m$/$0.046m$	$0.015m$/$0.057m$	$0.069m$/$0.057m$	$0.018m$/$0.058m$/$0.089m$

Table 1: The first row gives the numbers of the sequences being used in the collaboration. The second row gives the per collaboration mean distance from each point to the nearest point in the single-sensor reconstruction. In the last row we give the per sequence ATE RMSE for each collaboration.

5 Discussion

We have reported on our initial work on a dense collaborative mapping system, demonstrating its capabilities through a comparative analysis with EF. In future work we will focus on improving the temporal coherence of our system and increasing its generality. To address the former issue, each sensor will define its own active map region, allowing us to maintain local surface consistency even in cases where sensors enter each others active regions. To address the latter issue we plan to leverage a global alignment technique, *fast global registration* [Zhou et al., 2016], to align sensor specific maps into a single global map on the fly. An important aspect of future work will be the creation of synthetic ground-truth datasets, allowing us to measure the performance of our system.

References

[Dai et al., 2017] Dai, A., Nießner, M., Zollhöfer, M., Izadi, S., and Theobalt, C. (2017). Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM Trans. Graph.*, 36(3):24:1–24:18.

[Engel et al., 2014] Engel, J., Schöps, T., and Cremers, D. (2014). LSD-SLAM: Large-scale direct monocular SLAM. In *European Conference on Computer Vision (ECCV)*.

[Kerl et al., 2013] Kerl, C., Sturm, J., and Cremers, D. (2013). Dense visual slam for rgb-d cameras. In *Proc. of the Int. Conf. on Intelligent Robot Systems (IROS)*.

[Newcombe et al., 2011] Newcombe, R. A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., Kohli, P., Shotton, J., Hodges, S., and Fitzgibbon, A. (2011). Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the 2011 10th IEEE Intl. Symposium on Mixed and Augmented Reality*, ISMAR '11, pages 127–136, Washington, DC, USA. IEEE Computer Society.

[Saeedi et al., 2016] Saeedi, S., Trentini, M., Seto, M., and Li, H. (2016). Multiple-robot simultaneous localization and mapping: A review. *J. of Field Robotics*, 33(1):3–46.

[Whelan et al., 2014] Whelan, T., Kaess, M., Johannsson, H., Fallon, M., Leonard, J., and McDonald, J. (2014). Real-time large scale dense RGB-D SLAM with volumetric fusion. *Intl. J. of Robotics Research, IJRR*.

[Whelan et al., 2015] Whelan, T., Leutenegger, S., Moreno, R. S., Glocker, B., and Davison, A. (2015). Elasticfusion: Dense slam without a pose graph. In *Proceedings of Robotics: Science and Systems*, Rome, Italy.

[Zhou et al., 2016] Zhou, Q., Park, J., and Koltun, V. (2016). Fast global registration. In *European Conference on Computer Vision (ECCV)*, pages 766–782.

 ISBN 978-0-9934207-2-6

Video Based Piano Music Transcription

Robert McCaffrey & Kenneth Dawson-Howe

School of Computer Science and Statistics, Trinity College, University of Dublin

Abstract

This paper presents an early version of a system for the automatic transcription of video footage of a piano performance into sheet music. The work presented focuses on the (rarely addressed) image processing part of the task. In order to allow the system to be evaluated, a range of videos were captured and annotated with ground truth (spanning various levels of difficulty in terms of the piano pieces). Initial results gave an average precision of 79% and an average recall of 94%.

Keywords: Imaging, Image Processing, Machine Vision, Piano transcription

1 Introduction

The ability to automatically determine what notes are being played creates a range of new potential applications to help both those learning to play musical instruments, and those who are already proficient. It is easy to envisage the notion of an automatic musical tutor application which advises students on mistakes they are making and how to improve their playing. It is equally easy to envisage a system which automatically creates sheet music for an advanced musician who is composing a new piece. All of these applications require that we be able to determine what notes are being played, and this has mainly been addressed through audio processing in the past.

In monophonic audio signals, where at most one note is sounding at a time, audio signal processing for estimating the pitches and durations occurred by an acoustic piano can be considered solved [Klapuri, 2004]. In polyphonic signals, several sounds occur simultaneously within the signal; this results in a signal that needs to be separated before the estimation can even begin. Sound separation is extremely complex, especially with signals containing different pitches at similar levels; it is difficult for an algorithm to determine the exact point at which one pitch begins and another pitch ends. However, it can be accomplished to a degree through widely used techniques in audio processing known as multiple F_0 estimations [Klapuri, 2004] to identify the musical (harmonic complex) tones and rhythm parsing (such as periodicity transforms [Sethares and Staley, 1999]) to determine the duration of each note.

Due to the presence of polyphony in music, with each concurrent note having harmonic complex tones, the precision of recognition has not reached the accuracy required for real-world applications. However, there is potential to also use a video signal and combine information about when notes are pressed and released (e.g. on a keyboard). There is presently only one publication in the area of automated music transcription from video which is a work by Akbari and Cheng from 2015 [Akbari and Cheng, 2015]. However, they do not cover the requirements of having a reference to the beat that the performer is intending to play and it is, therefore, likely that a reference to the beat of the music may be hard-coded in the solution as a default value (which would mean that the problem addressed is more restricted). While an accuracy of 95% was reported, testing was only done using a single simple piece ("Twinkle Twinkle Little Star") at a variety of tempos.

The system presented here: a) addresses the issue of music's underlying pulse (beat) and how it is intrinsically related to transcribing (whether in the hands of a human or a machine), and, b) evaluates the system performance over a much wider range of musical pieces.

 ISBN 978-0-9934207-2-6

2 Location of Keyboard & Keys, and Determination of Beat

An overview of the initial processing required is shown in Figure 1.

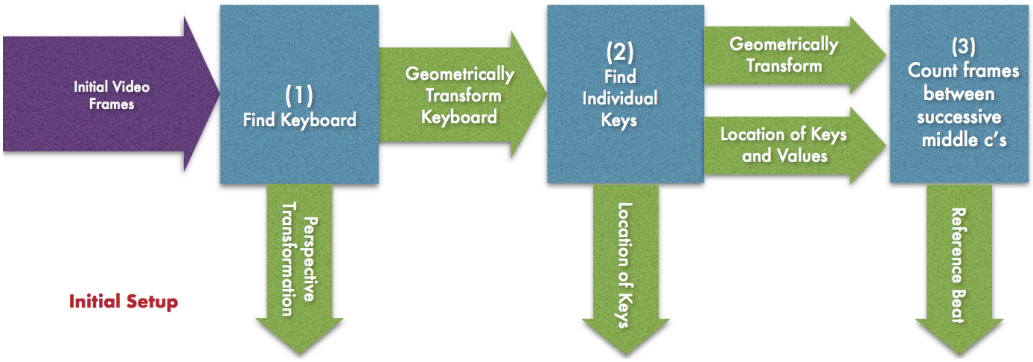


Figure 1: The steps involved in the initial phase of processing are shown, identifying the keyboard location, the location of the keys and the reference beat.

The keyboard is located using edge detection followed by probabilistic Hough transform. Parallel lines are located above and below the keys (See Figure 2 (left)). These lines can be identified relatively easily as the upper two-thirds of the region: between them is a mixture of black and white keys, whereas the lower one-third has just white keys. Having identified these lines, we can then distinguish the black and white keys using Otsu thresholding; identifying a quadrilateral (using the black keys at each end of the keyboard) which can then be used to transform the image to an 'aerial' view (See Figure 2 (right)). The individual black keys are found by thresholding, and the location of individual white keys can be defined based on the standard keyboard layout or may be refined through (faint) edge detection.

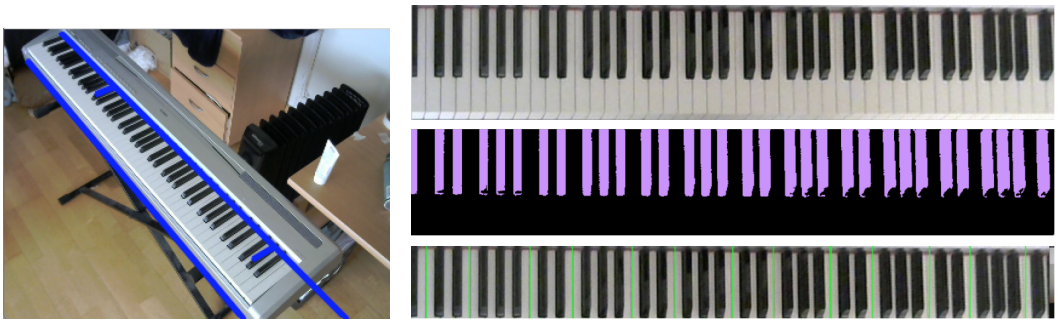


Figure 2: (left) The main lines located, showing the location of the keyboard. (right) The geometrically transformed keyboard (top), the black keys (middle; in purple) and the white keys separated by black keys or a green line (bottom).

The final stage in calibration is to get the user to identify the beat that they are intending to play to. This is done by asking the player to repeatedly depress the middle C note at the same interval that they would set a metronome. This can be analysed to identify the average number of frames per beat.

3 Identifying Notes and Music Transcription

Once the keyboard and keys have been located, and the beat identified, we can proceed to locate notes played by the pianist and convert these to sheet music. In fact, we are looking for two types of event: a key press and a key release. These events can be in turn stored as MIDI (Musical Instrument Digital Interface) events using a library such as the C++ library *Midifile* and sheet music can be generated from the MIDI file. A summary of the steps to generate the sheet music from video is shown in Figure 3.

 ISBN 978-0-9934207-2-6

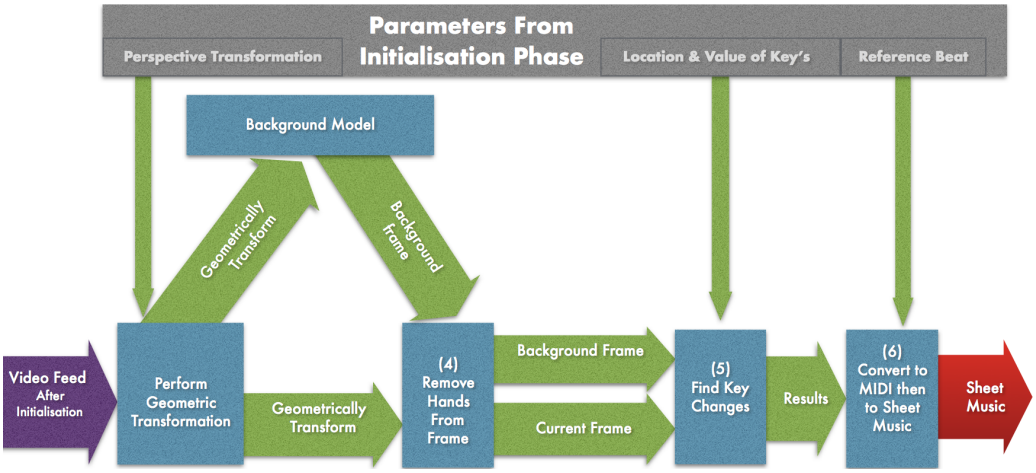


Figure 3: The processing steps involved in converting a video of a piano piece being played into sheet music.

The geometric transformation of the keyboard is performed first (using the perspective transformation identified during the initial calibration phase). This is followed by the use of a median background model to identify dynamic changes in the scene, and by the identification of skin regions in the scene (which should be dropped from the dynamic changes as they represent the hands of the pianist). Any skin pixel detection algorithm can be used, although to deal with differing illumination arguably the colour of the skin pixels should be learnt (or at least updated) based on the hands observed in the scene. To fill any small gaps and to ensure that shadow regions are not included, we dilate the binary skin pixels (See Figure 4) before removing these pixels from the dynamic changes to be considered further.

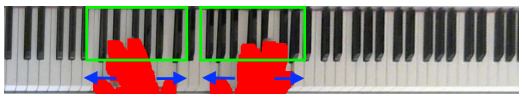


Figure 4: The location of the hands (as determined using skin detection and motion detection) are shown in red. These pixels are ignored when assessing which keys are pressed. However, the proximity of the hand places limits on the possible keys that can be pressed (shown by the green boxes and defined by the blue arrows).

The identification of key presses and releases is done by an analysis of the remaining differences between the background model and the current frame. This is analysed on a key-by-key basis for the upper parts of the keys only. Through experimentation we found that key presses (and releases) did not cause the surfaces of the keys under normal illumination to change sufficiently to be reliably identified. Instead we found that by placing the camera at an angle to the keyboard we were able to see more/less of the side of the adjacent key to the one being pressed/released and this provided a much more reliable cue to identify which keys had been pressed/released. See Figure 5

4 Dataset and Evaluation

There are no available video datasets of piano transcription containing ground truth (of key presses and releases) that could be used as a baseline for evaluation. Hence, we created our dataset with a range of piano pieces of varying levels of difficulty (Grade 1-6, from the Associated Board of the Royal Schools of Music grading system) at a constant tempo of 80 b.p.m (beats per minute). We identified all key changes and automatically determine the number of true positives (TP; key changes identified correctly), false negatives (FN; key changes missed), and false positives (FP; key changes identified which did not occur). From this we compute $Precision = \frac{TP}{TP+FP}$ and $Recall = \frac{TP}{TP+FN}$.

Figure 5: The keys currently depressed in one frame of a video are highlighted in blue.

We had anticipated that performance would dis-improve with the complexity of the piece, but this is not the

 ISBN 978-0-9934207-2-6

Table 1: Performance Metrics

Grade	Name	Precision	Recall
1	Dance (Elena Malycheva)	65.51	86.36
2	Twinkle, Twinkle Little Star, Nursery Rhymes (Mozart)	86.51	97.64
3	Bagatelle No. 25 in A minor, Fur Elise (Beethoven)	83.69	99.48
4	Prelude in C major, BWV 846 (Bach, Johann Sebastian)	84.61	93.07
5	Solfeggio in C minor, H.220 (Bach, Carl Philipp Emanuel)	84.67	90.20
6	Prelude, Op. 28, No. 20 (Chopin)	67.34	94.82
	Average	78.72	93.57
	Median	84.15	93.95

case. The number of occasions on which key changes were omitted due to occlusion (See Figure 6) were quite limited. Further study is required with additional datasets containing pieces with far greater complexity (beginning with diploma/undergraduate level and progressing onto those expected of a concert pianist) to investigate the affect of occlusion for an image processing solution.

5 Conclusions

There is potential to use video to aid in the automatic transcription of music and in many ways it has more potential than audio for the development of tools to aid learners (as it could be used to analyse hand placement and finger usage as well as key changes).

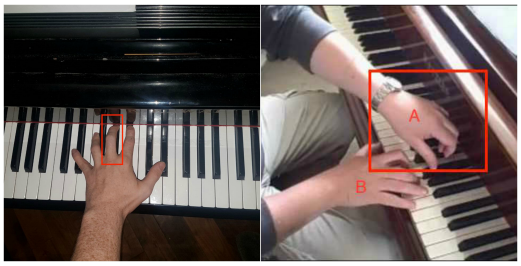


Figure 6: For some more advanced pieces the position of the hands may occlude the piano keys being pressed. In such cases only the audio can be used to identify when these keys are pressed and released.

References

[Akbari and Cheng, 2015] Akbari, M. and Cheng, H. (2015). Real-time piano music transcription based on computer vision. *IEEE Transactions on Multimedia*, 17(12):2113 – 2121.

[Klapuri, 2004] Klapuri, A. P. (2004). Automatic Music Transcription as We Know it Today. *Journal of New Music Research*, 33(3):269–282.

[Sethares and Staley, 1999] Sethares, W. A. and Staley, T. W. (1999). *Periodicity Transforms*.

Structure Based Matching between Aerial and Map Images using Brightness- and Rotation-Invariant Curve Features

Yoshikatsu Nakajima and Hideo Saito

Graduate School of Science and Technology, Keio University, Japan
{nakajima, saito}@hvrl.ics.keio.ac.jp

Abstract

Since conventional feature descriptor algorithms depend on brightness values, those algorithms cannot obtain point correspondences between aerial and map images where brightness values do not correspond but geometric structures do correspond. The research in this paper focused on obtaining point correspondences between those images. This study proposes a novel method by which curve features, edge shape, and road extraction by a CNN, independent on brightness value, produce robustness in image matching. Its effectiveness was confirmed through experiments on *Massachusetts Roads Dataset*.

Keywords: Feature descriptor, Binary features, Matching, Aerial images, Canny algorithm

1 Introduction

In recent years, due to widespread use of drones in taking aerial images, the demand for matching between aerial images and map images has increased. However, conventional feature descriptor algorithms, such as SIFT[Lowe, 2004], produce robust statistics for scaling and rotation, but do not enable one to find keypoint correspondences in images due to the changing brightness values around each keypoint. ORB[Rublee et al., 2011] and AKAZE[Alcantarilla and Solutions, 2011], conventional methods for describing features, shorten the processing time required for distance calculation during keypoint matching by describing features in binary form, however, these methods also depend on brightness values.

In this study, we propose a novel method by which curve features, edge shape, and road extraction by a CNN, independent on brightness value, produce robustness in image matching. Specifically, after extracting roads in an aerial image using a method proposed by Saito et al.[Saito et al., 2016], we detect its edge by using a Canny algorithm. Thereafter, we describe binary features for each point on the detected edge by using information as to whether or not the detected edge passes through each part of the ring-shaped feature descriptor (See Figure 1). Finally, we obtain robustness to rotation by setting the edge's orientation as the sum of vectors from the center of the ring to each part through which the edge passes.

2 Proposed Method

Figure 1 shows a flow of this proposed method. In this section, we describe the details.

2.1 Road Extraction

We employ a method proposed by Saito et al.[Saito et al., 2016], which segments an aerial image into road, building, and background classes by a CNN as pre-processing for describing features (See Figure 1, Road Extraction part). The output consists of gradient probabilities of the three classes. However, since this method describes features with curve shapes, we discard the building and background classes which have few features as a curve and process only the road class.

 ISBN 978-0-9934207-2-6

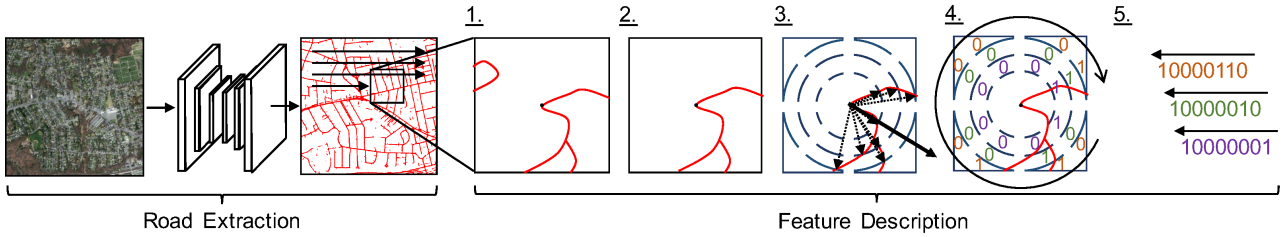


Figure 1: Flow of describing features with proposed method

2.2 Feature Desctiption

In this section, we describe the feature descriptions, which are the core of this research.

2.2.1 Generating Feature Descriptor

During pre-processing, we generate a feature descriptor $\mathscr{R}$ composed of several circles with the same center and different radii, as shown by the blue lines in Figure 1. A feature detector $\mathscr{R}$ consists of circles whose minimum radius is M and each radius is $M + k\sigma_{step}, k \subset \mathbb{Z}_{\geq 0}$. It is defined as a function which returns two values, as shown in the following equation (1) with $\boldsymbol{u} = (i, j)^{\mathrm{T}}, \boldsymbol{u} \subset \mathbb{R}^2$.

$$\mathscr{R}(\boldsymbol{u}) = (r(\boldsymbol{u}),\ a(\boldsymbol{u})) = \begin{cases} \left(\frac{\sqrt{i^2+j^2}-M}{\sigma_{step}} + 1,\ \lfloor \frac{\arctan(j,i)}{\theta} \rfloor + 1 \right) & \sqrt{i^2+j^2} \geq M \wedge \sqrt{i^2+j^2} \equiv M \pmod{\sigma_{step}} \\ (\phi,\ \phi) & otherwise \end{cases} \tag{1}$$

r indicates the number of circles to which $\boldsymbol{u}$ belongs counted from the circle with the smallest radius. a indicates the number of arcs to which $\boldsymbol{u}$ belongs counted from the top right arc when each circle is divided into arcs by the angle θ. The angle θ satisfies $l\theta = 360$ and $l \subset \mathbb{N}$. $\mathscr{R}(\boldsymbol{u})$ returns $(r(\boldsymbol{u}) = \phi, a(\boldsymbol{u}) = \phi)$ if $\boldsymbol{u}$ does not belong to any circle. Therefore, the range of r and a are $1 \leq r \leq (N - M)/\sigma_{step} + 1$ and $1 \leq a \leq 360/\theta$ except for ϕ, respectively, when the maximum radius in $\mathscr{R}$ is N. By dividing the circle by an angle, as the width of the arc becomes larger going away from the center, it is expected that robustness to a little projection distortion can be obtained in the process of describing features.

2.2.2 Detecting Edges

In this method, an edge image $\mathscr{E}(\boldsymbol{v})$ is generated by applying the Canny edge detector to the image $\mathscr{I}$. Here, the image $\mathscr{I}$ is the probability gradient of the road class as related to an aerial image, and it is a raw map as related to a map image. $\mathscr{E}(\boldsymbol{v})$ returns 1 if $\boldsymbol{v}$ is on an edge and 0 otherwise, where $\boldsymbol{v} = (x, y)^{\mathrm{T}}, \boldsymbol{v} \subset \mathbb{Z}^2$. We employed the Canny edge detector, because it is more robust to noise as compared to other edge detection methods. It is comprised of four procedures: Gauss smoothing, edge detection by the Sobel method, non-maximum value suppression, and hysteresis thresholding. For each $\boldsymbol{v}_i = (x_i, y_i)^{\mathrm{T}}$ satisfying $\mathscr{E}(\boldsymbol{v}_i) = 1$, we obtain $\mathscr{E}^c_{\boldsymbol{v}_i}$ by recursively processing eight neighbor search with $\boldsymbol{v}_i$ as the starting point after cropping $\mathscr{E}$ in the range of $x_i - N \leq x \leq x_i + N, y_i - N \leq y \leq y_i + N$, where $\boldsymbol{v}_i$ is the origin (See Figure 1, 1. and 2.). Note that $\mathscr{E}^c_{\boldsymbol{v}_i}$ has its origin at the center and has the same domain as $\mathscr{R}$.

2.2.3 Determining Orientation

Conventional methods of describing features, including SIFT[Lowe, 2004], are robust to rotation by computing orientation of each keypoint using intensity gradients around the keypoint. Even in this proposed method, the orientation of each point is determined in order to obtain robustness to rotation, however, intensity gradients are not used. Figure 1, part 3., shows a conceptual diagram on the determination of orientation. The orientation

 ISBN 978-0-9934207-2-6

$\boldsymbol{o}$ at $\mathscr{E}(\boldsymbol{v}_i)$ is determined with the following equation (2) by using $\mathscr{C}_{\boldsymbol{v}_i}$, which is a set of intersection of an edge and an arc.

$$\boldsymbol{o} = \begin{pmatrix} o_x \\ o_y \end{pmatrix} = N \cdot \frac{\sum_{\boldsymbol{w} \subset \mathscr{C}_{\boldsymbol{v}_i}} \boldsymbol{w}}{\| \sum_{\boldsymbol{w} \subset \mathscr{C}_{\boldsymbol{v}_i}} \boldsymbol{w} \|}, \ \boldsymbol{o} \subset \mathbb{R}^2, \ \mathscr{C}_{\boldsymbol{v}_i} = \left\{ \boldsymbol{w} \subset \mathbb{Z}^2 | \mathscr{R}(\boldsymbol{w}) \neq (\phi, \phi) \wedge \mathscr{E}^c_{\boldsymbol{v}_i}(\boldsymbol{w}) = 1 \right\} \tag{2}$$

Equation (2) shows that the vectors starting from the center of the circles and ending at the intersection of the edge and the arc are summed up and normalized the norm to N. Therefore, $\boldsymbol{o}$ calculated by the equation (2) satisfies the following equation (3).

$$\mathscr{R}(\boldsymbol{o}) = (r(\boldsymbol{o}) = \frac{N - M}{\sigma_{step}} + 1, \ a(\boldsymbol{o}) = a_o) \tag{3}$$

Since the results of the Canny edge detector include various elements, such as straight lines and isolated points, curves cannot always be easily detected. Elements other than a curve are difficult to differentiate when describing features based on edge information. To solve the problem, we do not describe features for the point $\boldsymbol{v}_i$ which satisfies $\| \sum_{\boldsymbol{w} \subset \mathscr{C}_{\boldsymbol{v}_i}} \boldsymbol{w} \| < \delta_{norm}$ for threshold δ_{norm}. If the edge around $\boldsymbol{v}_i$ is an element that does not have sufficient features, such as a straight line or an isolated point, the norm of the vector indicating the orientation is considered to be sufficiently small, so that elements not having those sufficient features are excluded from processing targets.

2.2.4 Describing Binary Features

In this study, in order to shorten the processing time required for distance calculation during keypoint matching, we describe features of each point $\boldsymbol{v}_i$ in binary form. First, the binary test τ is defined as the following equation (4) using a set $\mathscr{U}_{r_i, a_j}$.

$$\tau_{\boldsymbol{v}_i}(r, \ a) = \begin{cases} 1 & \exists \boldsymbol{u} \subset \mathscr{U}_{r_i, a_j}, \ \mathscr{E}^c_{\boldsymbol{v}_i}(\boldsymbol{u}) = 1 \\ 0 & otherwise \end{cases}, \ \mathscr{U}_{r_i, a_j} = \left\{ \boldsymbol{u} \subset \mathbb{Z}^2 | \mathscr{R}(\boldsymbol{u}) = (r_i, \ a_j) \right\} \tag{4}$$

The binary test τ returns the value of 1 or 0 by determining whether or not the edge intersects with the arc (r_i, a_j) (See Figure 1, 4.). Finally, binary features of a point $\boldsymbol{v}_i$ is described with a function f defined by the following equation (5).

$$f(\boldsymbol{v}_i) = \sum_{1 \leq r \leq \frac{N-M}{\sigma_{step}} + 1} 2^{(r-1) \cdot \frac{360}{\theta}} g(\boldsymbol{v}_i, \ r), \ g(\boldsymbol{v}_i, \ k) = \sum_{1 \leq a \leq \frac{360}{\theta}} 2^{a-1} \tau_{\boldsymbol{v}_i}(k, \ a_o + a - 1 \pmod{\frac{360}{\theta}}) \tag{5}$$

After describing binary features of each circle in a feature descriptor $\mathscr{R}$, in binary form, using binary test $\tau_{\boldsymbol{v}_i}$, and with function g, we describe binary features of a point $\boldsymbol{v}_i$ by combining features of each circle with function f (See Figure 1, part 4. and 5.).

3 Experiments

In this experiment, *Massachusetts Roads Dataset* [Mnih, 2013] proposed by Mnih was used for aerial images. We captured map images corresponding to the dataset from Google Maps. The image size was aligned to 300 x 300 for both aerial images and map images. Each parameter in this experiment was $\theta = 10$, $M = 4$, $N = 30$, $\sigma_{step} = 1$, $\sigma_{norm} = 5.2$. As for the parameters of the Canny edge detector, we set the lower limit as 30 and the upper limit as 60. We conducted experiments under the following environments: CPU: Intel Core i7-6950X, GPU: GeForse GTX1080, and RAM: 128GB. After applying RANSAC to correspondences obtained by brute force matching based on hamming distance of binary features, we drew point correspondences as matching results.

 ISBN 978-0-9934207-2-6

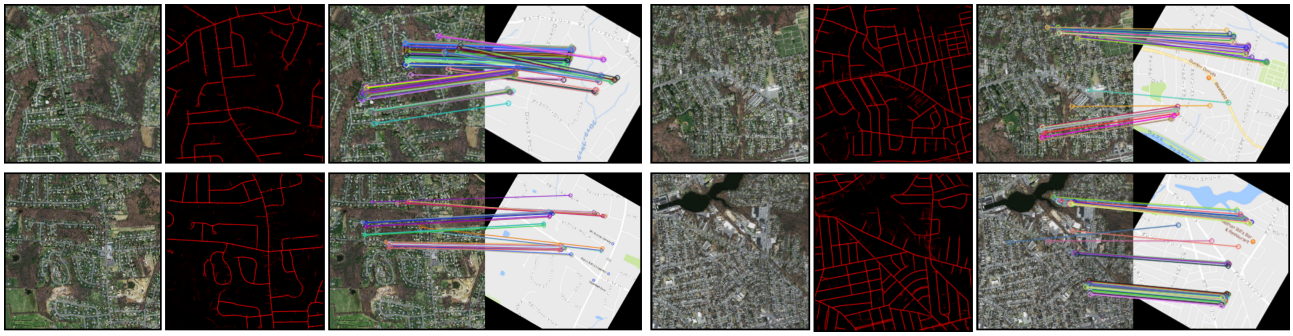


Figure 2: Example of matching results between aerial image and map (left to right: input aerial image, road extraction result, matching result)

3.1 Results

Figure 2 shows some experimental results of matching between aerial images and map images with this proposed method. We rotated map images to verify the robustness to rotation of this proposed method. Since the mean value of the number of inliers was 61.8, we were able to obtain sufficient point correspondences. Figure 2 also shows point correspondences were not biased on part of the image, but acquired from the whole. The processing time required for road extraction was 2.76 seconds on average, and the processing time required for the feature description was 0.58 seconds on average per image. Furthermore, the processing time required for matching features was 0.07 seconds on average. This is because the processing cost of distance calculation between features was reduced by describing features in binary form and using hamming distance.

4 Conclusion

In this study, we proposed a novel method by which curve features, edge shape, and road extraction by a CNN, independent on brightness value, produced robustness in image matching. We have experimentally confirmed its effectiveness and responsiveness. Our future research will focus on obtaining robustness to scaling.

Acknowledgment This research presentation is supported in part by a research assistantship of a Grant-in-Aid to the Program for Leading Graduate School for "Science for Development of Super Mature Society" from MEXT in Japan.

References

[Alcantarilla and Solutions, 2011] Alcantarilla, P. F. and Solutions, T. (2011). Fast explicit diffusion for accelerated features in nonlinear scale spaces. *IEEE Trans. Patt. Anal. Mach. Intell*, 34(7):1281–1298.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110.

[Mnih, 2013] Mnih, V. (2013). *Machine Learning for Aerial Image Labeling*. PhD thesis, University of Toronto.

[Rublee et al., 2011] Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2564–2571.

[Saito et al., 2016] Saito, S., Yamashita, T., and Aoki, Y. (2016). Multiple object extraction from aerial imagery with convolutional neural networks. *Electronic Imaging*, 2016(10):1–9.

A Dataset for Irish Sign Language Recognition

Marlon Oliveira*, Houssem Chatbri†, Ylva Ferstl ‡, Mohamed Farouk⋆, Suzanne Little†, Noel E. O'Connor† and Alistair Sutherland*

*School of Computing, Dublin City University, Ireland
†Insight Centre for Data Analytics, Dublin City University, Ireland
‡ADAPT Centre, School of Computer Science and Statistics, Trinity College Dublin, Ireland
⋆College of Computing & Information Technology, Arab Academy for Science & Technology, Egypt

Abstract

We introduce a new image dataset for Irish Sign Language (ISL) recognition. We filmed human subjects performing ISL hand-shapes and movements, resulting in 468 videos. Then, we extracted frames from the videos. This resulted in a total of 58,114 images for the 23 common hand-shapes from the ISL language. This dataset is a part of our ongoing work on ISL recognition using pattern recognition methods. In addition to the dataset, we report experiments using Principal Component Analysis (PCA) where we reached 95% recognition accuracy.

Keywords: Irish Sign Language, Pattern Recognition, Image Dataset

1 Introduction

Irish Sign Language (ISL) is an indigenous language that is used by around 5,000 Deaf people in the Republic of Ireland and 1,500 in Northern Ireland. In addition, it is known by 50,000 non-Deaf people. ISL is not based on English or Irish, it is a language in its own right [Leeson and Saeed, 2012].

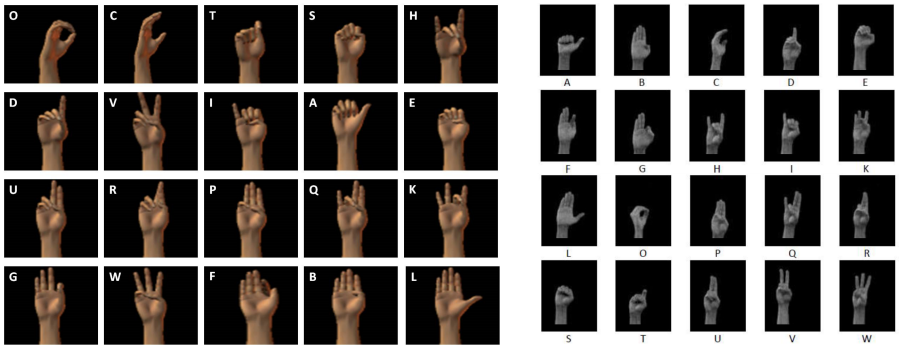
ISL contains more than 5000 signs. Each sign consists of a hand-shape and a motion in 3D space. There are around 23 basic, common hand-shapes in ISL and each hand-shape is labelled with a different letter of the alphabet. These hand-shapes can be seen in a wide range of possible angles in 3D space. The remaining three letters of the alphabet, 'J', 'X' and 'Z' are used to label gestures involving motion and actually use one of the 23 hand-shapes.

Computer vision provides the technology to assist people who use ISL with tools such as automatic transcript, human-machine interaction, machine translation, etc. In order to design such tools, large amounts of data are necessary for training and testing the system. In this paper, we introduce a new image dataset for ISL recognition. The dataset contains 58,114 images for the 23 ISL hand-shapes. In addition to the dataset being our main contribution in this paper, we also report recognition experiments using Principal Component Analysis (PCA).

Earlier works in this area have used rather smaller datasets. For instance, Farouk et al. proposed two ISL datasets [Farouk, 2015]. The first dataset is composed of computer generated images, produced by a the Poser software by SmithMicro; the total number of images is 920. The second dataset is composed of real hands, and has a total of 1620 images. Both datasets represent only 20 ISL hand-shapes as illustrated in Figure 1 (excluding 'm', 'n' and 'y and the dynamic shapes 'J', 'X' and 'Z'). The images show the hand and arm of a signer against a uniform black background.

Compared to previous works on ISL, our dataset is larger and contains all hand-shapes. It is then fit to train and test classifiers for ISL recognition. The rest of the paper is organised as follows: Sec. 2 details our data collection procedure and the final dataset. Sec. 3 reports a recognition experiment using PCA. We end the paper in Sec. 4 with concluding remarks and future work.

 ISBN 978-0-9934207-2-6



(a) Computer generated (b) Real hands

Figure 1: Datasets for ISL created by [Farouk, 2015]

2 The Irish Sign Language hand-shape (ISL-HS) dataset

The ISL-HS dataset contains real hand images, unlike synthetic images used in previous works. ISL-HS is composed of 23 hand-shapes combined with different motions.

To build the dataset, we recorded short videos. We asked 6 people (3 males and 3 females) to perform the finger spelling ISL hand-shapes. Each shape was recorded 3 times.

Each of the 23 hand-shaped was performed by moving the arm in an arc from the vertical to the horizontal position. This was performed to simulate rotated hand-shapes that can occur in real word conversations. For the 3 motion gestures 'J', 'X' and 'Z' there was no rotation, only the motion indicated in Figure 2. All the hand-shapes in our dataset, apart from the 3 with motion, are rotated in a plane.

The videos were converted into frames. Frames were converted to grayscale and the background was removed from the frame using a pixel-value threshold. This produced frames contain only the arm and the hand.

The number of frames for each video depends on the time taken by the human subject to perform the gesture. Videos were recorded at 30 frames per second (fps) and a resolution of 640×480 pixels. The device used to record the videos was an Apple iPhone 7. The videos were saved with *.mov* extension. The video format is RGB24.

The illumination sources were a combination of natural and artificial, as the videos were recorded in our laboratory of post-graduate computing students. Illumination was different for each person, because they were recorded at different times of day and on different days.

In total, 468 videos were recorded. From these videos we obtained a total of 58,114 frames, consisting of 52,688 frames for the rotated shapes and 5,426 for the 'J', 'X' and 'Z'. Figure 2 shows cropped images of our ISL-HS dataset, and Figure 3 shows the class distribution across the image dataset. The variation observed in Figure 3 is due to the speed variation among the subjects when performing the ISL hand-shapes and rotating them. Note that the letter '*X*' has the lowest number of frames because this is a dynamic feature with a short motion.

We are releasing the dataset online[1] and providing both videos and images.

3 Principal Component Analysis (PCA)

PCA is an efficient method for dimensionality reduction [Han and Liu, 2014]. It uses the covariance matrix of the data to create a space known as an eigenspace. Each dimension in the space is represented by an eigenvector of the covariance matrix. The number of eigenvectors required to represent the full data is considerably lower than the dimensionality of the original data.

[1] https://github.com/marlondcu/ISL

 ISBN 978-0-9934207-2-6

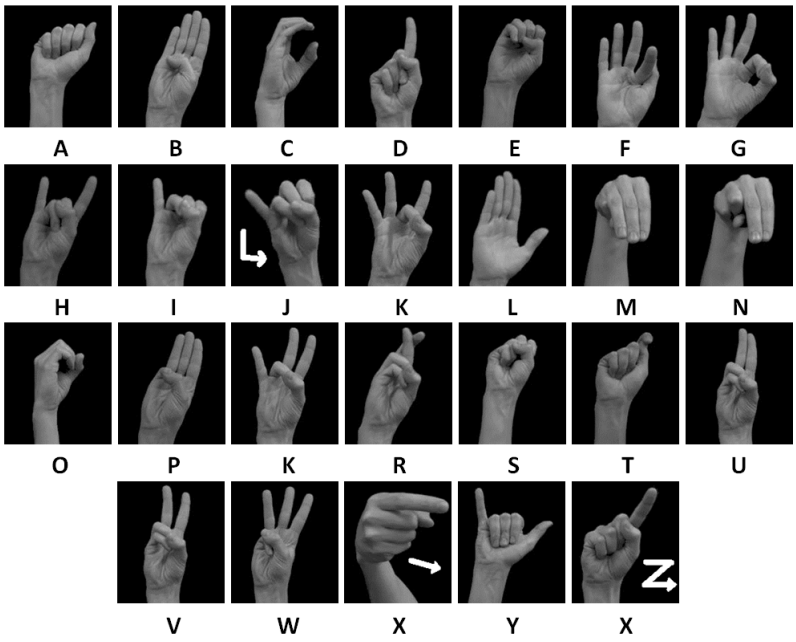


Figure 2: Irish Sign Language hand-shapes

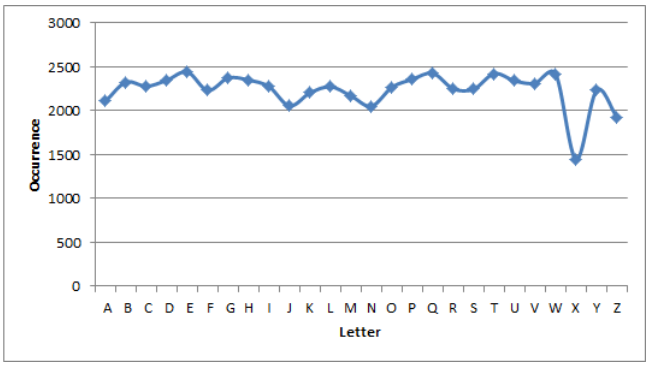


Figure 3: Frequency of the different hand-shapes in the dataset

In order to apply PCA over our training dataset we combine all the images into the same array and then compute PCA. Since each image has 640×480, we re-sized them to 160×120 pixels. When vectorised this becomes 19,200 pixels in a row array, for each image.

In this experiment we considered only the 23 common hand-shapes with rotation. Then images corresponding to the letters 'J', 'X' and 'Z' were not used. The dataset used contains 52.688 in total. This dataset was divided into a training set and a testing set, by iterating through the images and taking one image for training and the next for testing, and so on. Thus, both our training and testing datasets contain 26,344 images.

By projecting the images from the training set into the most significant D_i eigenvectors, we obtain a D_i-dimensional space containing N_{im} points for each pose angle. Each point represents an image. In this work we tested different numbers of eigenvectors and measured how it affects the accuracy.

In order to classify the correct hand-shape we used the k-Nearest Neighbour (k-NN) algorithm, with $k = 1$ and Euclidean distance. We projected each testing image into the training dataset eigenspace and classified according to the nearest point (shortest Euclidean distance).

The accuracy in recognising the correct hand-shape strongly depends on the number of the eigenvectors (dimensions) considered. For example, assuming $D_i = 15$, we obtained 88% of recognition accuracy, using more eigenvectors the accuracy increases as well. e.g. for $D_i = 29$ we obtained 95%. Figure 4 shows the

 ISBN 978-0-9934207-2-6

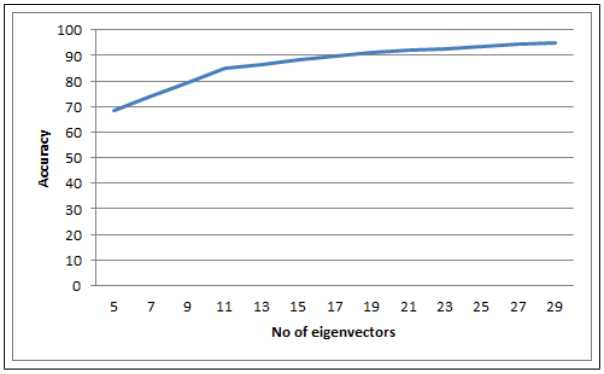


Figure 4: Accuracy according to the number of eigenvectors

accuracy according to the number of eigenvectors.

In this work blurring was applied over the images. A Gaussian kernel of size 36×36 pixels was used with standard deviation equal to 60. Using blurring was motivated by earlier results by Farouk [Farouk et al., 2013], which showed that such image filtering is beneficial for PCA accuracy.

4 Conclusions

In this work, we proposed an Irish Sign Language hand sign dataset (ISL-HS). Compared to previous works, our dataset is larger, more complete and contains rotation variation. In addition, we reported a recognition experiment using PCA, and we were able to reach 95% of recognition accuracy.

In the future, we are planning to try different classification methods in addition to PCA (e.g. Convolutional Neural Networks), and apply recognition to videos in addition to images to leverage the dynamic aspect of some of the ISL hand-shapes.

Acknowledgments

This research was funded by CAPES/Brazilian Science without Borders, process no.: 9064-13-3. The ADAPT Centre for Digital Content Technology is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund. This research also emanated from a grant in part from the IRC under Grant no. GOIPD/2016/61, in part from the EU H2020 Programme under grant agreement no. 688099 (Cloud-LSVA), and in part from SFI under Grant no. SFI/12/RC/2289 (Insight). The authors would like to thank Dr. Robert Smith from the Institute of Technology Blanchardstown for the feedback he gave about Irish Sign Language.

References

[Farouk, 2015] Farouk, M. (2015). *Principal Component Pyramids using Image Blurring for Nonlinearity Reduction in Hand Shape Recognition*. PhD thesis, Dublin City University, Ireland.

[Farouk et al., 2013] Farouk, M., Sutherland, A., and Shokry, A. (2013). Nonlinearity Reduction of Manifolds using Gaussian Blur for Handshape Recognition based on Multi-Dimensional Grids. *ICPRAM*.

[Han and Liu, 2014] Han, F. and Liu, H. (2014). Scale-invariant sparse PCA on high-dimensional meta-elliptical data. *Journal of the American Statistical Association*, 109(505):275–287.

[Leeson and Saeed, 2012] Leeson, L. and Saeed, J. I. (2012). *Irish Sign Language : A Cognitive Linguistic Account*. Edinburgh University Press.

Extending the Bag-of-Words Representation with Neighboring Local Features and Deep Convolutional Features

Daniel Manger and Dieter Willersinn

Fraunhofer IOSB, Karlsruhe, Germany

Abstract

In this work, we propose and compare two methods to extend the bag-of-words representation which is still widely used in the domain of content-based image retrieval where a query image is used to search for those images in a large image database that show the same object or scene. To this end, typically, local features such as SIFT are quantized and treated independently to leverage an inverted file indexing scheme for speedup. As the quantization of local features impairs their discriminability, the ability to retrieve the relevant database images is decreasing in larger databases. We address this issue by extending every quantized local feature with information from its local spatial neighborhood. More precisely, we make use of two approaches widely used for global image features: the Fisher Vector representation aggregating the neighboring local features and a representation based on pooling features from deep convolutional neural network layer outputs. Using four public datasets, we evaluate the representations in terms of their performance after quantization.

Keywords: Content-based Image Retrieval, Bag-of-Words, Spatial Context of Local Features

1 Introduction

While becoming apparent in many successful applications, e.g. apps for recognizing items based on snapshots of mobile devices, content-based image retrieval (CBIR) with local features is still limited to small databases compared to today's web-scale flood of pictures. The main reason for that is, that the methods to find common local content in a pair of images based on local features such as SIFT do not scale for searching in databases with billions of images. Nevertheless, since the seminal work of [Sivic and Zisserman, 2003] introducing the codebook-based quantization of features into visual words in order to manage the image retrieval problems with text retrieval methods, many work has been done to enrich the discriminative power of their bag-of-words (BoW) model. Alternatively, various *global* image representations based on local features have been proposed for image retrieval. The Fisher Vector encoding [Perronnin et al., 2010], for instance, aggregates local features into a high-dimensional embedding followed by a compression - typically PCA and whitening - to encode an image into a compact fix-sized code. Finally, the recent advances of convolutional neural networks (CNN) have led to many approaches using deep-learned features - either out-of-the-box or by pooling responses from fully-connected or convolutional layers. Since both CNN- and Fisher Vector-based representations are targeted for very compact codes of e.g. just 64 floats for one image, it becomes obvious that retrieving very small objects surrounded by plenty of heavily cluttered background becomes difficult for large databases.

In this work, we therefore analyze specific combinations of the three approaches by extending each quantized local feature in the BoW model with more context information from the respective local neighborhood encoded with a Fisher Vector- or CNN-based representation.

 ISBN 978-0-9934207-2-6

2 State of the Art

Many advancements of the bag-of-words model have been proposed which aim at different aspects of the image retrieval pipeline in order to incorporate more information into the retrieval process. In this work, we neglect methods refining the shortlist (re-ranking) and focus on those approaches that incorporate additional information into the inverted file indexing scheme (termed index) so that *all* images can benefit from. These can be separated into three strategies:

Extending the accumulator which holds bins for the scores of all the database images by new dimensions assuming that irrelevant features will spread along multiple bins of one database image while corresponding features will accumulate in one or few of the bins of a similar image. For instance, [Jegou et al., 2008] use orientation and scale information of SIFT features to push database images with features having consistent differences in scale and orientation compared to the query image.

Filtering of features: This keeps the accumulator compact (still one bin per database image) and adds additional information into the index to filter matches prior to casting votes into the accumulator. [Zhang et al., 2013] integrates information about the four closest features in the image coordinate space and during retrieval, each BoW match is further examined as to how many of the four neighboring features are consistent.

2D-Index: In order to overcome the runtime, performance and storage limits of both accumulator extension and filtering of features, [Zheng et al., 2014] uses a multi-index. The first dimension of the index is still dedicated to the BoW vectors while the second dimension is based on the color name descriptor [Khan et al., 2012], which is an 11-dimensional descriptor mapping color values to 11 categories. Using a Color-Codebook of size 200, every feature in the index is assigned up to the 100 closest Color-Words which however obviously eliminates the advantages of the second dimension because still up to 50% of the index has to be traversed.

In this paper, we target the latter strategy of integrating context information as a second dimension into the inverse file. However, with Fisher Vector encodings of local features on the one hand and CNN features on the other hand, we use different features as basis for the second dimension. Adding such a new dimension to the index is attractive in multiple aspects: In contrast to the filtering strategy, the runtime during retrieval can be optimized because only features which match both dimensions have to be considered for the accumulator leading to fewer memory accesses. Furthermore, retrieval accuracy can benefit from the second dimension because many incorrect matches of features are discarded that match w.r.t. the first dimension only (the quantized local feature descriptor) but not in the second dimension (the larger context of the feature).

3 Encoding context information

When encoding the larger context of a local 'central' feature into a context descriptor it is essential to not loose the existing invariances of the features (translation, scale and rotation for the SIFT features used in this work) for the final CBIR system. As first option, we encode local features which lie both in the spatial neighborhood and in nearby scales (yielding typically 10 to 100 features) using the Fisher Vector (FV) encoding. More specifically, we use their descriptors - reduced to 64D by PCA - and concatenate their spatial configuration relative to the central feature spending another four dimensions (scale-normalized distance, polar angle, feature scale ratio and difference of descriptor orientations). Using a GMM model with 32 mixtures, the 68D feature vectors are condensed into a fixed-size Fisher Vector of $62 \times 32 \times 2 = 4352$ dimensions.

As a second method to generate context descriptors, we pool the activations in the 512 channels of the last convolutional layer of the VGG16 network [Simonyan and Zisserman, 2014] in a rectangular region defined by the feature's position and scale. We use sum- and max-pooling since they proved successful for global CNN-based image retrieval [Babenko and Lempitsky, 2015, Tolias et al., 2015].

After reducing the Fisher Vectors to 512D with PCA and L2-normalizing the CNN features, we quantize both context features with separate Codebooks of sizes $10,000$ in order to obtain quantized context numbers.

 ISBN 978-0-9934207-2-6

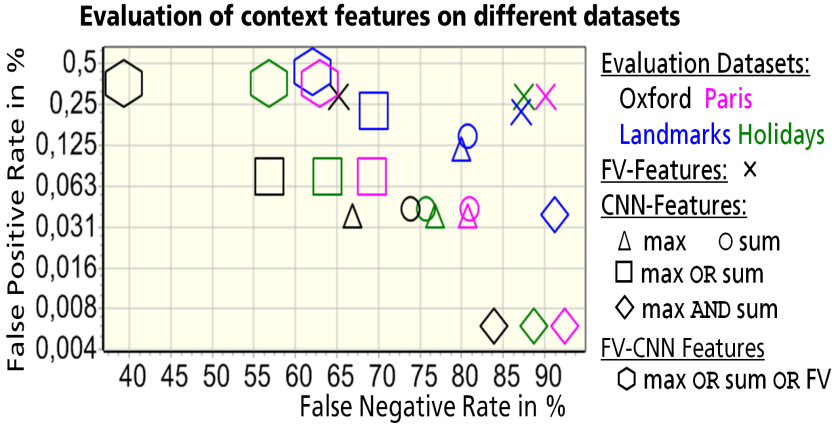


Figure 1: Overall results comparing the quantization of context features (Codebook size 10,000) based on Fisher Vectors (FV), CNNs and their combinations. Please note the logarithmic scale of the False Positive Rate (best viewed in color).

4 Evaluation framework and experiments

Instead of measuring the benefits with respect to the retrieval accuracy of an overall image retrieval system which would be very time consuming, we model the retrieval system's view to the features. More precisely, we consider the two possibilities every BoW match can be looked upon: either it is a correct match arising from a real object correspondence or it is an incorrect match originating from the quantization loss or random background clutter etc. Given the datasets, we therefore compile these two sets of feature pairs (correct and incorrect BoW matches) and evaluate the involved quantized context numbers accordingly, i.e. the feature pairs of a correct BoW match should also agree w.r.t. their quantized context number whereas for incorrect BoW matches, we want the context numbers to be different. We measure the *False Negative Rate* (FNR, the number of correct BoW matching pairs that are not quantized to the same value) and the *False Positive Rate* (FPR, the number of incorrect BoW matches that are quantized to the same value). Ideally, both FNR and FPR are low to not loose any recall and to skip all incorrect matches during retrieval, respectively. We additionally perform experiments by combining different context features using AND and OR combinations. In these cases, FNR and FPR are adapted accordingly, e.g. for AND, a False Negative occurs if for a correct BoW match none or only one context feature yields identical quantized values.

For experiments, we use public datasets often used in CBIR: Oxford5k (5,062 images, 11 different buildings), Paris6k (6,392 images, 11 different buildings), Holidays (1,491 images, 500 different scenes) and Landmarks ("clean" subset 35,224 images due to broken links, 586 landmarks). We extract SIFT features and apply the RootSift normalization [Arandjelović and Zisserman, 2012]. The features from the Oxford dataset are used to generate a visual Codebook of size 100,000 by hierarchical k-means clustering which is used for bag-of-words quantization of local features in all our experiments. We identify feature pairs for correct BoW matches using the annotation of the datasets (specifying pairs of images that show the same object or scene) and subsequently filter matches with spatial verification. Thus, for each of Oxford5k, Paris6k and Landmarks, we obtain some 600,000 correct BoW matches and some 80,000 for Holidays. Feature pairs for incorrect BoW matches are collected by randomly taking pairs of images (each time one image from Oxford5k and one from Paris6k), calculating the BoW matches and randomly keeping 30% of them to obtain about 600,000 pairs. Given the fact that the images from Oxford5k and Paris6k are taken in different cities, virtually all of the BoW matches are incorrect BoW matches. For results on Landmarks dataset, we collect incorrect BoW matches by randomly taking pairs of images from different landmarks and keeping 30% of the BoW matches.

We train all our models, i.e. the BoW- and Context-Codebook, the GMM for the Fisher Vector representation, the PCA and the quantizers with data from Oxford5k only.

 ISBN 978-0-9934207-2-6

Figure 1 condenses the results for both CNN-based and FV-based context features. As can be seen, the CNN-based context features outperform FV-based features both in terms of FNR and FPR for all datasets. Interestingly, the OR-combination of sum- and max-pooled CNN features further boosts FNR without sacrificing too much FPR compared to the Fisher Vectors. For example, the holidays dataset (green square) yields a FNR of 63.83% and a FPR of 0.0748%, which means 36 out of 100 correct BoW matches are preserved while only one incorrect BoW match out of 1,336 remains. Finally, OR-combining both sum- and max-pooled CNN and the FV-based context features offers another trade-off with a better FNR at more false positives.

5 Conclusion

In this work, we compared ways to increase the discriminability of bag-of-words based representations of local features in the context of image retrieval. We extended a local feature with more information from its larger neighborhood comparing a Fisher-Vector representation with a representation based on pooling features from deep convolutional neural network layer outputs. Representations based on CNN-features clearly outperformed the Fisher-Vectors and the combinations of different quantized features offer interesting trade-offs. The next step will be to evaluate the best representation in terms of the overall accuracy in a large-scale retrieval system with millions of images.

Acknowledgments

This work was supported by the German Ministry of Education and Research (BMBF) (grant number 13N14028).

References

[Arandjelović and Zisserman, 2012] Arandjelović, R. and Zisserman, A. (2012). Three things everyone should know to improve object retrieval. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2911–2918. IEEE.

[Babenko and Lempitsky, 2015] Babenko, A. and Lempitsky, V. (2015). Aggregating local deep features for image retrieval. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1269–1277.

[Jegou et al., 2008] Jegou, H., Douze, M., and Schmid, C. (2008). Hamming embedding and weak geometric consistency for large scale image search. In *European conference on computer vision*, pages 304–317. Springer.

[Khan et al., 2012] Khan, F. S., Anwer, R. M., Van De Weijer, J., Bagdanov, A. D., Vanrell, M., and Lopez, A. M. (2012). Color attributes for object detection. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3306–3313. IEEE.

[Perronnin et al., 2010] Perronnin, F., Sánchez, J., and Mensink, T. (2010). Improving the fisher kernel for large-scale image classification. In *European Conference on Computer Vision*, pages 143–156. Springer.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[Sivic and Zisserman, 2003] Sivic, J. and Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1470–1477. IEEE.

[Tolias et al., 2015] Tolias, G., Sicre, R., and Jégou, H. (2015). Particular object retrieval with integral max-pooling of cnn activations. *arXiv preprint arXiv:1511.05879*.

[Zhang et al., 2013] Zhang, S., Tian, Q., Huang, Q., Gao, W., and Rui, Y. (2013). Multi-order visual phrase for scalable image search. In *Proceedings of the Fifth International Conference on Internet Multimedia Computing and Service*, pages 145–149. ACM.

[Zheng et al., 2014] Zheng, L., Wang, S., Liu, Z., and Tian, Q. (2014). Packing and padding: Coupled multi-index for accurate image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1939–1946.

Local Shape and Moment Invariant Descriptor for Structured Images

Elena Ranguelova

Netherlands eScience Centre, Amsterdam, The Netherlands.

Abstract

Finding correspondences between two images to determine if they are from the same scene is a fundamental, yet challenging task. To cope with different viewpoints and lighting conditions, local salient regions are detected invariantly to transformations and encoded by descriptors such as Scale-invariant Feature Transform (SIFT) or Speeded-up Robust Features (SURF). While using image intensities around a single point, the centroid of each region, to compute SIFT-type descriptors often works well, we argue that for structured scenes it is beneficial to use descriptors based on the shape of the regions. We propose a 20-dimensional Shape and Moment Invariant (SMI) descriptor and show that it outperforms the 64-dimensional SURF on two benchmark datasets in precision with similar or higher accuracy, while having a better scalability.

Keywords: image matching, affine-invariant descriptor, shape invariants, moment invariants

1 Introduction

Automatically determining whether two images depict partially the same physical scene is a fundamental computer vision problem such as baseline stereo matching, image retrieval, etc. [Escalera et al., 2007, Matas et al., 2002]). The approach is to *detect* local (to cope with partial overlap) features, followed by matching of their *descriptors*. A class of such features are local regions, corresponding to the same image patches, detected independently in each of the two images. Many detectors and descriptors are invariant to photometric (due to different sensors and lighting) and affine geometric transformations (due to different viewpoints). In recent years, an approach of using large datasets of image patch correspondences has been established, [Snavely et al., 2008, Zagoruyko and Komodakis, 2015]. However, deep learning is not applicable when the *structured* images having homogeneous regions with distinctive boundaries, are only few. Such is the case, for example, in some scientific applications [Ranguelova, 2016].

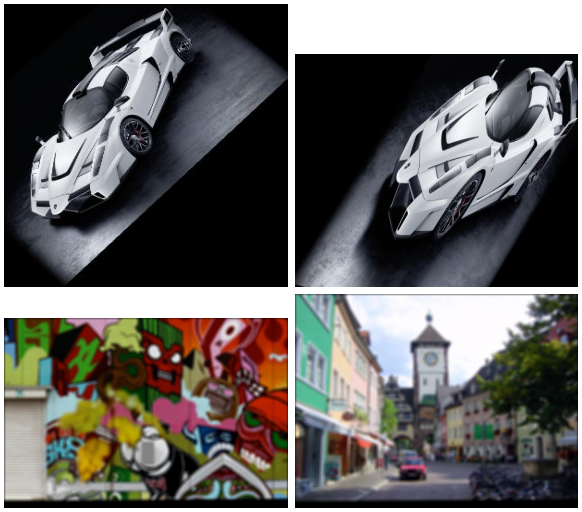


Figure 1: "Is it the same object or scene?" Matching an OxFrei image pair under different transformations using MSER regions. *Top image pair* (scale and viewpoint): SURF descriptor yields false negative (similarity score 0.096), while SMI true positive (0.89). *Bottom image pair* (blur): SURF descriptor yields false positive (similarity score 0.27), while SMI true negative (−0.11).

The Maximally Stable Extremal Regions (MSER) detector has become the standard in computer vision [Matas et al., 2002]. It is often used in combination with a histogram-of-gradients type descriptor such as Scale-invariant Feature Transform (SIFT) or Speeded-Up Robust Features (SURF) [Bay et al., 2008], computed from image intensities around the centroids of the MSER regions. We argue that using the shape information of the regions encoded by a *Shape and Moment Invariant (SMI)* descriptor is beneficial, compared to using image intensities around the central point of the region. Figure 1 illustrates two cases of image pairs, one depicting the same scene and the other not, where SMI outperforms SURF applied on pre-detected MSER regions.

 ISBN 978-0-9934207-2-6

2 Related work

The literature describes large number of local detectors and descriptors. For a recent introduction and overview the reader is referred to [Hassaballah et al., 2016]. Here, we mention very briefly only the closely related work.

A comparative performance evaluation of number of detectors has concluded that MSER is the best performing region detector for structured scenes [Mikolajczyk et al., 2005]. Since then, MSER has been integrated into MATLAB, OpenCV, VLFeat, etc. , making it the default baseline detector. However, despite its success, the detector has several drawbacks, which have been addressed by improved detectors, including the Data-driven Morphology Salient Regions detector (DMSR) [Ranguelova, 2016]. Here, we propose to use a Binary detector (BIN) using the first step of DMSR construction: data-driven binarization explained in [Ranguelova, 2016], with either all regions or only regions with large area ($A_{region} \geq f_A.A_{Image}$).

Another comparative performance evaluation of number of region descriptors has concluded that the "region-based SIFT descriptor" is the best performing for structured images [Mikolajczyk and Schmid, 2005]. Since we are interested in describing the shape of the detected regions, we have chosen efficient shape descriptors, known as moment invariants. Flusser et al. pointed our the dependency in the early set of 7 Hu moments and developed a coherent theory and general framework for derivation of Affine Moment Invariants (AMIs) using graph representation [Suk and Flusser, 2004, Flusser et al., 2009].

Research has been performed not only to determine the best region detector and descriptor, but also the best detector - descriptor combination. For example, the conclusion of the experiments in [Dahl et al., 2011] is that the best combination is DOG or MSER detector and SIFT (SURF was not included in the experiments) or DAISY descriptors. SURF has been introduced as an improvement over SIFT and since has become the standard of many computer vision software libraries, making it the default baseline descriptor choice [Bay et al., 2008]. Hence, we have chosen MSER - SURF as the baseline detector - descriptor combination.

3 Image matching with Shape and Moment Invariant descriptor

We propose a set of several Shape and Moment Invariants (SMI) derived from the binary shapes of the detected regions as a region descriptor. The SMI descriptor contains *shape invariants* and *moment invariants*.

Shape invariants. A shape of a region R_i can be described by a set of simple properties either of the original shape or of the equivalent (up to the second order moments) ellipse E_i. These are: the region's area a_i, the area a_i^c of the region's convex hull, the length μ_i of the major and ν_i of the minor axes of E_i and the distance ϕ_i between the foci of the ellipse. From these properties, a set of shape affine invariants are defined in Table 1.

Invariant	Definition	Description
Relative Area	$\tilde{a}_i = a_i / A$	region's area normalized by the image area A
Ratio Axes Lengths	$r_i = \nu_i / \mu_i$	ratio between E_i minor and major axes lengths
Eccentricity	$e_i = \phi_i / \mu_i$	$e_i \in [0, 1]$ (0 is a circle, 1 is a line segment.)
Solidity	$s_i = a_i / a_i^c$	proportion of the convex hull pixels, that are also in the region.

Table 1: Simple shape invariants.

Affine Moment Invariants. If $I(x, y)$ is a real-valued image with N points, the AMI functional is defined by

$$M(I) = \int_{-\infty}^{\infty} \prod_{k,j=1}^{N} C_{kj}^{n_{kj}} . \prod_{l=1}^{N} I(x_l, y_l) dx_l dy_l,$$

where n_{kj} are non-negative integers and $C_{kj} = x_k y_j - x_j y_k$ is the cross-product (graph edge) of points (nodes) (x_k, y_k) and (x_j, y_j), [Suk and Flusser, 2004]. For full details of the AMI's theory the reader is referred to [Flusser et al., 2009]. We use the set of 16 irreducible AMIs of $N = 4$th order, which are the functional coefficients $\{m_{ij}, j = 1 \ldots 16\}$, as implemented by the authors in an open source MATLAB software.

Hence, the final descriptor for the i-th region is a 20 element feature vector $SMI_i = (\tilde{a}_i, r_i, e_i, s_i, m_{i1}, \ldots, m_{i16})$.

 ISBN 978-0-9934207-2-6

Matching. Lets $SMI1$ and $SMI2$ be $n1 \times 20$ and $n2 \times 20$ matrices, where each row is the SMI descriptor for the $n1$ and $n2$ regions detected via MSER or BIN (all/largest) detector in the pair of images $< I1, I2 >$. We compare exhaustively $SMI1$ and $SMI2$ with Sum of square differences metric. The matching threshold for selection of the strongest matches is mt, the max ratio threshold for rejecting ambiguous matches is mr, the confidence of a match is mc and only unique matches are allowed. Then, we select the top quality matches above a cost threshold ct. From those, we estimate in it iterations the affine transformation $\tilde{T}$ between the two sets of points-centroids of the matching regions sets as average of nr runs with allowed max point distance md. The two images are then transformed $J2 = \tilde{T}(I1)$, $J1 = \tilde{T}^{-1}(I2)$ and a correlation $(cor[X,Y] = cov[X,Y]/\sqrt{var[X]var[Y]})$ between the original and transformed images is used for confirmation of a true match. If the average correlation similarity between both images and their transformed versions $(cor[I1,J1]+cor[I2,J2])/2$ is above a similarity threshold st, we declare the image pair $< I1, I2 >$ to be depicting (partially) the same scene.

Figure 2 illustrates the major steps of the image matching using BIN - SMI in case of viewpoint distortion. Note the better alignment in the right part of the images due to the larger number of correct matches there. The steps are the same when using MSER instead of the data-driven binarization or SURF instead of SMI descriptor.

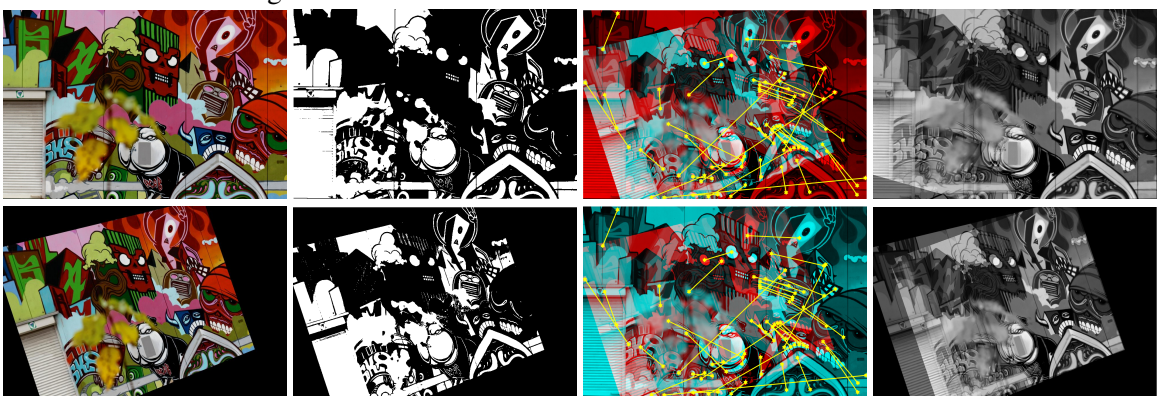


Figure 2: Matching two same scene images under viewpoint transformation using BIN region detector and SMI descriptor. *First column:* original images $I1, I2$; *second column:* binarization; *third column:* SMI descriptor-matched BIN regions used for transformation estimation (blend view with pseudocolours); *fourth column:* overlay of the original and transformed images $(I1, J1), (I2, J2)$.

4 Performance Evaluation

We have tested the performance of the MSER and BIN (all regions and only the largest) detectors in combinations with the SURF and SMI descriptors on two datasets: Oxford (VGG) [Mikolajczyk et al., 2005] and OxFrei [Ranguelova, 2016]. Each of the 4 structured image sequences of the Oxford set consists of 1 base and 5 increasingly distorted images. Each sequence can be used to test only one transformation T: viewpoint, scaling + rotation, decreased lighting and blur. OxFrei dataset overcomes the limitation: 9 structured scenes each with 21 images (original + 5 images for $4T$) using Oxford's real homographies. We compared all possible image pairs and assigned a flag *True/False* if a pair is depicting the same scene (see Section 3). The values of the parameters used in the evaluation have been determined experimentally: $mt = mr = 1$, $f_A = 2e-3$ (for BIN largest), $it = 1000$, $nr = 10$, $mc = 95$, $md = 8px$, $ct = 0.025$, $st = 0.25$. Figure 1 illustrates the result for 2 pairs and Figure 3 for all pairs (a pixel represents image pair and a square block a sequence) of the OxFrei dataset. Note the lower number of false positives and correlation similarity variance when using the SMI descriptor.

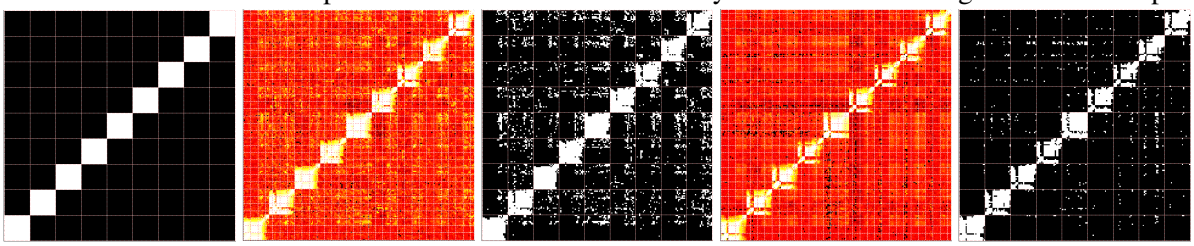


Figure 3: Matching all OxFrei pairs using MSER regions. "Is the image pair from the same scene?": *True*(white)/*False*(black). *First:* ground truth, *third:* SURF, *fifth:* SMI. Correlation similarity: the lighter, the higher. *Second:* SURF, *fourth:* SMI.

 ISBN 978-0-9934207-2-6

Table 2 summarizes the performance of the combinations of detectors and descriptors for the 2 datasets. When using the default MSER detector, it seems beneficial to combine with an SMI instead of the standard SURF descriptor as in both datasets almost all performance measures are improved. The BIN (all) detector does not outperform MSER in the matching task and using only the largest regions in BIN (largest) improves the recall at the expense of lower precision. The SMI descriptor achieves better precision in comparison to SURF independent of the detector or dataset. The best detector - descriptor combination is MSER - SMI when all measures are required to be high, especially the accuracy and precision.

Dataset	Oxford			OxFrei		
Detector - descriptor	Accuracy	Precision	Recall	Accuracy	Precision	Recall
MSER - SURF	**0.97**	0.97	**0.89**	0.90	0.53	**0.83**
MSER - **SMI**	**0.96**	**0.98**	**0.85**	**0.95**	**0.83**	**0.74**
BIN (All) - SURF	0.95	0.95	0.85	0.85	0.41	0.63
BIN (All) - **SMI**	0.89	**1**	0.58	0.91	0.73	0.32
BIN (Largest) - **SMI**	0.93	0.93	0.77	0.85	0.38	0.52

Table 2: Performance of salient region detectors and descriptors on the Oxford and OxFrei datasets.

The developed MATLAB software is released open source, [Ranguelova, 2017].

5 Conclusion

It is not possible to use deep learning of image patches approach when trying to automatically determine whether two images depict (partially) the same scene if only a few images are available. It is beneficial not to discard the shape of the salient regions detected by the detector. For structured scenes, a descriptor based on the properties of the binary regions alone performs better than one based on image intensities. The proposed shape and moment invariant descriptor, SMI, is a good choice when false positives should be minimal. In combination with the MSER detector, SMI achieves the highest precision and good accuracy and recall. In the future, the matching performance of the SMI descriptor should be tested on larger datasets.

References

[Bay et al., 2008] Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359.

[Dahl et al., 2011] Dahl, A. L., AanÃęs, H., and Pedersen, K. S. (2011). Finding the best feature detector-descriptor combination. In *3DIMPVT*, pages 318–325. IEEE Computer Society.

[Escalera et al., 2007] Escalera, S., Radeva, P., and Pujol, O. (2007). Complex salient regions for computer vision problems. In *CVPR*.

[Flusser et al., 2009] Flusser, J., Suk, T., and Zitova, B. (2009). *Moments and Moment Invariants in Pattern Recognition*. Wiley.

[Hassaballah et al., 2016] Hassaballah, M., Abdelmgeid, A. A., and Alshazly, H. A. (2016). Image Features Detection, Description and Matching. In Awad, A. and Hassaballah, M., editors, *Image Feature detectors and Descriptors: Foundations and Applications*, volume 630 of *Studies in Computational Intelligence*, pages 11–45. Springer.

[Matas et al., 2002] Matas, J., Chum, O., Urban, M., and Pajdla, T. (2002). Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In *Proceedings BMVC*, pages 36.1–36.10.

[Mikolajczyk et al., 2005] Mikolajczyk, K. et al. (2005). A comparison of affine region detectors. *Int. J. of CV*, 65(1-2):43–72.

[Mikolajczyk and Schmid, 2005] Mikolajczyk, K. and Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(10):1615–1630.

[Ranguelova, 2017] Ranguelova (2017). Salientdescriptor-matlab, matlab code. https://doi.org/10.5281/zenodo.835476.

[Ranguelova, 2016] Ranguelova, E. (2016). A Salient Region Detector for Structured Images. In *Proceedings of IEEE/ACS 13th Int. Conf. of Computer Systems and Applications (AICCSA)*, pages 1–8.

[Snavely et al., 2008] Snavely, N., Seitz, S. M., and Szeliski, R. (2008). Modeling the world from internet photo collections. *Int. J. Comput. Vision*, 80(2):189–210.

[Suk and Flusser, 2004] Suk, T. and Flusser, J. (2004). Graph method for generating affine moment invariants. In *17th International Conference on Pattern Recognition, ICPR 2004, Cambridge, UK, August 23-26, 2004.*, pages 192–195.

[Zagoruyko and Komodakis, 2015] Zagoruyko, S. and Komodakis, N. (2015). Learning to compare image patches via convolutional neural networks. *CoRR*, abs/1504.03641.

Open Source Dataset and Deep Learning Models for Online Digit Gesture Recognition on Touchscreens

Philip J. Corr, Guenole C. Silvestre and Chris J. Bleakley

School of Computer Science, University College Dublin, Belfield, Dublin 4, Ireland.

Abstract

This paper presents an evaluation of deep neural networks for recognition of digits entered by users on a smartphone touchscreen. A new large dataset of Arabic numerals was collected for training and evaluation of the network. The dataset consists of spatial and temporal touch data recorded for 80 digits entered by 260 users. Two neural network models were investigated. The first model was a 2D convolutional neural (ConvNet) network applied to bitmaps of the glpyhs created by interpolation of the sensed screen touches and its topology is similar to that of previously published models for offline handwriting recognition from scanned images. The second model used a 1D ConvNet architecture but was applied to the sequence of polar vectors connecting the touch points. The models were found to provide accuracies of 98.50% and 95.86%, respectively. The second model was much simpler, providing a reduction in the number of parameters from 1,663,370 to 287,690. The dataset has been made available to the community as an open source resource.

1 Introduction

Touchscreens are now pervasively used in smartphones and computing tablets. Text input on a touchscreen commonly uses a virtual keyboard. Unfortunately, the virtual keyboard occupies a significant portion of the screen. This loss of screen is noticeable on smartphones but is especially problematic on smaller devices, such as smartwatches. Text entry by means of handwriting using the finger or thumb has the advantage that the gestures can be performed on top of a screen image or background. Smaller screens can be easily accommodated by entering characters individually, one top of another [Kienzle and Hinckley, 2013].

Previous work on handwriting recognition has mainly focused on processing images of pen-on-paper writing, i.e. offline character recognition. Notably, the MNIST dataset was created using images of handwritten US census returns [LeCun et al., 1998]. Excellent recognition accuracy (99.2%) was demonstrated on the MNIST dataset using a convolutional neural network (ConvNet) [LeCun et al., 1998]. In contrast, online character recognition systems take input in the form of the continuously sensed position of the pen, finger, or thumb. Online systems have the advantage of recording temporal information as well as spatial information. To date, most work on online character recognition has focused on pen based systems [Guyon et al., 1991, Bengio et al., 1995, Verma et al., 2004, Bahlmann, 2006]. LeCun et al.'s paper proposed a ConvNet approach to the problem, achieving 96% accuracy. The method involved considerable preprocessing without which accuracy falls to 60%. The preprocessing step requires that the entire glyph is known *a priori*, removing the possibility of early recognition and completion of the glyph.

To date, there has been almost no work on using neural networks for online recognition of touchscreen handwriting using a finger or thumb. Our observation is that digits formed using a finger or thumb have greater variability than those formed using a pen, with more examples of poorly formed glyphs. Most likely, this is due to the users having better fine grained control of the pen. Furthermore, to enable operation on low cost, small form factor devices it is desirable that the resource footprint of the recognizer is low in terms of computational complexity and memory requirements. To date, an unexplored dimension of the problem is that online entry allows early recognition and confirmation of the character entered, enabling faster text entry.

 ISBN 978-0-9934207-2-6

Herein, we report on a investigation seeking to address these challenges. A large dataset of Arabic numerals was collected using a smartphone. A number of deep learning models were explored and their accuracy evaluated for the collected dataset. Of these architectures, two are reported herein. The first model uses an approach similar to offline character recognition systems, i.e. a 2D ConvNet taking the bitmap of the completed glyph as input. The second model uses a 1D ConvNet applied to the polar vector connecting touch positions. The accuracy and the size of the networks are reported herein together with an analysis of some of the errors. In addition, initial results on early digit recognition are provided. To the best of our knowledge, this is the first work to report on a low footprint recognizer using polar vector inputs for online finger or thumb touch digit recognition.

2 Dataset

A software application was developed to record the dataset. Prior to participation, subjects signed a consent form. The application firstly asked subjects to enter their age, sex, nationality and handedness. Each subject was then instructed to gesture digits on the touchscreen using their index finger. The digits 0 to 9 were entered four times. The sequence of digit entry was random. Instructions to the user were provided using voice synthesis to avoid suggesting a specific glyph rendering. The process was repeated for input using the thumb while holding the device with the same hand. This is to allow for applications where the user may only have one hand free. Cubic interpolation of touches during gesture input was rendered on the screen to provide visual feedback to the subject and to compute arclengths. The screen was initially blank (white) and the gestures were displayed in black. The subject could use most of screen to gesture with small areas at the top and bottom reserved for instructions/interactions/guidance. The subject was permitted to erase and repeat the entry, if desired.

The dataset was acquired on a 4.7 inch iPhone 6 running iOS 10. Force touch data was not available. The touch panel characteristics are not publicly available, specifically the sampling frequency and spatial accuracy are unknown. Values of 60Hz and $\pm$1mm are typically reported (Optofidelity datasheet). Data was stored in a relational database. Subject details such as handedness, sex and age were recorded along with the associated glyphs. Glyphs were stored as a set of associated strokes, corresponding to a period when the subject's finger was in continuous contact with the device panel. The coordinate of each touch position was sampled by the touch panel device and this, along with the timestamp of the touch, was stored. The dataset was reviewed manually and any incorrectly entered glyphs were marked as invalid. The final dataset contained input from 260 subjects with a total of 20,217 digits gestured and demographic details are summarized in Table 1a.

3 Deep Learning Models

Two deep learning models were developed. One takes an offline glyph bitmap as input and the other takes the polar vectors connecting touch points as input. The models were implemented using Keras with TensorFlow backend and trained on a NVIDIA TITAN X GPU.

3.1 Model with Bitmap Input

The first architecture investigated, as listed in Table 1b, consisted of two convolutional layers and two fully connected layers. Each of the convolutional layers are followed by a rectified linear unit activation layer and a max pooling layer. In the convolutional layers, kernels of size 5x5 were used with a stride of 1. Padding was set to ensure the height and width of the output is the same as the input. The max pooling layers use non-overlapping windows of size 2x2. The result of this is that the output of the second max pooling layer is 7x7. The two fully connected layers come after the aforementioned layers. 50% dropout is used during training to prevent over fitting and a momentum optimizer, implementing a variation of stochastic gradient descent, was used to minimise the error. The learning rate used for this optimiser was 0.9. Exponential decay was used and the decay rate was set to 0.95. When running for 10 epochs the network took approximately 8 seconds to train on the NVIDIA TITAN X graphics card.

 ISBN 978-0-9934207-2-6

3.2 Model with Polar Vector Input

The coordinates of the touch samples were converted to a series of polar vectors. For each touch point, the vector to the next touch point was calculated. The angle of the vector was calculated as the angle to the positive x axis in the range $\pm\pi$ where $+\pi/2$ is vertically upwards. The length of the vector was expressed in pixels.

The network architecture is listed in Table 1c. The input sequences were padded with zeros so that they were they were all the same length as the longest sequence in the dataset, 130 points. Dropout layers with a dropout rate of 25% were used to avoid co-adaption of the training data and hence, to reduce overfitting. Max pooling layers with pool size of 2 were used to progressively reduce the number of parameters in the network and hence, reduce the computation required in the training process. In the convolutional layers a kernel size of 5 was used as this was found to capture local features from within the sequence. The activation function used was ReLU as it was found to provide the highest accuracy of the commonly used activation functions. Softmax was used in order to perform the final classification.

Three input cases were considered: angle-only, vector length-only, and both angle and length. Some of the glyphs include multiple strokes. Only the longest stroke was input to the network. This was found to give better accuracy than inputting the entire multi-stroke gesture. Training was considered finished when the validation accuracy did not change for 18 epochs. This typically occurred after 80 epochs.

(a) Database Demographic

Parameter	Number of Entries
Male	126
Female	134
Right Handed	228
Left Handed	32
Nationalities	12
Age Range	18 - 80

(b) 2D Model with Bitmap Input

Layers	Output Size	F #	P #
2D Convolution	28x28	32	832
Max Pooling	14x14	-	0
2D Convolution	14x14	64	51264
Max Pooling	7x7	-	0
Fully Connected	512	-	1,606,144
Dropout	512	-	0
Fully Connected	10	-	5130

(c) 1D Model with Polar Vector Input

Layer	Output Size	F #	P #
1D Convolution	126	32	352
Dropout	126	-	0
1D Convolution	122	32	5152
Max Pooling	61	-	0
Dropout	61	-	0
1D Convolution	57	64	10304
Max Pooling	28	-	0
Dropout	28	-	0
1D Convolution	28	128	41088
Max Pooling	14	-	0
Dropout	14	-	0
Flatten	1792	-	0
Fully Connected	128	-	229504
Dropout	128	-	0
Fully Connected	10	-	1290

Table 1: Dataset and Network Architectures. F# and P# refer to the number of features and number of parameters.

4 Results and Discussion

The networks were evaluated on the dataset using a 60% training set, 20% validation set and 20% test set split. The accuracy of the networks is listed in Table 3. It can be seen that the network with bitmap input gives highest accuracy. The accuracy is close to the results reported in [LeCun et al., 1998] for the NMIST dataset, suggesting that the network is able to cope with the variability of the finger and thumb touch gestures. In the case of the polar vector input, the best results are obtained by using both angle and distance data. Also for the polar vector model, using only the longest stroke provided better results than using the full multi-stroke gesture. This may be due to a dataset deficiency or the artificial concatenation of the multi-strokes. The size of the networks is compared in Table 3. The 2D network is clearly larger due to the number of points on the screen, whereas the 1D network takes only the sequence as input.

 ISBN 978-0-9934207-2-6

Table 3: Network Accuracy

Model	Input	Accuracy (%)	# of Parameters
2D	bitmap	98.5	1,663,370
1D	distance	76.52	287,530
1D	angle	93.77	287,530
1D	distance & angle	95.86	287,690

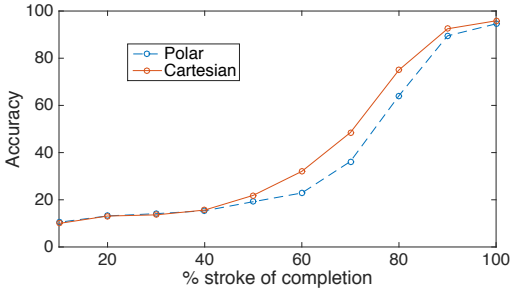


Figure 1: Accuracy vs. stroke completion

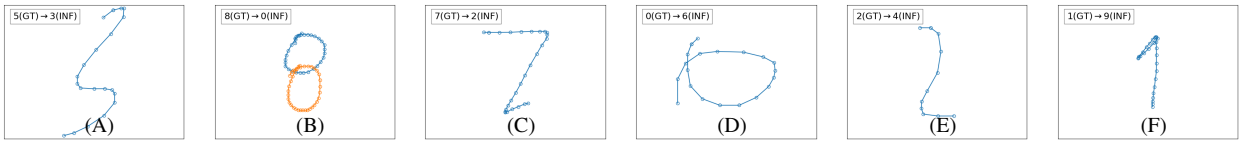


Figure 2: Selection of classification errors. A & B show glyphs where mis-clasification occurs due to omission of subsequent strokes. C & D are ambiguous glyphs. E & F show mis-classification due to glyph formation.

5 Conclusions and Future Work

A dataset was created consisting of Arabic numerals recorded on a smartphone touchscreen using single finger or thumb gestures. Two deep neural networks were trained to recognise the digits. Both models achieved high accuracy. One of the models used a novel polar vector data format and had a significantly lower footprint. In future work, we plan to enhance the accuracy of early digit recognition to accelerate the digit entry process. It is hoped that the open source dataset described here will facilitate further work on this topic. The dataset is available at [Corr et al., 2017].

References

[Bahlmann, 2006] Bahlmann, C. (2006). Directional features in online handwriting recognition. *Pattern Recognition*, 39(1):115 – 125.

[Bengio et al., 1995] Bengio, Y., LeCun, Y., Nohl, C., and Burges, C. (1995). LeRec: A NN/HMM hybrid for on-line handwriting recognition. *Neural Computation*, 7(6):1289–1303.

[Corr et al., 2017] Corr, P., Silvestre, G., and Bleakley, C. (2017). Numeral gesture dataset. https://github.com/PhilipCorr/numeral-gesture-dataset. Accessed: 2017-07-14.

[Guyon et al., 1991] Guyon, I., Albrecht, P., Le Cun, Y., Denker, J., and Hubbard, W. (1991). Design of a neural network character recognizer for a touch terminal. *Pattern Recognition*, 24(2):105–119.

[Kienzle and Hinckley, 2013] Kienzle, W. and Hinckley, K. (2013). Writing handwritten messages on a small touchscreen. In *Proc. Int. Conf. HCI with Mobile Devices and Services*, pages 179–182.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

[LeCun et al., 1998] LeCun, Y., Cortes, C., and Burges, C. J. (1998). MNIST handwritten digit database. `http://yann.lecun.com/exdb/mnist/`. Accessed: 2017-06-22.

[Verma et al., 2004] Verma, B. et al. (2004). A feature extraction technique for online handwriting recognition. In *Proc. IEEE Int. Joint Conf. on Neural Networks*, volume 2, pages 1337–1341.

 ISBN 978-0-9934207-2-6

Facial Image Aesthetics Prediction with Visual and Deep CNN Features

Mohamed Selim[1], Tewodros Amberbir Habtegebrial[1], and Didier Stricker[1,2]

[1]*Technical University of Kaiserslautern*
[2]*Augmented Vision, German Research Center for Artificial Intelligence (DFKI)*
Kaiserslautern, Germany
mohamed.selim, tewodros_amberbir.habtegebrial, didier.stricker@dfki.uni-kl.de

Abstract

Large number of images that has persons are being uploaded to the Internet, at a very high rate. However, they vary in quality and aesthetics. These variations affect the performance of the facial images analysis algorithms. This fact poses an interesting question: *Can we predict the aesthetics of the facial image in stills?*. In this work, we introduce a framework that uses deep face representations from CNNs and other visual features to tackle the problem. We evaluated our algorithms on large scale datasets of persons. Regarding the aesthetics, we used collected portraits from the AVA dataset, as well as the Selfie dataset. We thoroughly evaluated our algorithm. Moreover, we outperformed the state-of-the-art in aesthetic prediction in portrait images as we achieved accuracy of 84% while the state-of-the-art achieved 64.25% by using deep representations from our AestheticsNet combined with visual features.

1 Introduction

In this paper we study aesthetics problem of facial images. Aesthetics prediction has been assessed by computing average aesthetic scores given to images by human annotators directly or indirectly. In some datasets like [Redi et al., 2015], aesthetics scores are given by users. In some cases like [Kalayeh et al., 2015], the scores are inferred indirectly from other information like number of views of the image on a social network. When it comes to the computational modelling of aesthetics of images, aesthetics could be predicted through visual features and from other various cues. Especially on the web, images contain additional information in the form of user comments, number of views and likes by other users etc. However, we restricted the scope of our study to predicting aesthetics only from visual cues.

Selfies have become a common phenomenon. Google reported (http://goo.gl/53ZOjG by 2014) more than 93 million self portraits were being taken everyday on android devices. As of April 2016, a keyword search with "#selfie" retrieves 286,297,630 results on Instagram. As a study performed on the popularity of images on social media showed, images with faces are more likely to get comments and likes [Bakhshi et al., 2014]. In our study *"Aesthetics"* of portraits, we used different tools and mechanisms borrowed from computer vision and machine learning. Our image quality prediction algorithm is based on Convolutional Neural Networks (CNNs); for the purpose of image aesthetics prediction we used CNNs and other computer vision features like GIST, HOG and LBPs. We applied these tools on different Datasets of portrait images [Redi et al., 2015], and a collection of selfies [Kalayeh et al., 2015].

2 Related Work

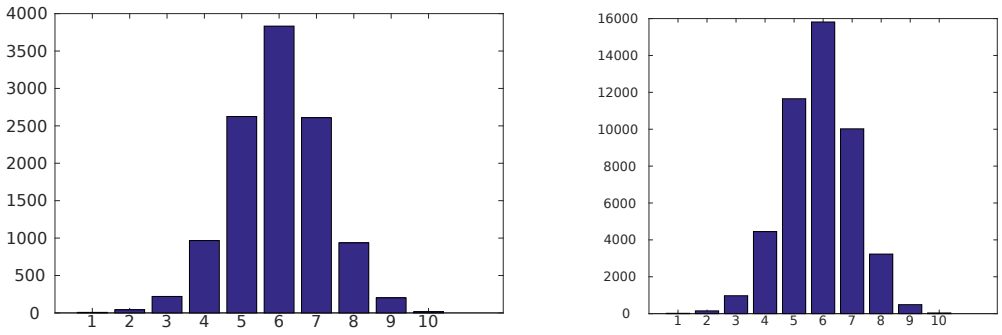
In recent years, a significant amount of work has been devoted to the problem of image aesthetics prediction [Lienhard et al., 2015, Kang et al., 2014]. The works done in the area could be separated into two major

 ISBN 978-0-9934207-2-6

groups: *Feature-based* and *Learning-Based*. Feature-based approaches represent methods that predict image aesthetics from low-level visual features and high-level attributes [Marchesotti et al., 2013]. [Dhar et al., 2011] used high-level describable attributes such as presence of people and portrait depiction, object and scene type, etc. However, deep CNNs[Lu et al., 2014] were also used for aesthetics prediction. Learning-based approaches require no hand-crafted features and they are also shown to be more robust. Recently, researchers are focusing on effective aesthetic analysis of facial images. [Redi et al., 2015] studied factors contributing to the aesthetic beauty of an image by studying portraits collected from AVA dataset. Their work uses various features to describe photo composition, quality, emotions, etc. Moreover, due to the recent "Selfie" phenomenon, researchers are studying facial images uploaded to the Internet. A study showed that images with faces get more likes and comments on social networks [Bakhshi et al., 2014]. [Kalayeh et al., 2015] initiated the study on selfies by creating a new selfie Dataset to open the door for more in-depth studies on selfies.

3 Proposed Approach

Datasets For the purpose of aesthetics prediction, we used a newly introduced Dataset of portraits from [Redi et al., 2015]. The portrait dataset contains 11,400 images. Every image in the Dataset has an aesthetics rating value ranging from 0 to 10. These ratings are average ratings given by users of the DPChalenge website. Figure 3 shows the distribution of ratings of the images. We modelled this problem as binary classification problem where labelling all images whose average score was above 5.55 as "good" and those with score below 5.55 as "bad" similarly to[Redi et al., 2015]. As suggested in [Murray et al., 2012, Redi et al., 2015], we also introduced a margin parameter, denoted by parameter δ , for discarding ambiguous images. We discarded all images within 5.55 $\pm\delta$. In our experiments we tested with values of delta 0.1 and 1. The Selfie Dataset [Kalayeh et al., 2015] contains more than 46,000 selfies collected from *selfeed.com*. The popularity score of the selfies was calculated as log_2 *normalized* views counts of the selfies.



(a) (b)

Figure 1: Score Distribution of (a) Portraits Dataset [Redi et al., 2015], and (b) Selfies Dataset [Kalayeh et al., 2015].

Aesthetics Prediction using Visual Features We analyzed the effectiveness of many computer-vision features in the task of aesthetics evaluation in the context of Portrait images. For this purpose, we extracted features by using GIST (global feature vector for an image by applying oriented Gabor filters at multiple scales), HOG and LBP.

Aesthetics Prediction using CNNs We used CNNs as direct aesthetics prediction by training them on Aesthetics datasets and methods to extract features which are useful for aesthetics. For extracting high level information, we used neural Networks trained on classifying *Image Style* and detecting *Adjective-Noun-Pairs*.

We used different CNN architectures in our work. We used StyleCNN which was introduced to classify image style on Flickr [Karayev et al., 2013]. We also tested Deep SentiBank [Borth et al., 2013] which showed effectiveness in Adjective-Noun-Pairs in visual sentiment detection. We trained AestheticsNet which is based on AlexNet [Krizhevsky et al., 2012] after changing last layer to predict image aesthetics.

4 Evaluation and Results

We conducted experiments on two different tasks: *Aesthetics of Portraits*, and *Popularity of selfies*. In case we fuse features together, we normalize the feature vectors using sigma normalization (early normalization).

Aesthetics Prediction In Table 1 we present our best combination of features, which clearly shows better results than the results of [Redi et al., 2015]. With the exception of the CNN, our tests on the Portraits Dataset [Redi et al., 2015] are done in a 10-fold-cross validation manner. Due to the high computational cost of the CNNs, we experimented by splitting (randomly) the dataset in to 8000 training and 3400 test images.

Delta	SentiBank + Flickr Styles	SentiBank + AestheticsNet	Redi et. al. [Redi et al., 2015]
0.1	66.65	67.32	64.25
1.0	**80.14**	**84.00**	64.25

Table 1: Results on combining features for portrait aesthetic classification. We outperform SOA by $\tilde{2}0\%$.

Selfie Popularity Prediction We have used our computer vision features and CNN features in Selfie Popularity prediction task also. As shown in Table 2 the combination of CNN, HOG and GIST gives a 0.59 Spearman's rank correlation, which is better than the 0.55 state-of-the-art (SOA) correlation, reported by Kalayeh et. al. [Kalayeh et al., 2015].

LBP	CNN	Style	SB	HOG	GIST	CNN-HOG-GIST	Kalayeh et. al. [Kalayeh et al., 2015]
0.11	0.45	0.36	0.45	0.49	0.52	**0.59**	0.55

Table 2: Spearman's rank correlation for popularity prediction on Selfies [Kalayeh et al., 2015]. We outperform the SOA.

In the selfie popularity prediction, the GIST feature was very efficient, achieving a 0.52 correlation level. In Figure 2, we show the effectiveness of our algorithms by presenting sample images classified as bad or good. In Figure 2, we show the top images 64 and bottom 64 images in the prediction by the CNN and the ground truth from the selfie dataset. The CNN was able to differentiate between good and bad selfies according to their popularity scores.

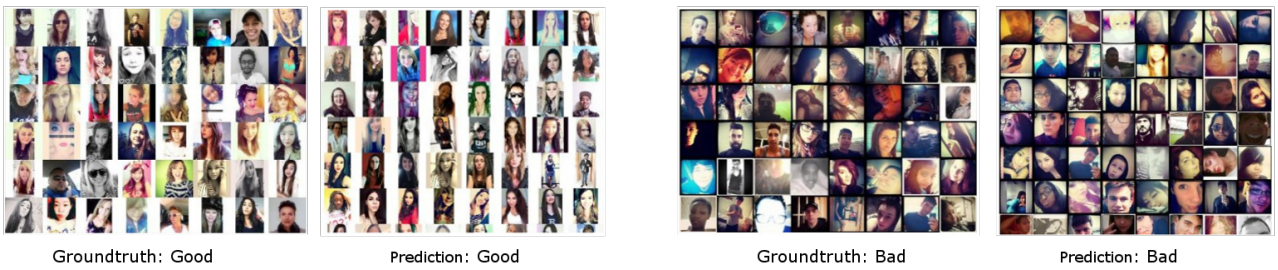


Figure 2: Sample results from our CNN trained on the Selfies Dataset[Kalayeh et al., 2015]. We show visually that the ground truth and prediction are similar

 ISBN 978-0-9934207-2-6

5 Conclusion

In this paper we propose an approach that combines computer vision features and Convolutional Neural Networks in Facial Image Aesthetics prediction. Our results indicate that using CNNs for learning to detect high level features like computer vision and Image Style can significantly improve the classification accuracy of an aesthetics classifier. In predicting Portraits' aesthetics and selfies' popularity, DeepSentibank's ANPs were better than a CNN trained on the dataset. These results indicate that using a CNN trained on a relevant problem can easily solve other problems. Computer vision features like HOG and GIST were surprisingly effective in predicting popularity of selfies. Thus, by combining various features we created an aesthetics prediction algorithm which outperforms the SOA, as we reach classification accuracy of 84% while the SOA reaches only 64.25%. The results we achieved so far, opens the door for investigating our proposed approach on videos datasets and tackle visual challenges existing in videos compared to still images.

Acknowledgments. This work has been partially funded by the University project Zentrums für Nutzfahrzeugtechnologie (ZNT).

References

[Bakhshi et al., 2014] Bakhshi, S., Shamma, D. A., and Gilbert, E. (2014). Faces engage us: Photos with faces attract more likes and comments on instagram. In *Proceedings of the SIGCHI*.

[Borth et al., 2013] Borth, D., Ji, R., Chen, T., Breuel, T., and Chang, S.-F. (2013). Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *21st ACM international conference on Multimedia*.

[Dhar et al., 2011] Dhar, S., Ordonez, V., and Berg, T. L. (2011). High level describable attributes for predicting aesthetics and interestingness. In *CVPR*.

[Kalayeh et al., 2015] Kalayeh, M. M., Seifu, M., LaLanne, W., and Shah, M. (2015). How to take a good selfie? In *Proceedings of the 23rd Annual ACM Conference on Multimedia Conference*.

[Kang et al., 2014] Kang, L., Ye, P., Li, Y., and Doermann, D. (2014). Convolutional neural networks for no-reference image quality assessment. In *CVPR*.

[Karayev et al., 2013] Karayev, S., Trentacoste, M., Han, H., Agarwala, A., Darrell, T., Hertzmann, A., and Winnemoeller, H. (2013). Recognizing image style. *arXiv preprint arXiv:1311.3715*.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *NIPS*.

[Lienhard et al., 2015] Lienhard, A., Ladret, P., and Caplier, A. (2015). Low level features for quality assessment of facial images. In *10th Int. Conf. on computer Vision Theory and Applications, VISAPP*.

[Lu et al., 2014] Lu, X., Lin, Z., Jin, H., Yang, J., and Wang, J. Z. (2014). Rapid: Rating pictorial aesthetics using deep learning. In *Proceedings of the ACM International Conference on Multimedia*, pages 457–466.

[Marchesotti et al., 2013] Marchesotti, L., Perronnin, F., and Meylan, F. (2013). Learning beautiful (and ugly) attributes. In *BMVC*, volume 7, pages 1–11.

[Murray et al., 2012] Murray, N., Marchesotti, L., and Perronnin, F. (2012). Ava: A large-scale database for aesthetic visual analysis. In *CVPR*.

[Redi et al., 2015] Redi, M., Rasiwasia, N., Aggarwal, G., and Jaimes, A. (2015). The beauty of capturing faces: Rating the quality of digital portraits. In *Automatic Face and Gesture Recognition (FG)*.

Correlation of Pre-Operative Cancer Imaging Techniques with Post-Operative Gross and Microscopic Pathology Images

Gabriel Reines March, Xiangyang Ju, Stephen Marshall

Medical Devices Unit, West Glasgow Ambulatory Care Hospital, NHS Greater Glasgow and Clyde, Dalnair St, Glasgow G3 8SJ (UK)
Hyperspectral Imaging Centre, Dept. of Electronic and Electrical Engineering, University of Strathclyde, 204 George St, Glasgow G1 1XW (UK)

Abstract

In this paper, different algorithms for volume reconstruction from tomographic cross-sectional pathology slices are described and tested. A tissue-mimicking phantom made with a mixture of agar and aluminium oxide was sliced at different thickness as per pathological standard guidelines. Phantom model was also virtually sliced and reconstructed in software. Results showed that shape-based spline interpolation method was the most precise, but generated a volume underestimation of 0.5%.

Keywords: Tomographic Image Processing, Volume Reconstruction, Biomedical Image Processing

1 Introduction

Medical imaging techniques have evolved dramatically over the last two decades. The increase in acquisition and processing speeds, resolution and availability has fostered their widespread use as a diagnostic and assessment tool. However, in cancerous tissue, cellular-level features play a fundamental role in determining the stage and type of carcinoma, as well as its local metabolic behaviour. To date, the gold standard for analysing tissue at these high resolutions is by taking a sample (biopsy) and analysing it under a microscope, which is a highly invasive procedure. The current study aims to retrospectively bridge the gap between post-operative pathology and diagnostic imaging, by fusing pathological findings to pre-operative PET-CT scans. Upon alignment of both datasets, clinical oncologists will be able to identify and match certain pathological features in the pre-operative images. This new insight could potentially help them to better predict cancer staging and prognosis from diagnostic images, therefore avoiding unnecessary surgery.

The challenges of this study are the disparity in image resolution between modalities, low sampling rate in the transverse plane and the non-linear deformations undergone by lung tissue during the different clinical stages. To date, simulations with a synthetic tissue-mimicking phantom have been used to test several volume reconstruction algorithms and methods.

2 Materials and Methods

Several synthetic tissue-mimicking phantoms have been used to test the performance of the different algorithms implemented in this project. Its shape was especially designed in CAD software to include several features useful for testing and challenging different volume reconstruction algorithms (see Figure 1). Materials tested include poly(vinyl alcohol) and agar gelatine, the latter showing the most appropriate mechanical properties for our purpose. Aluminium oxide is added as scattering agent to make the material opaque in the x-ray spectrum.

 ISBN 978-0-9934207-2-6

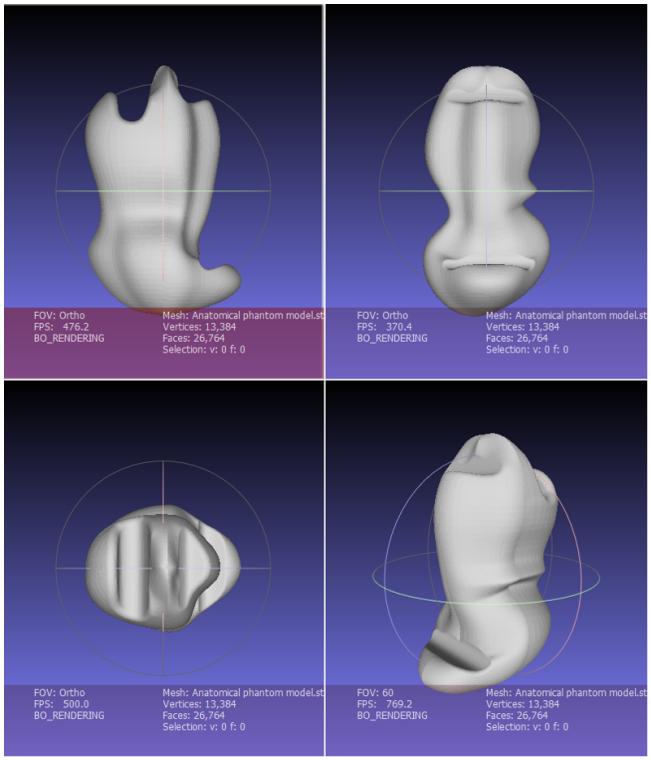


Figure 1: Original phantom design

A custom-designed mould has been 3D-printed to produce the phantom samples. Figure 2 shows a sample cross-section of the phantom in both x-ray and visible spectra.

One of the main challenges of this project is to truthfully reconstruct the original volume and shape of the tumour from tomographic slices. The main reason behind this complexity is because lung carcinomas can adopt highly unpredictable, heterogeneous shapes, which involve fast-changing features along its surface. Combined with the fact that lung tissue has a spongy texture and therefore thin slices cannot be produced without tearing the sample, reconstructing the specimen volume from discrete pathology slices remains an ill-posed problem, as per basic sampling theorem directives. One of the motivations for using a phantom has been precisely the availability of a ground truth model to compare our reconstruction against.

Shape-based interpolation method has been used, first developed by Raya and Udupa [1]. It consists of first segmenting the image to generate a binary mask of the slice, where pixels belonging to the shape of interest are represented by boolean true values. Next, the binary image is converted into a grayscale image, wherein the grey value of a pixel represents the shortest Manhattan distance to the cross-sectional boundary of the binary mask. As a convention, points inside the boundary (i.e. belonging to the shape) are assigned positive distances, and negative values are assigned to outsiders. Those grey values are then interpolated along the z-axis. The non-negative values of the resulting volume constitute the interpolated object.

In our first experiments, several agar phantoms were embedded in a bespoke slicing rig and slices were made at regular intervals (2.5 mm). With the use of a digital camera (Canon EOS M3, Canon Inc., Tokyo, Japan), pictures of each cross-section were taken (see example in Figure 2 right). Next, all photographs were processed to segment the phantom region of interest. Two segmentation approaches were tried: colour thresholding and region growing. The former is based on absolute pixel colour values, whereas the latter depends on the gradient of the scene. In our self-implemented region growing algorithm, the user first needs to interactively select a colour plane, preferably one offering high contrast between the phantom and the surroundings. Three colour spaces are available to choose from, namely RGB, LAB and HSV. Then the user selects a series of seed pixels on the image, which can be any pixel inside the region of interest. If the grey level difference between a

 ISBN 978-0-9934207-2-6

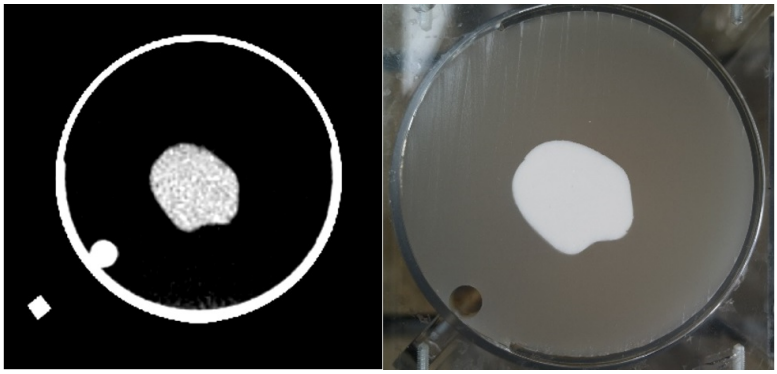


Figure 2: Sample phantom cross-section in x-ray (left) and visible spectra (right)

pixel and a neighbour is less than a given threshold, then the neighbour is included in the region of interest. Otherwise, the neighbour is defined as an outsider. The algorithm iterates until convergence is reached.

3 Results

Initially, several agar phantoms were physically sliced at $2.5mm$ intervals. A picture of a fiducial marker of known size was also taken to establish a relationship between pixel size and real world units. The phantom area was segmented using the algorithms mentioned above. The volume recovered on all experiments laid between 92% and 93% of the ground truth value (measured to be $48,864mm^3$).

At this point the question was whether this underestimation was intrinsic to the interpolation method, or due to imprecisions in the segmentation routine, or a combination of both. Therefore, in order to isolate the interpolation problem, it was proposed to perform a virtual slicing of the original CAD volume and try to reconstruct it from the subset of slices. This way the binary masks were intrinsically defined by the volume cross-sections, avoiding the need to segment the images.

When designing the experiment, however, another question arose: what position should we start slicing our model at? The obvious answer seemed to be at the apex. Whilst this is possible with a virtual model, on an embedded phantom or real carcinoma it is very difficult to perform a cut which corresponds to the plane where the very first distinguishable feature situated on the apex shows. Therefore, this randomness in the position of the first cut had to be accounted for. It was assumed that the first cut could lie in between 0 (i.e. the apex of the object) and one slice thickness, with a uniform probability.

The virtual model was sliced at intervals ranging from 0.5 to $10mm$ in $0.5mm$ steps. For each slice thickness and interpolation method (i.e. nearest neighbour, linear and cubic spline), the routine was run 200 times, introducing a random offset at each iteration. Results are shown in Figure 3. It can be observed that in the region of interest (i.e. slices from $1mm$ up to $5mm$ thickness), nearest neighbour provides the most accurate results, and cubic spline shows a systematic volume underestimation of 0.5%. In order to verify these results, the original phantom CAD model was divided in quarters, and each new shape was reconstructed independently using the same conditions. Results obtained were coherent with those shown in Figure 3.

It was also proposed to perform a local comparison between the ground truth model and reconstructed phantom to evaluate which shape features give the most error. For this purpose, both objects are first converted to point clouds. Then, the reconstructed volume is rigidly registered to the model using Iterative Closest Point (ICP) algorithm. Next, for each point belonging to the generated volume, the distance to its nearest neighbour in the model point cloud is calculated. These distances are then represented in a 3D map, which allows the user to perform a quick qualitative evaluation of the shape reconstructed. After running the routine on the volumes obtained, it was seen that the features giving the largest error were mainly the non-reconstructed shallow valleys and the extrema. These results will be taken into account when designing other phantoms to test our algorithms.

 ISBN 978-0-9934207-2-6

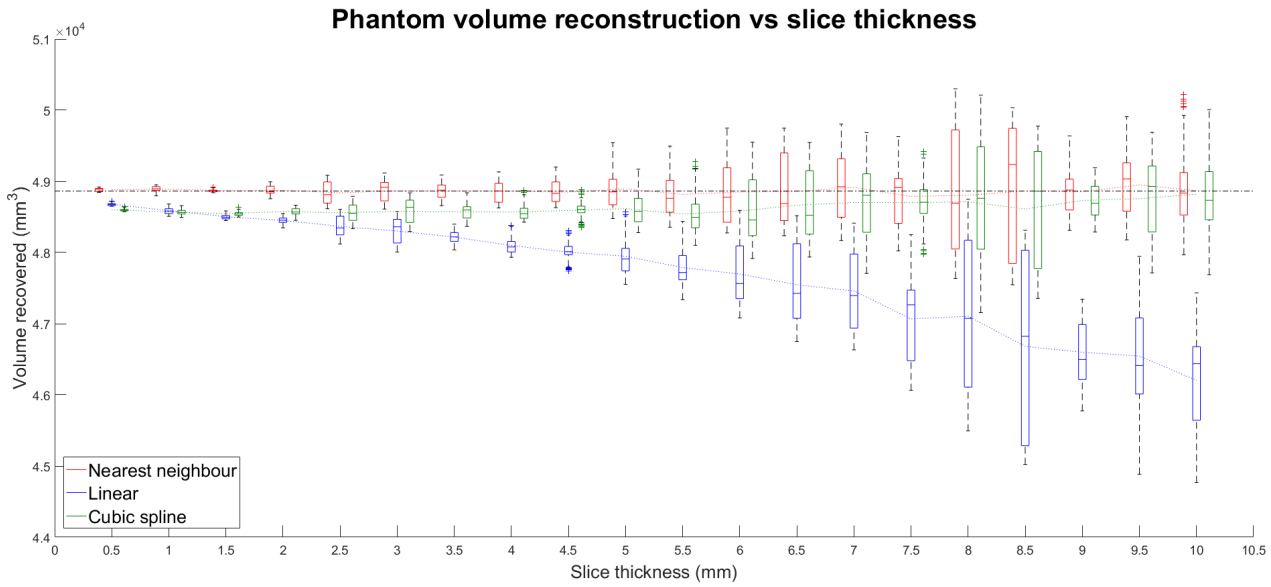


Figure 3: Boxplot showing the volume reconstructed after 200 runs using nearest neighbour (red), linear (blue) and cubic spline (green) interpolation algorithms. Boxes indicate the 25 and 75 percentile of the distribution, horizontal marker indicates the median, and whiskers show the spread of data. Outliers are marked with a cross symbol. Dotted lines represent the evolution of the mean reconstructed value. Ground truth ($48,864 mm^3$) is indicated with a dash-dotted black line.

4 Future work

Future work will be focused on three main areas. **Morphology-based interpolation**: the method proposed in [2, 3] is currently being implemented. It is then going to be tested using the same dataset as used in the shape-based interpolation experiment, so that results can be compared. This method is expected to show better performance around the extrema regions and better handling of topological changes. **Comparison of our algorithms vs commercial software**: the phantom will be scanned in a Computerised Tomography (CT) scanner and its boundary will be delineated by an oncologist on a commercial radiotherapy planning unit. Resulting volume will be compared against ground truth and our implementation. **Trial on human tissue**: volume reconstruction algorithms will be tested on actual lung tissue. Hyperspectral imaging will be tested to explore whether cancerous tissue presents a different absorption spectrum in the near IR region compared to healthy tissue. Automatic tumour segmentation approaches will also be investigated in both gross and pathology specimens.

References

[1] S. P. Raya and J. K. Udupa, *Shape-Based Interpolation of Multidimensional Objects*, IEEE T Med Imaging **9**(1), 32-42 (1990)

[2] A. B. Albu, T. Beugeling and D. Laurendau, *A Morphology-Based Approach for Interslice Interpolation of Anatomical Slices From Volumetric Images*, IEEE T Bio-Med Eng, **55**(8), 2022-2038 (2008)

[3] J. Vidal, J. Crespo, V. Maojo, "Recursive Interpolation Technique for Binary Images Based on Morphological Median Sets", in *Mathematical Morphology: 40 Years On*, C. Ronse *et al.*, eds. (Springer, 2005)

Evaluating Quantized Convolutional Neural Networks for Embedded Systems

Simon O'Keeffe, Rudi Villing

Department of Electronic Engineering, Maynooth University-National University of Ireland Maynooth, Maynooth, Co. Kildare, Ireland

Abstract

This paper presents a deep learning approach which evaluates accuracy and inference time speedups in deep convolutional neural networks under various network quantizations. Quantized networks can result in much faster inference time allowing them to be deployed in real time on an embedded system such as a robot. We evaluate networks with activations quantized to 1, 2, 4, and 8-bits and binary weights. We found that network quantization can yield a significant speedup for a small drop in classification accuracy. Specifically, modifying one of our networks to use an 8-bit quantized input layer and 2-bit activations in hidden layers, we calculate a theoretical 9.9× speedup in exchange for an F_1 score decrease of just 3.4% relative to a full precision implementation. Higher speedups are obtainable by designing a network architecture containing a smaller proportion of the total multiplications within the input layer.

Keywords: Convolutional Neural Networks, Deep Learning, Network Quantization

1 Introduction

Deep Convolutional Neural Networks (CNNs) are recognized as the state of the art for image classification [Krizhevsky et al., 2012]. However, Deep CNNs are too computationally intensive to run on low power embedded devices such as the NAO robot. Quantization of the weights and activations can reduce the computational requirements of Deep CNNs and speed up the inference times. Our previous work [O'Keeffe and Villing, 2017] evaluated various network architectures for ball detection in robot soccer on the Softbank NAO robot using both full precision and binarized networks. Our contribution in this paper is to evaluate the performance of networks with quantized activations and weights binarized to +1 or -1. We also examine the maximum theoretical speedup obtainable for these networks.

2 Related Work

Most of the time taken to perform inference in Deep CNNs results from the multiplication of real-valued weights by real-valued activation values. High precision parameters have been shown to be unimportant to the performance of deep networks [Gong et al., 2014]. As such, network quantization has been proposed to improve the computational efficiency of the network, particularly at inference time. BinaryConnect [Courbariaux et al., 2015] trains a DNN with binary weights during forward and backward propagation, but retains the precision of the stored weights in which gradients are accumulated. The authors found that BinaryConnect acted as a regularizer and obtained near state-of-the-art results on MNIST, CIFAR-10, and SVHN datasets. BinaryNet [Courbariaux and Bengio, 2016] was proposed as an extension of BinaryConnect where both weights and activations are constrained to +1 or -1. If all operands of the convolution are binary, then the convolutions can be calculated by XNOR and bit counting operations. XNOR-Net [Rastegari et al., 2016] is another method that binarizes the weights and activations in a network but uses a different binarization method and network

 ISBN 978-0-9934207-2-6

structure than BinaryNet. BinaryNet achieved a top-1 accuracy 27.9% while XNOR-Net achieved 44.2% on the ImageNet 1000 classification task. Both BinaryNet and XNOR-Net are binarized versions of AlexNet [Krizhevsky et al., 2012] which has a 56.6% top-1 full precision accuracy.

In an effort to increase the top-1 accuarcy towards that of a full precision network, several researchers have looked towards using low bitwidth quantization of the acitivations with binary weights. DoReFa-Net [Zhou et al., 2016] trained a selection of networks with low bitwidth activation including a network with 2-bit activations and binary weights achieving a top-1 score of 50.7%. Quantized Neural Networks [Hubara et al., 2016] extended the work on BinaryNet and achieved a top-1 score of 51.03% with 2-bit activations and binary weights. In this paper we examine how the activation bitwidth affects accuracy for our ball detection task and explore the maximum theoretical speedup attainable for various activation bitwidths.

3 Approach and Experiment

Binarizing the weights and activations of a network allows for full precision convolutions to be approximated by XNOR and popcount operations. This can be extended to B-bit quantization with binary weights which increases the amount of time needed to run the XNOR and popcount operation by B. Equation (1) shows the convolution operation with B bit activations and binary weights.

$$s = \sum_{b=1}^{B} 2^{b-1} \left(x^b \cdot w \right) \tag{1}$$

where $x = \{x_1, x_2, \cdots, x_N\}$ is a vector of B bit inputs, x_n^b is the b^{th} significant bit of the n^{th} input, w is a vector of 1-bit weights, and s is the resulting weighted sum. Here $\cdot$ indicated the XNOR and popcount operation. The networks were quantized to k-bits according to Equation (2) where $\lfloor \cdot \rceil$ indicates the rounding operation and $x \in [-1, 1]$

$$q_k(x) = 2 \left(\frac{\left\lfloor \left(2^k - 1\right) \left(\frac{x+1}{2}\right) \right\rceil}{2^k - 1} - \frac{1}{2} \right) \tag{2}$$

3.1 Maximum Theoretical Speedup

A typical convolutional kernel consists of cN_wN_o multiplications where c is the number of channels, N_w is the number of weights in the kernel, and N_o is the number of outputs resulting from the convolution. When weights are binarized and activations are quantized then a convolutional kernel consists of cN_wN_oB binary operations and N_ob popcount operations, where B is the activation bitwidth. Therefore the maximum theoretical speedup for a convolution relative to full precision on a CPU only can be computed by

$$s = \frac{cN_wN_o}{\frac{1}{R}cN_wN_oB + pN_oB} = \frac{RcN_w}{cN_wb + RpB} \tag{3}$$

where R is the number of binary operations that can be performed in one cycle and p is the number of cycles needed to perform a popcount. Single Instruction Multiple Data (SIMD) operations perform the same operation on multiple data points simultaneously. The Softbank NAO robot has an Intel Atom processor which supports the Streaming SIMD Extensions (SSE) instruction set. Using SSE instructions 128 binary operations ($R = 128$) can be performed in one cycle. The popcount operation can be completed in one cycle on CPUs that support SSE4 instructions, however the NAO robot's processor only supports up to SSSE3. Therefore we assume the popcount operation then takes 4 cycles to perform on average. Equation (3) does not take into account the cost of loading up the SSE registers or storing the result.

Using binary weights eliminates all multiplication operations and reduces convolutions to only additions and subtractions. This can be performed by either using Equation (1) or by packing 8-bit quantized values into 16-bit integers (to prevent overflow) in an SSE register and performing 8 addition operations in parallel. This offers a maximum theoretical speedup of 8×.

 ISBN 978-0-9934207-2-6

3.2 Experiment

For our experiment we trained two different network architectures to detect black and white balls in robot soccer environments in the presence of distractors such as field lines and robots. We trained on the Caffe framework [Jia et al., 2014] using a dataset that consists of 16133 20×20 image patches. The full details of the dataset creation can be found at [O'Keeffe and Villing, 2017]. Network 1 consisted of 2 convolution layers with 3×3 kernels and 2 fully connected layers and network 2 consists of 5 convolutional layers using both 3×3 and 1×1 kernels. The networks were trained under two different convolutional block layouts. The first version used the convolutional block outlined by XNOR-Net [Rastegari et al., 2016] and consisted of Batch Normalization, activation, convolution, and pooling layers. The second version followed the typical block in a CNN and consisted of convolution, Batch Normalization, activation, and pooling. Each network was trained in full precision along with networks with 1, 2, 4, and 8-bit quantizations of the activation. Each network was trained for 10,000 iterations.

4 Results

The classification accuracy of the networks can be seen in Figure 1(a). The classification accuracy is given by the F_1 score which is a balanced score between the precision and recall of a given network. We can see that the classification accuracy performs poorly for 1-bit activations, with a mean F_1 score of 85.1%. Moving to 2-bit activations increases the mean F_1 score to 93.6% before increasing to 95.5% and 96.26% for 4 and 8-bit activations respectively.

Since the theoretical speedup only depends on the kernel size and the number of channels, an input image with one channel yields a relatively small speedup as seen in Figure 1(b). The internal layers however are better suited to a speedup from quantization. For example, network 1 with 2-bit activatons yields a 12× speedup. However, internal layers in networks with 4-bit or 8-bit activations do not attain a speedup over SSE-enabled 8-bit integer additions with binary weights.

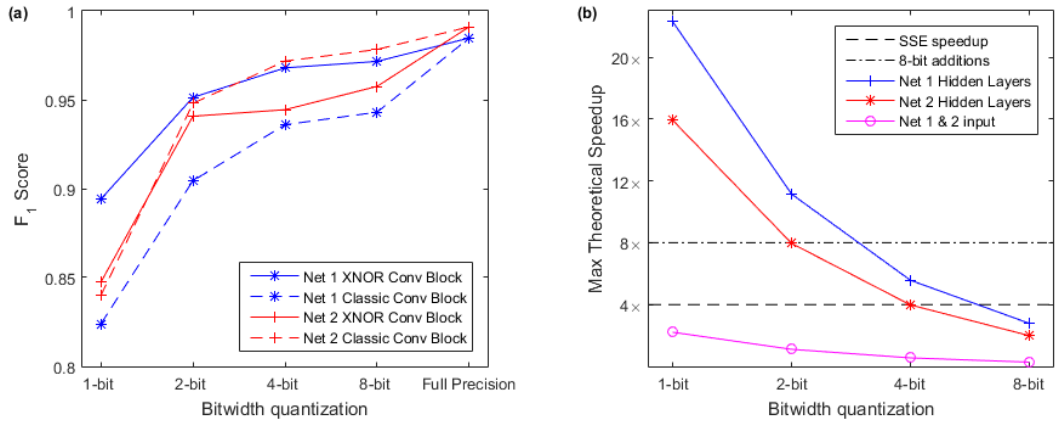


Figure 1: This figure shows (a) the F_1 score classification against the activation bitwidth for both networks and convolution block layouts and (b) the maximum theoretical speedup using XNOR and popcount operations for activation bitwidth

5 Discussion and Conclusion

This evaluation shows that network quantization can yield a significant speedup for a small drop in classification accuracy. For example network 1 with binarized weights and 2-bit activations decreases 3.4% in F_1 score relative to full precision with a potential 12× speedup for the hidden layers. However the theoretical speedup does not hold for the whole network and varies depending on the layer architecture within a network. Layers

 ISBN 978-0-9934207-2-6

with a smaller number of channels, such as input layers, do not yield much speedup through XNOR and popcount operations. In these cases full precision convolutions can be implemented using SSE to deliver a 4× speedup relative to CPU only. Alternatively, in exchange for a minor decrease in classification accuracy, an even better speedup of 8× is possible using quantized 8-bit values and the addition technique for convolutions described in Section 3.1.

Convolution speedup in the input layer is not as important an issue for large Deep CNNs such as AlexNet where the input layer has far fewer channels (e.g. 3) than the internal representation (e.g. 512) meaning that the input layer contains only 14.9% of the multiplications for the network. However the ball detection networks evaluated in this work do not use such Deep CNNs, as such the input layer contains 33.4% of the total multiplications. For our networks, we can achieve a 9.9× maximum theoretical speedup for the whole network by using the addition technique for the input layer and 2-bit activations for the hidden layers. Higher speedups can be obtained by designing a network architecture that contains a smaller proportion of the total multiplications within the input layer.

Acknowledgements

This work was supported by the Irish Research Council under their Government of Ireland Postgraduate Scholarship 2013.

References

[Courbariaux and Bengio, 2016] Courbariaux, M. and Bengio, Y. (2016). BinaryNet: Training Deep Neural Networks with Weights and Activations Constrained to +1 or -1. *arXiv*, page 9.

[Courbariaux et al., 2015] Courbariaux, M., Bengio, Y., and David, J.-P. (2015). BinaryConnect: Training Deep Neural Networks with binary weights during propagations. *Nips*, pages 1–9.

[Gong et al., 2014] Gong, Y., Liu, L., Yang, M., and Bourdev, L. (2014). Compressing Deep Convolutional Networks using Vector Quantization. pages 1–10.

[Hubara et al., 2016] Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., and Bengio, Y. (2016). Quantized Neural Networks: Training Neural Networks with Low Precision Weights and Activations.

[Jia et al., 2014] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional Architecture for Fast Feature Embedding. *Proceedings of the ACM International Conference on Multimedia - MM '14*, pages 675–678.

[Krizhevsky et al., 2012] Krizhevsky, A., Hinton, G. E., and Sutskever, I. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *the Neural Information Processing Systems Foundation 2012 conference*, pages 1097–1105.

[O'Keeffe and Villing, 2017] O'Keeffe, S. and Villing, R. (2017). A Benchmark Data Set and Evaluation of Deep Learning Architectures for Ball Detection in the RoboCup SPL. In *Accepted for publication at RoboCup 2017*.

[Rastegari et al., 2016] Rastegari, M., Ordonez, V., Redmon, J., and Farhadi, A. (2016). XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks. *Eccv*, pages 1–17.

[Zhou et al., 2016] Zhou, S., Ni, Z., Zhou, X., Wen, H., Wu, Y., and Zou, Y. (2016). DoReFa-Net: Training Low Bitwidth Convolutional Neural Networks with Low Bitwidth Gradients. *arXiv*, 1(1):1–14.

Stitching Skin Images of Scars

S. M. Iman Zolanvari & Rozenn Dahyot

School of Computer Science and Statistics
Trinity College Dublin, Ireland

Abstract

This paper introduces an automatic procedure for aligning and stitching the medical images of skin scars that have the various amount of overlapping into one single registered image. The alignment procedure is based on the rigid transformation of the pair of images regarding detected matched features. The proposed paper compares four different feature detection methods and evaluates the methods on several clinical cases. For each case, the initial image is divided into four smaller sub-images with the different dimension. The result shows that the Harris Corner Detector algorithm achieves nearly 99% accurate result with the minimum overlapping of 160 pixels as the fastest method.

Keywords: Registration, Feature Detection, Skin Imaging, Corner Detector, Image Processing

Introduction

IImage registration is vastly used in medical fields (e.g. radiological and microscopic images [Hill et al., 2001, Yankovich et al., 2014]) as the camera field of view is often not large enough to cover the region of interest. We focus here on skin images captured in extreme close-up (the camera is literally touching the skin without pressing against it to not create artificial deformations of the skin on the image border) in small overlapping patches; where the camera emits its lights allowing all recorded images to have the same controlled lighting conditions (cf. Fig. 1). One application of stitching these image patches together is for scar follow-up (e.g. occurring from surgery or an accident) as a cosmetic treatment where the scar needs to be accurately measured over time.

Figure 1: Exemplars for Quantitative Assessments.

This paper introduces an automatic procedure for aligning and stitching the pair of skin scar images that have the various amount of overlapping into one single composed image. Four different feature detection methods are evaluated for the rigid registration to align these images [Fookes and Bennamoun, 2002].

 ISBN 978-0-9934207-2-6

Proposed Pipeline for Registration

The processing pipeline for registration of a pair of images has the following steps:
• Converting RGB images to grayscale;
• Detecting features and their orientation; that is a primary step for the registration procedure as the correspondences should be matched, and the transformation coefficient must be defined [Na et al., 2016]. The combined corner and edge detection method [Harris and Stephens, 1988] detects the feature once two different edge directions of the local neighbourhood are present near the point. Matas et al. [Matas et al., 2004] introduced maximally stable extremal regions technique for feature detection and establishing the correspondences between the pair of images. Speeded-Up Robust Features (SURF) method was proposed by Bay et al. [Bay et al., 2008] to detect interest points using integral images which are scale independent. More recently, Leutenegger et al. [Leutenegger et al., 2011] proposed a method to detect, describe and match the key-points which configurable circular sampling pattern from which computes brightness comparisons to form a binary descriptor string. Therefore, These four feature detection methods are tested in this paper:

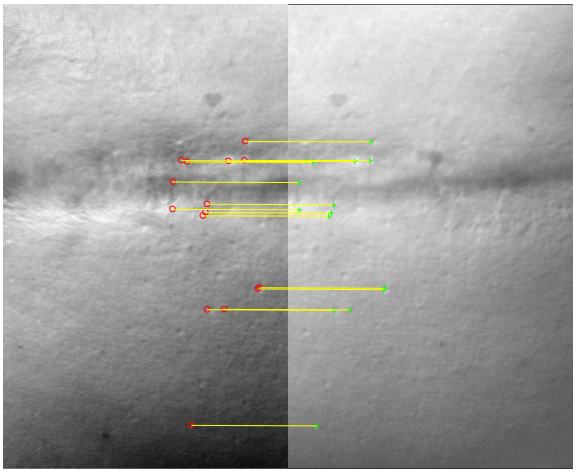


Figure 2: Matched Features for the Smallest Sub-Image of Case 4.

1. Combined Corner and Edge Detector (Harris) [Harris and Stephens, 1988];
2. Maximally Stable Extremal Regions (MSER) [Obdržálek et al., 2009] [Nistér and Stewénius, 2008] [Mikolajczyk et al., 2005];
3. Speeded-Up Robust Features (SURF) [Bay et al., 2008];
4. Binary Robust Invariant Scalable Keypoints (BRISK) [Leutenegger et al., 2011].

• Finding correspondences between detected features in the image pair (Fig. 2);
• Matching features and calculating the rigid transformation between the two images [Lowe, 2004, Muja and Lowe, 2009, Muja and Lowe, 2012];
• Registering the pair of images using the matched transformation;

Experimental Results and Conclusion

Seven cases of skin images of various scars are tested (Fig. 1). There are many input variables involved for the computation time comparison of the four feature detection methods. Therefore, for each method, only the calculation time is considered that could detect the number of 500 to 5000 features. The input variables are chosen regarding this assumption as described below.

Parameter Choices. For each feature detection method, the parameters are selected for having a similar number of detected reliable features (i.e. 500 to 5000) and match to their correspondences. Parameters for the SURF algorithm is assumed with octave greater than 1 (e.g. filter sizes 9×9, 25×25, 21×21, etc.). In MSER algorithm, the intensity threshold levels considered as 0.1 and the pixels below that refer as black and those above or equal as white with maximum area variation between extremal regions of one. Also, the minimum

 ISBN 978-0-9934207-2-6

accepted quality of corners of 3 is assumed for the Harris-Stephens corner detector method. The minimum intensity contrast threshold of the BRISK algorithm is assumed between 0.02 to 0.03.

Experimental Design. Each case is divided into smaller sub-image along the scars direction (e.g. horizontal or vertical) as the images are captured with a different orientation. Also for each pair of patches to stitch, 760, 560, 360 and 160 overlapping pixels are considered. The procedure for all cases is processed on the same machine and the calculation time are only for the feature detection algorithm, excluding the loading data and registration step. Also, for accuracy assessment, the final dimension of the registered patches are considered regarding the initial input.

Registration Results. All methods successfully were registered the patches and archived the accuracy of ±2 pixels. Since, the dimension of each side was 960, then the 2 pixels over the initial dimension generates around 99.79% accuracy rate for the registered result.

Computation Time. In Figure 3, the graph shows relative computation time of the four algorithms for all cases. The processing time is normalised w.r.t. the maximum time that are observed with SURF algorithm in case one. Harris Corner Detector detects the features for all cases at around less than one-fifth time of the SURF algorithm. Therefore, it was the fastest approach among the other three. The average processing time of MSER is almost the second fastest method. Table 1 presents computation time (in absolute term) and the number of detected correspondences. Although all methods may detect a various number of features with different thresholds, the parameters are utilised to have close range of detected features to compare the processing time.

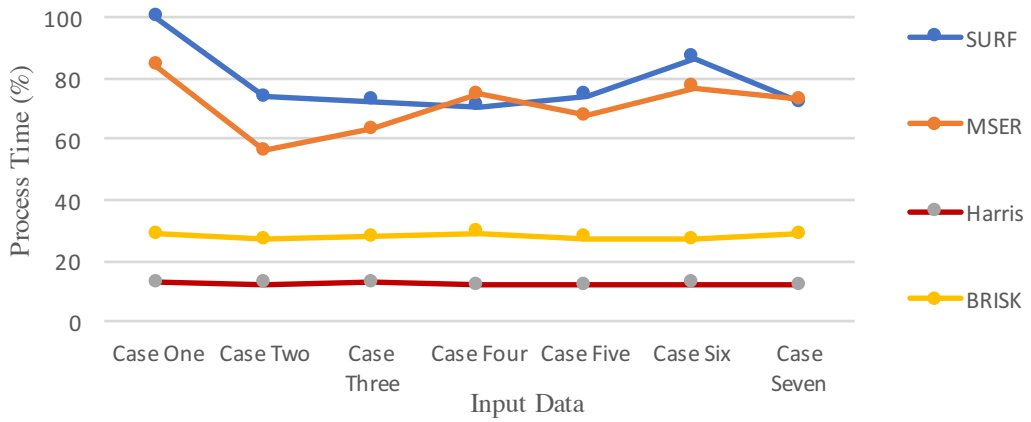


Figure 3: Relative Computation Times for feature detection in a 860 × 960 grey-scale image.

	Processing Time (ms)				Number of Detected Features			
scar #	SURF	MSER	Harris	BRISK	SURF	MSER	Harris	BRISK
1	0.408796	0.118176	0.071697	0.146892	2522	810	3915	2704
2	0.517467	0.393132	0.070115	0.137689	1487	1119	3578	2760
3	0.436677	0.088883	0.056365	0.142646	2430	980	3619	2347
4	0.392414	0.095381	0.085964	0.141218	1940	1672	4725	2913
5	0.406424	0.371482	0.066391	0.149461	3491	2195	3279	2880
6	0.474918	0.420386	0.068236	0.148263	3837	2458	3319	2713
7	0.395713	0.398235	0.065662	0.156705	3390	2298	3029	4291

Table 1: Absolute Computation Time and number of detected features in the feature detection step.

Conclusion. Harris Corner Detector algorithm achieves nearly 99% accurate result with the minimum overlapping of 160 pixels between patches. Harris Corner Detector is also the fastest which can be valuable when

 ISBN 978-0-9934207-2-6

registering multiple patches. Future work will test the algorithm further with skin patches with various skin colours (with and without scars).

Acknowledgements. This research has been co-funded by Enterprise Ireland (Innovation Partnership Project IP-2016-0460), the European Union through the European Regional Development Fund and Ireland's European Structural and Investment Fund Programmes 2014-2020.

References

[Bay et al., 2008] Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359.

[Fookes and Bennamoun, 2002] Fookes, C. B. and Bennamoun, M. (2002). Rigid and non-rigid image registration and its association with mutual information: a review.

[Harris and Stephens, 1988] Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Manchester, UK.

[Hill et al., 2001] Hill, D. L. G., Batchelor, P. G., Holden, M., and Hawkes, D. J. (2001). Medical image registration. *Physics in Medicine & Biology*, 46(3):R1.

[Leutenegger et al., 2011] Leutenegger, S., Chli, M., and Siegwart, R. Y. (2011). Brisk: Binary robust invariant scalable keypoints. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2548–2555. IEEE.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.

[Matas et al., 2004] Matas, J., Chum, O., Urban, M., and Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image and vision computing*, 22(10):761–767.

[Mikolajczyk et al., 2005] Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., and Van Gool, L. (2005). A comparison of affine region detectors. *International journal of computer vision*, 65(1-2):43–72.

[Muja and Lowe, 2009] Muja, M. and Lowe, D. G. (2009). Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*, 2(331-340):2.

[Muja and Lowe, 2012] Muja, M. and Lowe, D. G. (2012). Fast matching of binary features. In *Computer and Robot Vision (CRV), 2012 Ninth Conference on*, pages 404–410. IEEE.

[Na et al., 2016] Na, Y., Liao, M., and Jung, C. (2016). Super-speed up robust features image geometrical registration algorithm. *IET Image Processing*, 10(11):848–864.

[Nistér and Stewénius, 2008] Nistér, D. and Stewénius, H. (2008). Linear time maximally stable extremal regions. *Computer Vision–ECCV 2008*, pages 183–196.

[Obdržálek et al., 2009] Obdržálek, D., Basovník, S., Mach, L., and Mikulík, A. (2009). Detecting scene elements using maximally stable colour regions. In *International Conference on Research and Education in Robotics*, pages 107–115. Springer.

[Yankovich et al., 2014] Yankovich, A. B., Berkels, B., Dahmen, W., Binev, P., Sanchez, S. I., Bradley, S. A., Li, A., Szlufarska, I., and Voyles, P. M. (2014). Picometre-precision analysis of scanning transmission electron microscopy images of platinum nanocatalysts. *Nature Communications*, 5.

Characterisation of CMOS Image Sensor Performance in Low Light Automotive Applications

Shane P. Gilroy[1,2], John O'Dwyer[2] and Lucas C. Bortoleto[2]

[1]*Department of Mechanical and Electronic Engineering, Institute of Technology Sligo, Co. Sligo, Ireland*
[2]*Department of Engineering Technology, Waterford Institute of Technology, Co. Waterford, Ireland*

Abstract

The applications of automotive cameras in Advanced Driver-Assistance Systems (ADAS) are growing rapidly as automotive manufacturers strive to provide 360˚ protection for their customers. Vision systems must capture high quality images in both daytime and night-time scenarios in order to produce the large informational content required for software analysis in applications such as lane departure, pedestrian detection and collision detection. The challenge in producing high quality images in low light scenarios is that the signal to noise ratio is greatly reduced. This can result in noise becoming the dominant factor in a captured image thereby making these safety systems less effective at night. This paper outlines a systematic method for characterisation of state of the art image sensor performance in response to noise, so as to improve the design and performance of automotive cameras in low light scenarios. The experiment outlined in this paper demonstrates how this method can be used to characterise the performance of CMOS image sensors in response to electrical noise on the power lines.

Keywords: Image Sensor Characterisation, ADAS, Electrical Noise, Low light, CMOS

1 Introduction

There were 472 road deaths in Ireland in 1997 according to the Road Safety Authority [RSA, 2015]. There were 166 road deaths in 2015 and this number of fatalities has decreased in an almost linear fashion in the intervening years as shown in Figure 1. This reduction is due in part to the advancement and standardisation of automotive safety systems. Automotive safety systems fall into two main categories, passive safety systems which protect the driver and passengers in the event of a collision such as seat belts, airbags etc. and active safety systems which prevent the occurrence of a collision such as traction control, ABS Brakes and blind spot monitoring. Intelligent vision systems are quickly becoming a larger component of active automotive safety systems for a wide range of applications as manufacturers strive to provide 360˚ protection for their customers. One such example of a vision based safety system is lane departure prevention. Failure to stay in the correct lane was the largest single factor in fatal collisions in the US in 2002 [NHTSA, 2003], playing a role in 32.8% of all cases. Vision Systems can be used to detect lane departure before a collision occurs and mitigate the risk by alerting the diver, adjusting the steering angle or applying the brakes on the opposite side of the vehicle in order to prevent lane departure.

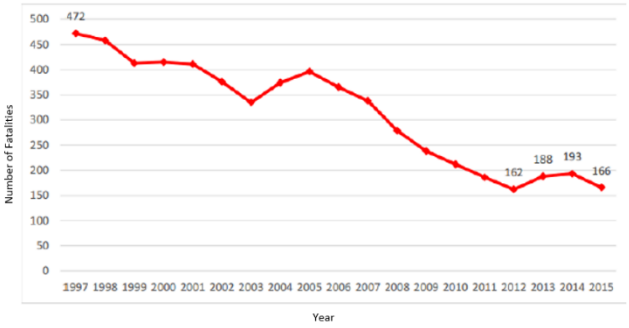


Figure 1: Road Deaths Ireland 1997-2015. Almost linear decline in road deaths in modern times as automotive safety systems improve.

Another example of a vision based safety system is backover protection. There is an average of 232 fatalities

 ISBN 978-0-9934207-2-6

and over 13,000 injuries each year in the US as a result of backovers [Naylor, 2014, Singh S, 2014]. The victims are primarily children under the age of 5 years old and the elderly. As a result, new US legislation will be enforced from May 2018 requiring the mandatory installation of rear view cameras on all new vehicles weighing under 4,500kg. This law will apply to all cars, SUVs, buses and light trucks. A minimum field of view of a 3m by 6m zone directly behind the vehicle and minimum image size is specified. These cameras will not only reduce the risk of backovers by allowing the user to see behind the vehicle, but also through the use of advanced software features such as Object Detection which can allow the vehicle to mitigate the risk of backovers by autonomously alerting the user or by applying emergency braking etc.

In order to be effective in safety applications, these systems must perform without error in both daytime and night time scenarios. In low light situations, the Signal to Noise Ratio (SNR) of a captured image can be greatly reduced. This can lead to noise becoming the dominant signal, as in Figure 2, reducing the amount of useful information available for software analysis thereby impacting the performance of the safety system or in extreme cases making the safety system defunct. In addition to this, technological trends have shown a demand for increased spatial resolution in image sensing applications. In order to obtain increased resolution without increasing the physical size of the sensor, pixel size must be reduced [Gamal, 2009, Gao, Yao et al., 2013]. This further reduces the amount of light captured by each individual pixel and contributes to the reduction of Signal to Noise Ratio in low light applications.

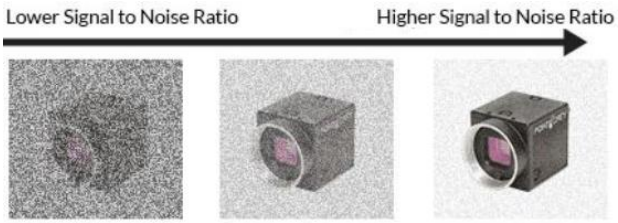


Figure 2: Effect of Signal to Noise Ratio (SNR) on Image Quality. A reduced SNR can result in less analysable content making vision based safety system defunct in low light applications.

2 Current State of the Art and Knowledge Gap

Current research such as Seo et al 2013 [Seo, Sawamoto et al., 2013], Chen et al 2012 [Chen, Xu et al., 2012] and Feruglio et al 2006 [Feruglio, Pinna et al., 2006] all attempt to address this issue at image sensor chip level with a view to reducing internal noise such as temporal, fixed pattern, shot and read noise. These chip level improvements can be very difficult to observe in practical automotive applications as the dominant sources of noise are often created external to the image sensor from the surrounding circuitry and the automotive environment. This can result in camera designers not achieving the full capability of the image sensors low light performance. A knowledge gap exists for a systematic method to analyse how individual state of the art image sensors perform in response to the application specific sources of noise that may occur within the camera unit and particularly within the automotive vehicle.

3 Hypothesis

The method outlined in this paper can be used to systematically characterise modern image sensor performance in response to injected noise as desired by the designer. This allows engineers to tailor schematic and PCB design to filter the precise critical ranges unique to each new model of image sensor in order to maximize the low light performance of automotive cameras.

3.1 Methodology

A method has been developed to characterise CMOS image sensor performance in response to electrical noise on the power supply lines, Figure 3. Image sensor characterisation is carried out by setting the image sensor to stream video data and covering the lens with a non-transparent material in order to ensure that no light strikes any part of the image sensor throughout the characterisation process. Electrical noise of a specific frequency is then coupled on to the image sensor power supply lines using a noise generator. RAW images are captured from the image sensor

 ISBN 978-0-9934207-2-6

and converted into RGB form for analysis using a custom row noise algorithm developed in Matlab in order to quantify image noise. The noise frequency step can be increased and the process repeated for the desired frequency range allowing systematic characterisation of the image sensor performance in response to power supply noise.

The proposed characterisation method is automated with the use of LabVIEW and National Instruments TestStand in order to manage the operation of this analysis for the thousands of frequency steps required when conducting characterisation over a wide frequency spectrum [Gilroy, 2016].

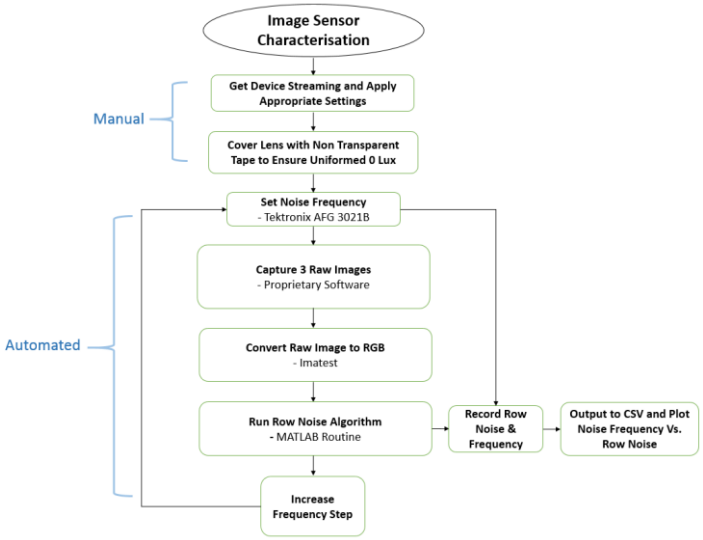


Figure 3: Image Sensor Characterisation Flowchart

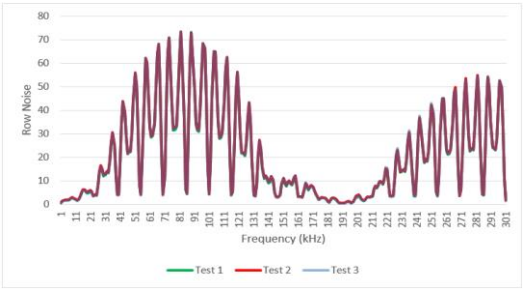


Figure 4: Characterisation Repeatability. Ominvision OV7955 image sensor performance characterisation in reponse to electrical noise carried out 3 times and overlaid to demonstrate repeatability.

Repeatability of the characterisation method has been confirmed using an Omnivision OV7955 image sensor. Image sensor characterisation was repeated three times in succession for a frequency input range of 50Hz to 300 kHz, Figure 4. The test setup has been dismantled and reinstalled between the second and the third test run to rule out any dependency on setup. The results indicate that the characterisation method is highly repeatable.

4 Experimental Results

Performance characterisation in response to electrical noise on the power supply lines has been carried out on two state-of-the-art image sensor models in order to display the use of the methodology proposed in this article. The results of this characterisation can be seen in Figure 5.

Detailed information can be derived from the characterisation process to identify critical ranges of power supply noise that an individual image sensor model is particularly susceptible to. This information can be used in the design of tailored hardware and software filters to reduce the impact of noise on a specific CMOS image sensor and maximise the performance of vision systems for safety critical applications at low light [Gilroy, 2016].

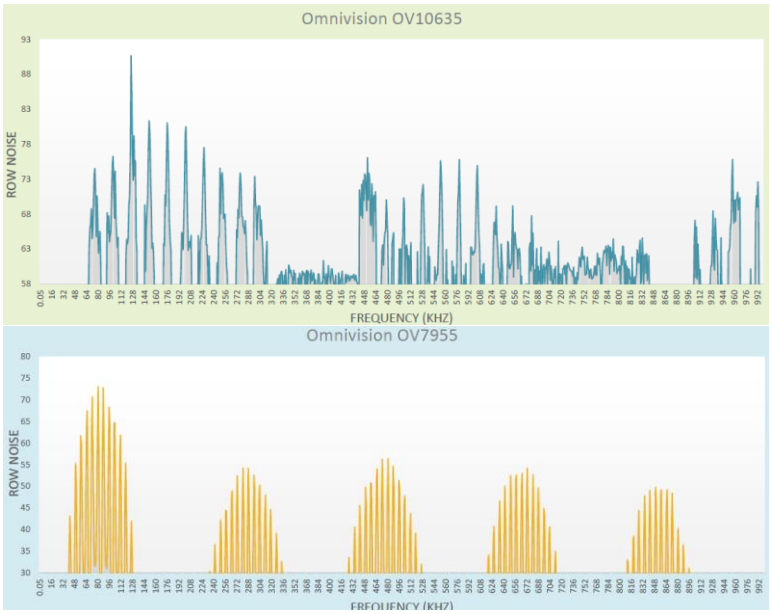


Figure 5: Image sensor performance characterisation in response to electrical noise conducted on to the power supply lines 0Hz-100kHz

 ISBN 978-0-9934207-2-6

5 Conclusions

A systematic, repeatable characterisation method of image sensor performance in response to power supply noise has been proposed which can be used to identify the specific noise frequency ranges that each individual model of image sensor is immune or susceptible to across a wide frequency spectrum. Characterisation of two state-of-the-art image sensor models has been carried out in order to demonstrate the use and effectiveness of the proposed method. The results provided by the characterisation method can be used to identify peak impact and critical ranges of application specific noise that can be used as a design input for the component selection, schematic, PCB and software design of vision systems to improve low light performance in critical safety applications. The proposed characterisation method has focused on image sensor performance in response to electrical noise on the power supply lines only, however the method can be adapted in future to conduct image sensor characterisation in response to any noise source or stimulus as required by the vision system application. The structured, systematic nature of the proposed characterisation method allows the system to be updated by switching the custom row noise algorithm executable file with one defined by the new noise input allowing characterisation to be implemented by following the same steps. The method outlined could also be modified in future to allow the characterisation of complete camera modules for the purposes of debug or validation of vision systems in response to electrical noise.

Acknowledgements

This research was carried out in association with Valeo Vision Systems, Tuam, Co. Galway, Ireland.

References

[Chen, Y., et al., 2012] Chen, Y., et al. (2012). *A 0.7 e− rms-temporal-readout-noise CMOS image sensor for low-light-level imaging*. Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International, IEEE.

[Feruglio, S., et al., 2006] Feruglio, S., et al. (2006). *Noise characterization of CMOS image sensors*. Proceeding of the 10th WSEAS International Conference on CIRCUITS.

[Gamal, A. E., 2009] Gamal, A. E. (2009). *Computational Image Sensors*. Department of Electrical Engineering Stanford University.

[Gao, T.-C., et al., 2013] Gao, T.-C., et al. (2013). *Optical performance simulation and optimization for CMOS image sensor pixels*. Optik-International Journal for Light and Electron Optics **124**(23): 6330-6332.

[Gilroy, S. P., 2016] Gilroy, S. P. (2016). *Impact of Power Supply Noise on Image Sensor Performance in Automotive Applications*. Department of Engineering Technology, Waterford Institute of Technology.

[Naylor, N., 2014] Naylor, N. (2014). *NHTSA Announces Final Rule Requiring Rear Visibility Technology*. from http://www.nhtsa.gov/About+NHTSA/Press+Releases/2014/NHTSA+Announces+Final+Rule+Requiring+Rear+Visibility+Technology.

[NHTSA, U. S. D. o. T. N. H. T. S. A., 2003] NHTSA, U. S. D. o. T. N. H. T. S. A. (2003). *Traffic Safety Facts 2002: A Compilation of Motor Vehicle Crash Data from the Fatality Analysis Reporting System and the General Estimates System*. 06/03/2016, from http://www-nrd.nhtsa.dot.gov/Pubs/TSF2002.pdf.

[RSA, R. S. A., 2015] RSA, R. S. A. (2015). *Provisional Review of Fatalities 31 December 2015*. 13/04/2016, from www.rsa.ie/Documents/Road%20Safety/Crash%20Stats/Provisional%20Review%20of%20Fatalities%202015.pdf.

[Seo, M.-W., et al., 2013] Seo, M.-W., et al. (2013). *A low noise wide dynamic range CMOS image sensor with low-noise transistors and 17b column-parallel ADCs*. Sensors Journal, IEEE **13**(8): 2922-2929.

[Singh S, S. S., Subramanian R. , 2014] Singh S, S. S., Subramanian R. (2014). *Not-in-Traffic Surveillance: Child Fatality and Injury in Nontraffic Crashes—2008 to 2011 Statistics*.

IRISH MACHINE VISION & IMAGE PROCESSING

Conference proceedings 2017

30 August - 1 September 2017

Maynooth University,

Maynooth, Co. Kildare,

Ireland

Published by the Irish Pattern Recognition & Classification Society (web: iprcs.org)

ISBN 978-0-9934207-0-2