Vision based Detection and Tracking in Dynamic Environments with Minimal Supervision

A THESIS SUBMITTED TO
THE SCIENCE AND ENGINEERING FACULTY
OF QUEENSLAND UNIVERSITY OF TECHNOLOGY
IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Alex Bewley

School of Electrical Engineering and Computer Science
Science and Engineering Faculty
Queensland University of Technology

2017

Copyright in Relation to This Thesis

Statement of Original Authorship

The work contained in this thesis has not been previously submitted to meet requirements for an award at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made.

Signature:

Date:

I certify that I have read this thesis and that, in my opinion, it is fully adequate in scope and quality as a thesis for the degree of Doctor of Philosophy.

_____

(Principal Supervisor)

Approved for the University Committee on Graduate Studies:

_____

Abstract

This thesis presents vision based detection and tracking techniques which are suitable for both dynamic and outdoor applications with a moving camera. Existing detection and tracking algorithms rely on strict assumptions – static camera, known scene geometry, or pretrained appearance models – preventing their deployment in new environments. While considerable progress has been made with supervised appearance models, their performance is ultimately governed by biases contained in the training data which are often limited and frequently differs from the data observed when deployed.

A vision based detection and tracking system should be capable of identifying similar objects without pretrained appearance models, while also being robust to distracting background objects. Rather than focusing on training supervised appearance models, this thesis draws on the notion of spatial and temporal consistency to facilitate the use of self-supervised approaches which can learn to adapt their models from the continuous stream of unlabelled video data. This allows the model to learn the context specific nuances in visual appearance for dynamic and unstructured environments with minimal human supervision.

To achieve this goal, three contributions are made. Firstly, a method is presented to detect objects which move independently of the camera motion by identifying and clustering non-rigid keypoints. These motion clusters are then used to incrementally correct an appearance based classifier, allowing it to adapt for new and unseen object appearances. Secondly, the challenge of background distractors in a novel environment is addressed where two methods are presented for modelling the background appearance from a large set of negative only images. Fusing these background models with a pretrained object detector demonstrated superior performance when deployed in a novel environment, particularly with regards to distracting background objects. Thirdly, a framework for associating detections across frames is presented that exploits spatial and temporal constraints, enabling life-long improvement through self-supervised learning.

Keywords

Object detection, motion detection, visual tracking, moving object detection, mobile robotics, dynamic environment, computer vision, collision avoidance, non-stationary modelling, data association, motion segmentation, clustering, online learning, machine learning, deep learning, convolutional neural networks, video analysis.

Acknowledgments

Firstly, I would like to thank my supervisors: Ben Upcroft and Peter Corke. Thank you Ben for frequently creating a reality distortion field during our meetings to help keep me motivated. Thank you Peter for providing a stimulating environment to work in where ideas and help is ubiquitous. I would also like to thank Fabio Ramos from the University of Sydney, who although not one of my supervisors has been a wealth of knowledge and instrumental to several publications throughout the PhD journey.

I would also like to thank the other students and academics at the Robotics lab (past and present) for allowing me to bounce ideas off. You are too numerous to name entirely, but generally if you have enjoyed an afternoon coffee with me, chances are that I've found your presence particularly refreshing. I'd like to acknowledge my desk buddy, Zongyuan Ge for the valuable discussions over the last few years about both work and life. This sentiment also goes to Hu He and Michael Warren for their friendship and guidance in the early years of the PhD.

Moreover, I gratefully appreciate the PhD scholarship I received from the Australian Coal Association Research Program (ACARP). Additionally, I would like to thank members of Anglo American for their support and industry related insights, particularly thanks to Hans Hayes, Steve Amor and Donovan Bellingham.

My final thanks goes to my partner Tracey for her endless support and patience. Especially when I've been socially absent around the times of writing up papers and this thesis.

List of Publications

This section lists all peer-reviewed journal and conference publications that the author has contributed to over the duration of the PhD. The core publications to this research are used to form this thesis-by-publication, while the others were tangential, yet influenced the research direction. The following defines this partition.

Publications which form part of this thesis:

- **A. Bewley**, V. Guizilini, F. Ramos, and B. Upcroft, "Online Self-Supervised Multi-Instance Segmentation of Dynamic Objects," (published) in Proceedings of the International Conference on Robotics and Automation (ICRA), 2014.
- **A. Bewley** and B. Upcroft, "From ImageNet to Mining : Adapting Visual Object Detection with Minimal Supervision," (published) in Proceedings of the 10th International Conference on Field and Service Robotics (FSR), 2015.
- **A. Bewley** and B. Upcroft, "Background Appearance Modelling with Applications to Visual Object Detection in an Open Pit Mine," (published) in Journal of Field Robotics (JFR), 2016.
- **A. Bewley**, L. Ott, F. Ramos, and B. Upcroft, "ALExTRAC: Affinity Learning by Exploring Temporal Reinforcement within Association Chains," (published) in Proceedings of the International Conference on Robotics and Automation (ICRA), 2016.

Publications that are related but not forming part of this thesis:

- **A. Bewley** and B. Upcroft, "Advantages of Exploiting Projection Structure for Segmenting Dense 3D Point Clouds," (published) in Proceedings of the Australian Conference on Robotics and Automation (ACRA), 2013.
- Z. Ge, C. McCool, C. Sanderson, **A. Bewley**, Z. Chen, and P. Corke, "Fine-Grained Bird Species Recognition via Hierarchical Subset Learning," (published) in Proceedings of the

International Conference on Image Processing (ICIP), 2015.

- Z. Ge, **A. Bewley**, C. McCool, B. Upcroft, P. Corke, and C. Sanderson, "Fine-Grained Classification via Mixture of Deep Convolutional Neural Networks," (published) in Proceedings of the Winter Conference on Applications of Computer Vision (WACV), 2016.
- **A. Bewley**, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple Online and Realtime Tracking," (published) in the International Conference on Image Processing (ICIP), 2016.
- N. Wojke, **A. Bewley** and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," (published) in the International Conference on Image Processing (ICIP), 2017.

Table of Contents

List of Figures

List of Tables

Acronyms

ACARP Australian Coal Association Research Program. 1, 11

CRF Conditional Random Fields. 1, 45

DBN Dynamic Bayesian Network. 1, 44, 133

DBSCAN Density Based Spatial Cluster Analysis with Noise. 1, 26, 51

DCN Deep Convolutional (Neural) Networks. xvii, 1, 13, 14, 17, 29–31, 36, 37, 48, 63, 64, 108, 109, 111, 112, 121, 123, 124, 126, 127, 131, 149

DPM Deformable Part Model. 1, 36

EM Expectation Maximisation. 1, 25, 38

FOV Field Of View. 1, 133

GMM Gaussian Mixture Model. 1, 25, 38

GPS Global Positioning System. 1, 127

HOG Histogram of Orientated Gradients. 1, 22, 24, 35

HSV Hue, Saturation and Value. 1, 20, 22

ISM Implicit Shape Model. 1, 36

JPDAF Joint Probabilistic Data Association Filters. 1, 43

k**NN** k Nearest Neighbour. 1, 28

LBP Local Binary Patterns. 1, 20

LDA Linear Discriminate Analysis. 1, 27, 35

LMedS Least Median Squares. 1, 24, 38

MAP Maximum a Posteriori. 1, 36

mAP Mean Average Precision. 1, 41

MCMCDA Markov Chain Monte Carlo Data Association. 1, 43, 44

MHT Multiple Hypothesis Tracking. 1, 43

ML Mostly Lost (not tracked). 1, 46

MOT Multiple Object Tracking. 1, 9, 11, 42, 48, 125

MOTA Multiple Object Tracking Accuracy. 1, 46

MOTP Multiple Object Tracking Precision. 1, 46

MRF Markov Random Field. 1, 38

MT Mostly Tracked. 1, 46

RANSAC RANdom SAmple Consensus. 1, 24, 38

RCNN Region Convolutional Neural Network. 1, 63, 129

RFID Radio Frequency Identification. 1, 127

RGB Red, Green and Blue. 1, 19, 20

ROC Receiver Operator Characteristic. 1, 40

ROI Regions of Interest. 1, 36

SfM Structure from Motion. 1, 38, 127

SGD Stochastic Gradient Descent. 1, 30

SIFT Scale Invariant Feature Transform. 1, 21, 22, 24, 30

SVM Support Vector Machine. 1, 27, 31, 34, 35, 42

YUV luma (Y) and chomiance (UV). 1, 20

Chapter 1

Introduction

Autonomous mobile equipment is increasingly becoming prevalent in several field applications, including agriculture, forestry, mining and transport. A critical sensing capability for autonomous equipment is the ability to detect and track other mobile agents including humans. Due to the highly dynamic nature of typical situations surrounding a mobile plant, a high level of situational awareness is required which includes robust sensing of the number, location and velocity of nearby objects. Inspired by human perception and its natural ability to identify potential hazards using primarily vision, this thesis explores the utility of vision based approaches to detect and track other nearby objects.

Computer vision has the potential to be the Swiss army knife of robotic perception with a proliferative set of capabilities including but not limited to: 3D scene reconstruction, place localisation, semantic scene segmentation, recognition, object detection and tracking. While computer vision has successfully been applied in structured environments such as indoors [Castro et al., 2004, Marron et al., 2006] and urban areas [Cornelis et al., 2008], extending this capability to outdoor and unstructured environments remains a challenge. This research focuses on developing computer vision techniques specifically for object detection and tracking without making assumptions about the scene or objects that have limited the deployment in novel environments.

The primary responsibility of a detection and tracking system is to gather and maintain an estimate of the current state of external objects. This is of particular importance in many driver assistance systems and autonomous vehicle applications where the potential of collisions are a concern. As a result of the race to put autonomous vehicles on main roads, both the

detection and tracking problems have become an increasingly active area of research within the robotics community. While significant progress has been achieved with high resolution laser range finders [Yang and Wang, 2011] and detailed maps [Ferguson et al., 2008, Zhu et al., 2012], vision based approaches remain under-utilised.

Visual object detection is concerned with locating instances of a certain semantic class within the image. Its companion, visual object tracking seeks to continuously locate a specific object instance across frames in a sequence of images to effectively maintain a temporally coherent trajectory. Together, these capabilities offer a passive approach to locate and estimate the velocity of nearby moving objects as required in many industrial applications. For example in a mining context, it is necessary for a perception system on a haul truck to distinguish between moving objects (such as other vehicles and pedestrians) from background clutter. Furthermore, tracking these objects by associating the detections across frames leads to predicting their future position which is critical in avoiding collisions.

Computer vision based detectors are increasingly employing data driven approaches such as a classifier to decide whether an image region contains an object of interest or only background [Dalal and Triggs, 2005]. These classifiers are typically trained offline using supervised methods (also referred to as batch learning) where the learnt model is fixed for online deployment. Training such models requires extensive object labelling which is performed by a human making it time-consuming and results in models that are generally inflexible to changes. Additionally, in many real-life scenarios the variation in appearance between images used to train the model and the data experienced during deployment may increase detection errors. This variation becomes especially challenging when using a small training set to learn a detector for a large number of different objects since the number of cases which can be learnt is limited.

Ideally, it is preferable that the appearance models used for detection and tracking are learnt automatically from the observed environment with minimal to zero human supervision. This will allow computer vision based detection and tracking systems to be rapidly deployed in new environments, where their performance increases overtime as they observe the various visual patterns specific to the environment in which the system has been deployed. This raises the overarching question of: *How can appearance models for object detection and tracking be learnt during deployment?* To answer this question, we must first consider how appearance models are used in existing detection and tracking algorithms and what limits their deployability

in new environments.

The next two sections provide a brief introduction to vision based detection and tracking where specific gaps are identified leading to three open research questions.

1.1 Visual Object Detection

The aim of visual object detection is to locate the multiple instances (if any) of a specified object type within the image boundary. Many classical [Dalal and Triggs, 2005, Viola and Jones, 2001] and modern [Benenson et al., 2013, 2012, Dollar et al., 2014] detection algorithms employ the popular top-down sliding window paradigm which treats a single image as a collection of sub window images. This is generally described by the following steps:

1. Select a set of sub windows covering every location, scale and aspect ratio that is likely to contain a potential object.
2. Extract a feature representing the visual appearance of each sub window.
3. Use a classifier based model to label each window as one of a number of known categories which could be a type of background or an object of interest.

This framework presents two key benefits, firstly, multiple objects are naturally detected by the multiple window sampling of step 1, and secondly it can be easily adapted for different types of objects by changing the features and classifier used in steps 2 and 3 respectively. For example, the classifier can be trained to discriminate between background and faces [Viola and Jones, 2004], pedestrians [Dalal and Triggs, 2005, Viola and Jones, 2003], or cars [Tzomakas and von Seelen, 1998]. However, these classifiers generally rely on supervised pretraining within an offline batch optimisation process, requiring a large set of manually labelled training samples to capture variation in the feature distribution.

Obtaining comprehensive labelled datasets is generally expensive and often impractical for specific outdoor applications, especially when considering multiple types of objects, each with a high variation in visual appearance. In recent years, extensive databases [Deng et al., 2009, Fei-Fei et al., 2007, Torralba et al., 2008] of labelled image data for object classification have been made available to standardise the evaluation of visual object detection algorithms. Despite the continuous improvement in performance as measured by these benchmarks, it is still questionable as to how well this performance translates to real world scenarios [Pinto et al.,

2008]. Particularly, as several such benchmark datasets are known to contain various forms of dataset bias [Torralba and Efros, 2011], their performance is likely to be negatively impacted when applied to images captured in a different context with different statistical properties.

Differences between the appearances captured in the training data and that observed during deployment presents a challenge for object detection. Specifically, objects – novel or under-represented in the training dataset – may go undetected using a pretrained appearance model alone. Ideally, these models should be updated with new objects during deployment, however when such models are not able to detect an object, the following question is raised.

1. *How to discover objects which may have appearance characteristics unknown by the detector?*

To address this question, motion is explored as an independent visual cue where the objects of interest are considered to be non-stationary in the environment. Additionally, as this thesis targets mobile applications, the presented motion analysis techniques are designed to compensate for camera ego-motion. This allows objects which are moving in the scene to be discovered independently of any pretrained object appearance model. Furthermore, with the ability to identify moving objects, missed detections in future frames are reduced by correcting the appearance model to identify static objects with similar appearance.

Contrary to correcting missing detections, another detection error occurs when background regions are falsely classified as objects. This is further exacerbated when the detector is deployed in an environment where the background has a different visual appearance to images in the training set. As it is unfeasible to anticipate all potential background appearances a priori, it is desirable to adapt the detector from observations made at deployment. This leads to the following research question.

2. *How to adapt an appearance based detector for deployment in new and novel environments with different background characteristics to the training data?*

This thesis addresses this question by exploring background modelling techniques to distinguish objects of interest from challenging, deployment specific background regions. By focusing only on images containing background, vast quantities of training data can be collected

with minimal effort compared to annotating objects. Additionally, the size of the background training set can be further magnified by considering temporal segments in a video sequence in combination with selecting multiple region patches per frame.

In this scenario, the number of background images significantly exceeds the number of object training examples. Retraining a classifier directly on such an imbalanced dataset is likely to result in a bias towards treating everything as background. While a pretrained detector can be used to select only challenging background samples [Felzenszwalb et al., 2010], given the magnitude of the background set of novel images, the challenging patches remain numerous.

The background modelling techniques in this thesis are presented to work in parallel to a pretrained object detector. This allows the background model to be trained in isolation using unsupervised techniques to characterise common background appearances which bypasses the challenges of training on an imbalanced dataset. Finally, a fusion framework is presented to combine the output of both models and facilitate the adaptation of a detector to handle distracting appearance features encountered in the new environment.

In addressing both the first and second research questions, techniques are presented for learning the appearance of novel objects and background respectively. Particularly, these techniques focus on minimising the amount of manual supervision required to deploy vision based detection into new environments. To achieve this, focus is given to uncovering and exploiting the implicit structure of video sequence data to characterise objects as anomalies in this structure. Firstly, geometric constraints are used to define background motion as a rigid structure where anomalies indicate moving objects. Secondly, unsupervised techniques are explored to identify common and re-occurring patterns in the background, where anomalies indicate novel objects which are further resolved with a general pretrained model.

1.2 Visual Object Tracking

Once an object of interest is detected, it is desirable to track the object from frame-to-frame in a video sequence. Moreover, this thesis considers Multiple Object Tracking (MOT) as there could be multiple objects detected per frame. MOT can be described as associating detections in the current frame with potentially many track hypotheses constructed from previous frames. These associations are guided by an assignment cost which considers the similarity between

current detections and tracked objects.

While visual object tracking has been extensively researched [Li et al., 2013, Wu et al., 2013, Yang et al., 2011, Yilmaz et al., 2006], the majority of what is considered state-of-the-art in tracking is either designed for single object tracking [Babenko et al., 2010, Hare et al., 2011, Kalal et al., 2012, Pernici and Del Bimbo, 2014] or performed in an offline batch process where information in future frames can be exploited [Dicle et al., 2013, Pirsiavash et al., 2011, Zamir et al., 2012]. Since this thesis is concerned with the use of tracking for autonomous applications, it is imperative to handle an arbitrary and dynamic number of objects while also being constrained to information available only in the current and past frames.

Perera et al. [2006] and Huang et al. [2008] showed that a classical data association [Kuhn, 1955] technique can be applied to match detections to existing trajectory hypotheses. In this framework, each detection of the current frame is matched to a single trajectory based on optimising the total assignment cost [Kuhn, 1955]. The assignment cost is traditionally represented as the difference between the detected position and a predicted position along the trajectory. However, these methods rely on accurate motion estimation restricting them to tracking in a registered 2D ground plane [Perera et al., 2006] or with a static camera [Huang et al., 2008]. Appearance information is increasingly being used to help alleviate these restrictions by estimating the probability that two detections to represent the same object, referred to as visual affinity. This commonly takes the form of employing the Bhattacharrya coefficient [Milan et al., 2014] or via normalized cross-correlation [Geiger et al., 2014]. However these affinity models are fixed, making them sub-optimal since they do not consider deployment specific nuances of the objects' appearance.

The aim of the first two research questions is to adapt appearance models by learning from the deployed environment for detection. This idea can also be applied to estimating a visual affinity based assignment cost for tracking. Particularly, given the underlying appearance representation describing detected objects, this thesis seeks an appearance based assignment cost which is tuned to the specific detections encountered during deployment. This gives rise to the third and final research question addressed in this thesis:

3. *How to adapt the assignment cost using data collected at deployment?*

The online and adaptive nature of single object trackers [Hare et al., 2011, Kalal et al.,

2012] make them alluring for this work since they incrementally improve their appearance models from the data observed during deployment. While these approaches could be extended to MOT by instantiating a new tracker with different initial conditions from a base detector [MacCormick and Blake, 1999], issues regarding long-term knowledge retention restrict their applicability to answering the research question. Particularly, single object trackers learn a separate appearance model for detecting a specific object instance from other instances of the same category. Once the object leaves the scene the learnt model loses its value as it was trained to treat everything - including other object instances - as not the target object.

As data association is at the centre of MOT, this thesis addresses the third research question by formulating the estimation of visual affinity as a binary classification problem where the goal is to predict if two detections correspond to the same object. In this framework, the training data is in the form of a pair of detections with their appearance features and a label indicating if they match. Labels for this data are obtained automatically by exploiting spatial constraints for non-matching pairs and temporal consistency to find pairs with strong affinity. Finally, as this classifier models the affinity of any two detections it generalises to MOT while also enabling lifelong learning of deployment specific nuances through the self-supervised framework.

1.3 Scope and Contributions

This section describes the research contributions in the areas of vision based detection and tracking in unstructured environments.

1.3.1 Scope

The goal of this thesis is to investigate methods for passive vision based object detection in field environments. The work is carried out as part of an Australian Coal Association Research Program (ACARP) project which is aimed towards improving the situational awareness of both in-vehicle and tele-operated drivers in a surface mining context. For this reason, the components of this research that involves using environmental specific cues, includes experiments that are focused on surface mining environments. Due to the constantly changing landscape this thesis focuses on techniques that use as little previous knowledge as possible about the environment (assumptions of the scene structure, previous build maps, etc.).

The work outlined in this thesis focuses on using computer vision as the sole sensing modality with the objective to detect and locate multiple objects within a scene and maintain individual object identities across frames. For the methods presented where the main contributions are broadly applicable (e.g. object tracking), standard publicly available benchmark data was used to facilitate fair comparison to existing literature. The integration of this work with other sensing technologies or into a complete collision avoidance solution is beyond the scope of this thesis and left for future work.

1.3.2 Contributions

This thesis is presented in the form of a thesis-by-publication where the main methodology chapters are composed using related research publications as follows.

In Chapter 3, this thesis addresses the first research question of: *How to discover objects which may have appearance characteristics unknown by the detector?* Specifically, a motion clustering based approach is presented for object detection that operates independently of pre-trained appearance models. Multi-view geometry is used to model camera motion and represent the background as the main rigid structure while outliers are clustered and filtered to locate independently moving objects. These motion clusters were further used to learn online an appearance based detector for discriminating between multiple newly discovered objects along with the background. This chapter is composed of the paper titled "Online Self-Supervised Multi-Instance Segmentation of Dynamic Objects", which is published in the proceedings of the International Conference of Robotic Automation in 2014. This includes the following contributions:

- A motion clustering approach to identify independently moving objects in the scene. This method is able to identify any number of objects [Ester et al., 1996] and is able to handle the case of a moving camera [Guizilini and Ramos, 2013].
- A self-supervised learning framework that incrementally adapts a multi-class classifier to discriminate the individual objects from each other and the static background.
- The use of similarity between the clustering output and a multi-class classifier for assigning temporally consistent labels enabling the continual adaptation of the classifier online.

Chapter 4 of this thesis describes two background modelling approaches that aims to characterise the appearance of the main semantic categories that make up the background. The chapter

contains a published conference paper and an accepted journal paper. The first modelling approach is published in the paper titled: "From ImageNet to Mining: Adapting Visual Object Detection with Minimal Supervision" and was presented at the International Conference on Field and Service Robotics in 2015. This work is extended with an alternative model suited for Bayesian fusion as presented in the accepted paper titled: "Background Modelling for Adapting to the Deployed Environment" which is to appear in the Journal for Field Robotics. Together these papers address the second research question: *How to adapt an appearance based detector for deployment in new and novel environments with different background characteristics to the training data?* Where the background model is used to adapt a generic state-of-the-art DCN detector. This was demonstrated in a mining context where the environment is visually different from many of the datasets for which a pre-trained classifier is trained. This chapter provides the following contributions:

- An investigation of a state-of-the-art detector [Girshick et al., 2014] originally tuned for internet image (ImageNet) data and applied to object detection in the unstructured environment of an open pit mine.
- Adaptation of a DCN detector by formulation of a novelty based detector using a background model learnt by clustering intermediate DCN features.
- A Bayesian approach to fusing the discriminative power of a modern DCN detector with the novelty likelihood modelled using the background model.

In Chapter 5, appearance based descriptors are extracted from the detector in a similar fashion to the background model but instead purposed for the task of data association in a tracking framework. Additionally, by leveraging spatial and temporal constraints, a motion invariant appearance model for measuring similarity between these descriptors is learnt allowing for detections to be associated across frames. This overall tracking framework satisfies the third research question: *How to adapt the assignment cost using data collected at deployment?* The details of this framework is presented using the paper titled: "ALExTRAC: Affinity Learning by Exploring Temporal Reinforcement within Association Chains", which is published in the proceedings of the International Conference of Robotic Automation in 2016. This paper forms the content of Chapter 5 and offers the following contributions:

- The modelling of visual affinity with a probabilistic classifier. This is used to assign detections in the current frame to existing objects.

- A method for exploring temporal history of associations to gather matching examples with high variance to improve classification in such cases. This provides a source of positive training examples – without explicit labelling – to facilitate self-supervision.
- An online approach for balancing classifier bias by automatically collecting non-matching examples using spatial co-existence constraints.

The contributions outlined above address each of the research questions as a set of publications to form the methodology chapters of this thesis.

1.4 Thesis Outline

The remaining chapters of this thesis are outlined below.

Chapter 2 contains a review of relevant literature covering various sub-components related to vision based detection and tracking with particular focus given to methods suited for mobile platforms and methods which adapt to changes in visual appearance.

Chapter 3 presents a novel motion clustering based approach to detection with no pretraining. This method exploits geometric constraints to model the rigid background motion in a video sequence while simultaneously detecting independent moving objects. A multi-class classifier is continuously updated using information from the motion clusters to segment the objects from the background and also maintain object identities across frames. This method is applied to vehicle mounted and mining video sequences to demonstrate the systems ability to learn object models from the deployed environment.

Chapter 4 presents an appearance based detector that adapts a pre-trained DCN detector to a different domain by incorporating a novel background model. This background model is constructed from a large reservoir of patches extracted from background only image sequences requiring minimal annotation effort. At test time, detections are validated using its similarity to the background model and is shown to dramatically reduce the false positive rate with only a minor drop in recall. Two variants of the background model are proposed and investigated for their improvement of the detector in a mining environment. Furthermore, a Bayesian fusion framework is proposed which is demonstrated to provide the best detection performance compared to the sub components.

Chapter 5 develops a tracking approach based on the learning of an affinity model that

measures the similarity of two detections. This proposed method exploits both spatial constrains and temporal consistency to actively model the pairwise similarity between detections with a probabilistic classifier. The output of this model can be inserted into any tracking-by-detection framework to improve data association in any arbitrary environment.

Chapter 6 provides a discussion of the presented work, conclusions drawn and directions for future work. This chapter also includes a summary of the key contributions made in this thesis.

Chapter 2

Background and Related Work

This chapter presents an overview of existing research in the areas related to computer vision based detection and tracking. While the primary focus is on learning to adapt to the visual changes in unstructured environments, this chapter also discusses some of the theory underlying both multi-view geometry and pattern recognition using machine learning techniques as these concepts have a critical role in the development of this research field as a whole.

A brief survey of visual feature representations is provided in Section 2.1.1. This includes both basic features such as colour and texture to more complex hand-crafted visual features traditionally used in pattern recognition. While in recent years data driven feature representations have surpassed the performance of hand-crafted features in many vision related tasks, this brief survey provides some historical context and serves as a light introduction to the later described data driven features. Additionally this section covers how these features are used to describe and identify distinctive keypoints in an image.

Section 2.1.2 addresses how keypoints with their respective features are matched, either to the same point in a different frame or to a database of features which hold additional information regarding what the feature may represent. This section is further broken down into relevant types of matching with a discussion on the relationship between geometric constraints and computation complexity.

Section 2.1.3 describes the principals of machine learning (a.k.a. data driven) techniques for identifying specific patterns. These techniques frequently form a basis for detection and tracking methods discussed later in this chapter. This includes a brief overview of DCNs based learning architectures that have recently made a profound impact to computer vision.

Issues around availability of training data and specificity for industrial environments are also discussed.

Various formulations of visual object detection are discussed in Section 2.2. This section details how combinations of feature representations, feature matching and machine learning principals are used to identify and locate objects of interest within both still images and video sequences. While detection has long been a topic of research, particular focus is given to machine learning based visual detection methods covering both unsupervised and supervised approaches. For completeness this section explains common detection evaluation metrics which are used in later chapters of this thesis.

Section 2.3 addresses the issue of maintaining object identities overtime in a video sequence for object tracking. This includes the problems of data association, motion prediction and the accumulation of appearance information. Various approaches are compared by considering their functionality and their limitations. For completeness this section contains tracking metrics which are used for evaluation throughout the literature and in later chapters of this thesis.

Section 2.4 provides a discussion about the research reviewed in this chapter. It summarises some of the major trends in vision-based detection and tracking along with their application to challenging outdoor environments. Finally, this section concludes by positioning the work presented in this thesis with relation to the observations drawn from the prior literature.

2.1 Background

This section provides a brief overview of basic computer vision and machine learning concepts used in the related work listed in the next section and also in the later chapters of this thesis. A reader familiar with computer vision and machine learning may choose to skip to Section 2.2 where the work of this thesis is positioned among existing detection literature. The background concepts of this section are broken into appearance feature representations, matching across views and machine learning methods.

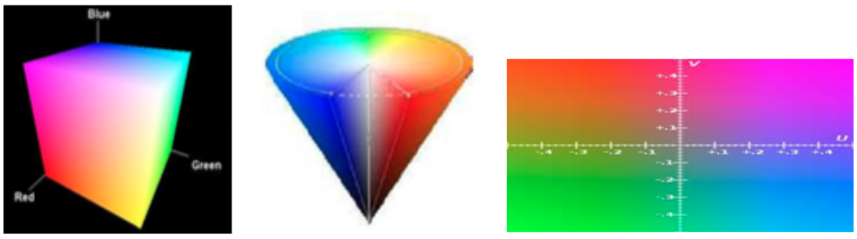


Figure 2.1: Geometric representations of the RGB (left), HSV (middle) and UV (right) of the YUV colour spaces. Reproduced from [Trifan et al., 2013].

2.1.1 Appearance Feature Representations

Extracting a feature vector which represents visual appearance forms the basis of many detection and tracking techniques presented later in this chapter. This section provides an overview of various visual representations used for describing appearance within an image. The types of representations can be partitioned into either basic visual cues such as colour and texture, or as a more complex hand crafted appearance descriptor.

Colour and Texture Cues

Intuitive and simple visual attributes such as colour and texture play a key role in how humans describe the visual appearance of their surroundings. For example, in a mining context, the background is visually dull while foreground objects are purposely visually apparent in contrast through the use of bright and salient colours. For example, the background can mostly be described as *coarsely textured* rock and *clear blue* sky, while foreground objects include *yellow* truck, *white* light vehicles and personnel wearing *orange* shirts. Colour and texture based features are easy to compute with many methods designed to tolerate changes in illumination.

Colour cameras capture the light falling onto the sensor by using Red, Green and Blue (RGB) colour filters to separate the light across three channels. In structured environments, if the colour of the object of interest is known a priori then simple colour filters can be constructed to segment the object from the background [Trifan et al., 2013]. However, in outdoor environments, due to the various lighting conditions and materials observed, RGB becomes less informative when distinguishing the semantic class of objects in the scene.

Several colour spaces have been proposed over the years with the aim of separating the reflectance properties of the objects surface from the spectral power distribution of the illumination source [Kuehni, 2003]. Other than the default RGB other common colour spaces

used in computer vision applications include Hue, Saturation and Value (HSV), luma (Y) and chomiance (UV) (YUV) [Reinhard and Pouli, 2011]. Geometric representations of these colour spaces are shown in Figure 2.1. Alternatively, Van De Weijer et al. [2007] map the RGB coordinates to linguistic colour names and demonstrate improvements in image retrieval applications. Variations caused by shadows in natural daylight have also been modelled to remove the lighting component from the material property components of the surface reflections [Corke et al., 2013]. However, shadows may also be informative as Tzomakas and von Seelen [1998] exploited the shadows created by vehicles to aid detection.

The texture of an image region can provide an alternative visual cue to indicate the surface material and consequently lead to higher level classification. Gabor [1946] first characterised texture as a set of functions defined by spatial frequency and orientation, which also describe the fundamental human attentive responses to patterns [Julesz and Bergen, 1983]. Leung and Malik [2001] defined a discrete representation as a set of linear Gaussian derivative filter kernels that are convolved with the input image (see Figure 2.2). Ojala et al. [2002] proposed an alternative approach that produces a compact binary representation by comparing the centre pixel intensity with surrounding image locations. These Local Binary Patterns (LBP)s can be efficiently computed and further reduced to 36 unique patterns after considering rotation invariance (see Figure 2.2).

When representing an image region, a histogram of colours can be constructed from the colour values of the pixels within the region [Bradski, 1998, Swain and Ballard, 1990, Town and Moran, 2004]. This characterises the colour distribution found in a selected region which can be compared to other regions using either the intersection [Swain and Ballard, 1990] or using the χ^2 (chi squared) test statistic [Hafner et al., 1995]. Another traditional method of representing an image region using colour is to compute the mean and covariance [Fieguth and Terzopoulos, 1997].

Several studies have shown that combining colour and texture results in improved performance for tasks such as segmentation [Brox et al., 2003, Micusik and Pajdla, 2007] and tracking [Serby et al., 2004, Town and Moran, 2004]. Alvarez and Vanrell [2012] investigated the co-occurrence of texture and colour types in a database of texture images to develop a joint representation that improves surface classification. Khan et al. [2013] also showed that evaluating colour and texture visual cues independently and then combining them before classification

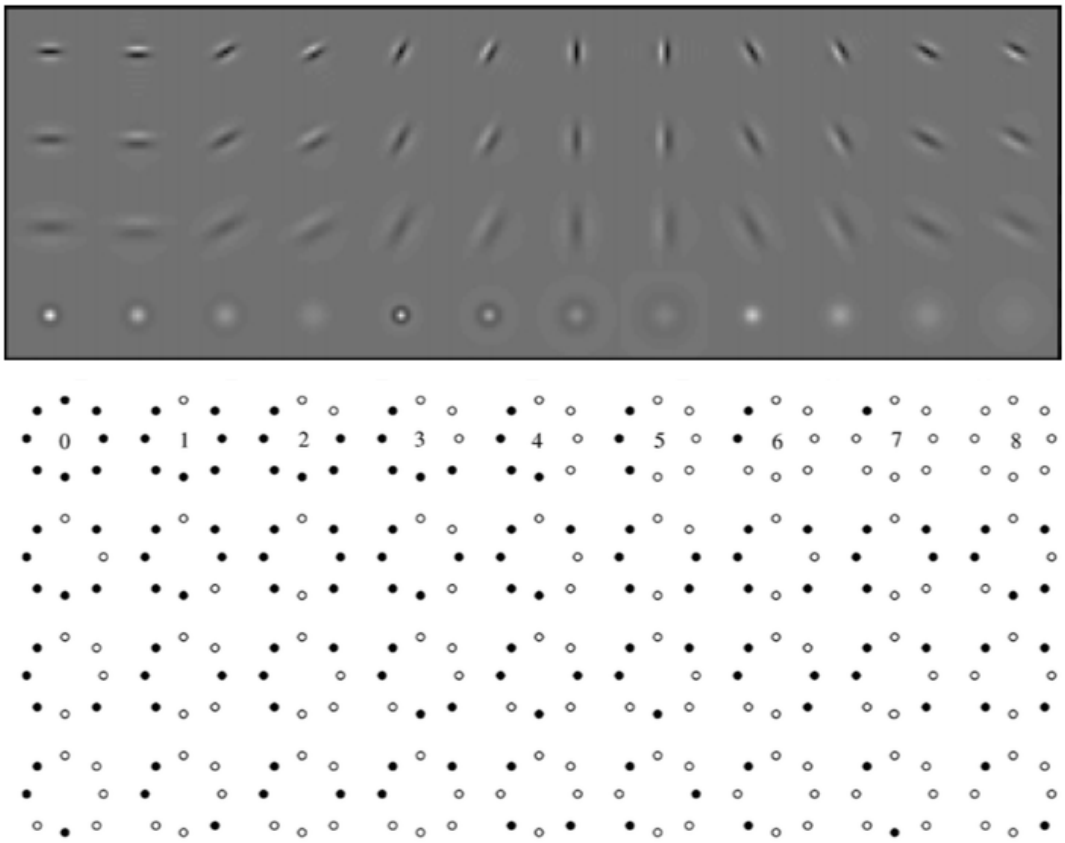


Figure 2.2: Texture representations. Top: 2D kernels representing 48 different Gaussian filters as presented in [Leung and Malik, 2001]. Bottom: The 36 unique and rotationally invariant 8-bit local binary patterns. Reproduced from [Ojala et al., 2002].

achieves superior results in categorisation tasks.

Hand Crafted Descriptors

For a given image location, appearance properties of the local neighbourhood are encoded to form a *descriptor*. Descriptors express the visual information of a point by forming either a histogram or vector containing various attributes. While a rudimentary descriptor can be formed by considering a histogram of the previously discussed colour and texture cues, this section discusses hand crafted features specifically designed for salient and distinctive image locations called *keypoints*. Several methods have been designed to optimise repeatable matching of keypoints under different viewing and lighting conditions. Another objective of descriptors is to capture distinctive attributes of a keypoint to minimise the number of false matches to other features.

Arguably the most popular feature detector and descriptor is Lowe's Scale Invariant Feature

Transform (SIFT) approach [Lowe, 2004], which set a significant benchmark for accuracy feature matching offering superior distinctive and transform invariant characteristics at the expense of computational cost. Other commonly used features include: Speeded Up Robust Features (SURF) [Bay et al., 2008], Centre Surround Extrema (CenSurE) [Agrawal et al., 2008], Features from Accelerate Segment Test (FAST) [Rosten and Drummond, 2006] and Maximally Stable Extremal Regions (MSER) [Matas et al., 2004].

Recently, a diverse array of binary descriptors have also been proposed to further address storage and computation complexity of descriptor extraction including: Binary Robust Independent Elementary Features (BRIEF) [Calonder et al., 2010], Oriented FAST and Rotated BRIEF (ORB) [Rublee et al., 2011], Binary Robust Invariant Scalable Keypoints (BRISK) [Leutenegger et al., 2011] and Fast Retina Keypoints (FREAK) [Alahi et al., 2012]. The descriptor vectors extracted by these methods are directly compared to other descriptors by evaluating either the Euclidean distance or Hamming distance for the binary case between two vectors.

Descriptors are also used to describe a rectangular region within an image. In contrast to simply using a histogram of colour and texture cues, hand crafted features attempt to characterise the overall shape of the region. Viola and Jones [2003] make use of a cascade of Haar wavelet features to detect both faces and pedestrians. The Histogram of Orientated Gradients (HOG) proposed by Dalal and Triggs [2005] separates the edge gradients into a grid structure covering the patch region. When combined with a standard classifier this HOG feature has demonstrated significant discriminative performance in pedestrian detection.

These hand crafted features aim to characterise the local shape and therefore designed to be invariant of colour. However, in situations where the colour of the objects of interest is known and dissimilar to the background context colour descriptors can be used. For example van de Weijer et al. [2006] propose a concatenation of Hue histogram with the SIFT descriptor. Bosch et al. [2008] computed the SIFT descriptor on all three channels of the HSV colour space before concatenating them. [van de Sande et al., 2010] proposed a new colour space derived from the colour within the image and computed a descriptor in a similar fashion to Bosch et al. [2008]. A structured evaluation of colour descriptors can be found in either [Burghouts and Geusebroek, 2009] or [van de Sande et al., 2010].

2.1.2 Matching between Views

Feature matching forms the basis for many visual motion estimators and image recognition based approaches covered later in this chapter.

Visual Similarity Matching

Many object detection and object tracking methods rely on the matching of image feature descriptors in the current image with descriptors from a database or the previous image respectively [Lehmann et al., 2009, Leibe et al., 2007, Pernici and Del Bimbo, 2014]. This is generally achieved by computing the Euclidean distance between the reference descriptor from the current image and each of the stored descriptors to find the closest match [Lowe, 2004]. Due to the large number of features detected in each image or stored in a database, evaluating the distance for every potential match is computationally expensive. To address this, various methods have been proposed in the literature for efficiently restricting search for suitable matches by exploiting data structures [Muja and Lowe, 2009]. See [Dhanabal and Chandramathi, 2011, Kumar et al., 2008] for a comparative review of various nearest neighbour methods.

Optical Flow

Optical flow is a field of computer vision that is generally concerned with the pixel-wise correspondence between images taken at two different times. This is often used to evaluate the pixelwise motion between consecutive video frames in a video sequence. The optical flow problem is considered under constrained as the relative motion of objects in the scene can move independently of the camera motion. The types of optical flow can be divided into sparse and dense techniques, where pixel motion is computed for only salient points or every pixel respectively.

Lucas and Kanade [1981] pioneered the sparse optical flow by proposing a locally differentiable framework that constrains the problem by imposing smoothness and spatial consistency assumptions. Since this approach works on a local image patch, it is sensitive to the common aperture problem created by an ambiguous match along edges. To overcome this, salient or corner features are selected as the reference points [Shi and Tomasi, 1994, Tomasi and Kanade, 1991]. Later, Bouguet [2000] proposed a hierarchical pyramid structure for matching

pixels with motion greater than the pixel neighbourhood used in [Lucas and Kanade, 1981]. Alternatively, the matching of rich descriptors discussed earlier could also be used to estimate keypoint displacements across frames [Guizilini and Ramos, 2013].

One of the earliest and successful methods for dense optical flow is that of Horn and Schunck [1981] which combines the brightness consistency assumption with a first-order Taylor series approximation. This method is optimised over a number of iterations, where the optical flow is constrained with a form of regularisation that penalises large displacements. More recent approaches robustly deal with large displacements by matching HOG [Brox and Malik, 2011] or SIFT [Liu et al., 2008, 2011] descriptors to avoid local minima and reduce overall computation time. Another recent approach to multi-frame optical flow based on motion layers [Sun et al., 2012] offers a favourable approach to motion segmentation towards moving object detection.

Epipolar Geometry

The epipolar geometry is the intrinsic projective geometry between two views. It is independent of scene structure, and only depends on the internal camera parameters and relative pose. The geometric relationship are encapsulated by the essential matrix for calibrated cameras or more generally the fundamental matrix. The internal parameters of the fundamental matrix can be computed from image correspondences alone and is independent of scene structure [Hartley and Zisserman, 2004].

Computing these parameters in the presence of outliers from independently moving objects requires a robust estimator. Two commonly used robust estimators are RANdom SAmple Consensus (RANSAC) [Fischler and Bolles, 1981] and Least Median Squares (LMedS) [Rousseeuw, 1984]. As these methods generally require high inlier ratios, several improved variations have been made which maximise the posterior of the solution [Torr and Zisserman, 2000], weight candidates on matching scores [Chum and Matas, 2005] or checking spatial consistency [Sattler et al., 2009].

2.1.3 Machine Learning

The visual feature representations discussed earlier in Section 2.1.1 can be thought of as data models while this section introduces algorithmic models for analysing the data extracted [Breiman,

2001a]. In particular, this section focuses on general machine learning concepts and models that are commonly used in object detection and tracking. Namely, approaches for unsupervised clustering and supervised classification. The aim of these techniques is to learn a model that takes input data and assigns it to one of a set number of categories.

Unsupervised Clustering

In unsupervised clustering the labels are not defined a priori but rather the data is partitioned based on the structure in the data itself. This assumes that the data naturally forms groups known as clusters where samples within a cluster are more similar to each other than from samples from other cluster. While the membership or the number of clusters is unknown, cluster analysis present techniques for uncovering these clusters, which can be used to group features based on their similarity. Concretely, data samples could be any of the features mentioned in Section 2.1.1 extracted from the points or regions in an image. It is anticipated that points on the same object would share visual characteristics as captured by the feature and can be grouped to discover the extent of the object. Here a few common clustering techniques are introduced and their capabilities are also illustrated in Figure 2.3.

A classical approach to clustering is the k-means partitioning method of Jain and Dubes [1988] initially select k of the data samples to represent the cluster centroid and then iteratively alternates between assigning samples to the nearest cluster centroid and updating the centroid to be the mean of the samples assigned to the cluster. This process is analogous to fitting a Gaussian Mixture Model (GMM) with k components to the dataset using the Expectation Maximisation (EM) algorithm [Dempster et al., 1977], while assuming a spherical covariance for each component. The mean shift algorithm [Comaniciu and Meer, 2002] assumes a smooth gradient in the local density of points and uses an uphill climbing strategy to shift the centroid in the direction of the locally densest region. Unlike k-means this method uses all the points and only computes the local density defined by a bandwidth parameter. Finally, near duplicate centroids are merged. From Figure 2.3 it can be seen that the clusters found using (with the exception of the agglomerative clustering) tend to partition the data based on their euclidean (or Mahalanobis distance for the GMM) which disregards the manifold of the data. A common approach to overcome this is to construct a similarity matrix between all points and using the eigenvector of this matrix as a partitioning point to split the dataset into two halves [Meila and

Shi, 2000, Shi and Malik, 2000].

Another family of clustering techniques take a bottom up approach by incrementally merging samples to form clusters according to some merging criteria. The agglomerative clustering approach proposed by [Sibson, 1973] initially treats each point as its own cluster and recursively merges the closest pairs of clusters until a set number of clusters remain. Similarly, Jarvis and Patrick [1973] proposed a merging strategy that considers the k nearest neighbours of each point and only merged points if they shared a set number of neighbours.

The Density Based Spatial Cluster Analysis with Noise (DBSCAN) proposed by Ester et al. [1996] considers the local point density as a merging criteria. Beginning with a point chosen at random, the radial neighbourhood is checked that there are a minimum number of local neighbours to grow the cluster. If the point satisfied this point density test then it and its neighbours are used to form a cluster, the neighbours are then checked recursively until no more neighbours can be added to the cluster. Unchecked sample points are also tested for a minimum local density and the process continues until all points have been checked. This technique has many desirable properties such it naturally identifies the number of clusters in the data, identifies outliers and also forms irregular shaped clusters that follow the data manifold (shown in Figure 2.3). Additionally, as it makes a single pass over the data it is computationally efficient. However, it does require two parameters to define a fixed point density: radius and minimum number of points. Extensions such as [Ankerst et al., 1999, Ertoz et al., 2003] address this limitation with added computational cost of ordering points by local density or constructing a nearest neighbour graph.

Alternative methods also attempt to estimate the number of clusters by finding representative points in the data. Frey and Dueck [2007] proposed the affinity propagation algorithm is given a similarity matrix of all pairs of points where messages are exchanged between data points until a high-quality set of exemplars and corresponding clusters gradually emerges. Constructing the similarity matrix can be slow, often requiring efficient data structures such as [Muja and Lowe, 2009] or compressing the data [Ott and Ramos, 2013]. For specific applications where the data is dense and represents a real scene, smoothness constrains and projection geometry can also be used to simplify the search for clustering applications [Bewley and Upcroft, 2013].

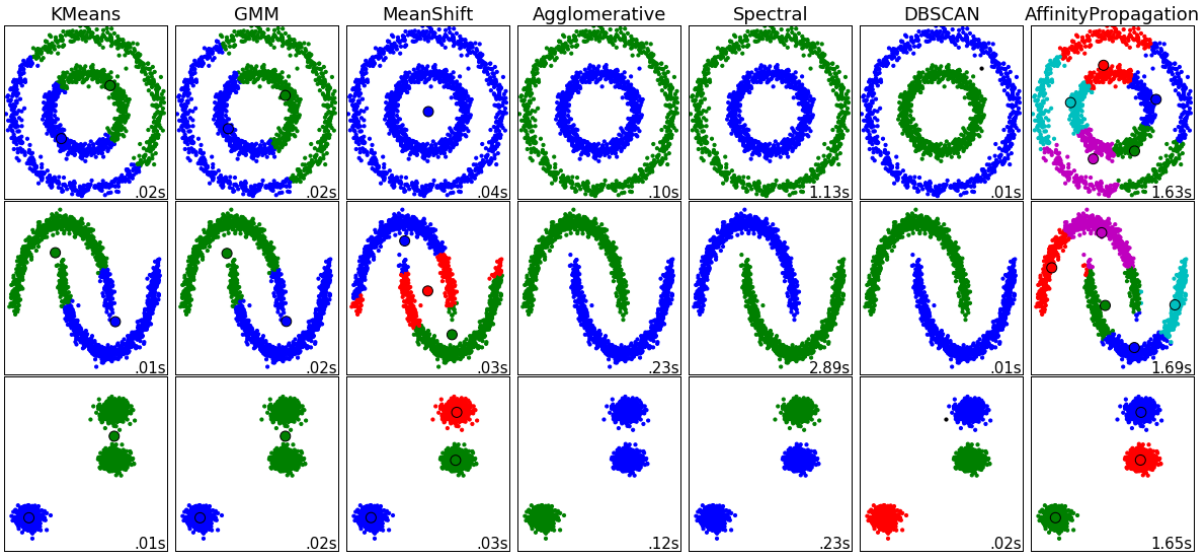


Figure 2.3: Simple comparison of different unsupervised clustering methods on simulated 2D data. Methods which estimate a representative point/centroids are indicated with a black border. The default number of clusters is set to two. The runtime is shown in lower right. This plot is best viewed in colour and was generated using code modified from the scikit-learn python toolkit [Pedregosa et al., 2011].

Supervised Classification

Classification is the process of learning a mapping from an input feature vector to one of a set of discrete outputs denoting the potential classes. This mapping requires a set of training data where each sample is labelled with its corresponding class. This classifier model can then apply what has been learnt from the training data to assign new data.

The most basic type of classifiers are binary classifiers which learn to fit a decision boundary that separates the negative sample data from the positive sample data. One of the earliest statistical models for separating two populations is that of Fisher [1936] where each population is modelled as a normal distribution with their own mean and co-variance. The optimal boundary separating the two distributions is described with a quadratic equation. For the special case where the co-variance of each distribution are equal this decision boundary degenerates to a linear hyper-plane that is equidistant to each mean commonly referred to as Linear Discriminate Analysis (LDA). When considering only two classes the model learnt can be considered as a Bernoulli distribution $y|x$ since the dependent variable y is binary. This is typically modelled with the probabilistic classifier known as logistic regression [Walker and Duncan, 1967]. I. Guyon et al. [1993] proposed the Support Vector Machine (SVM) classifier that maximises a margin between the samples closest to the decision boundary. This property is reported to give the SVM classifier better generalisation performance to unseen data over other linear classifiers

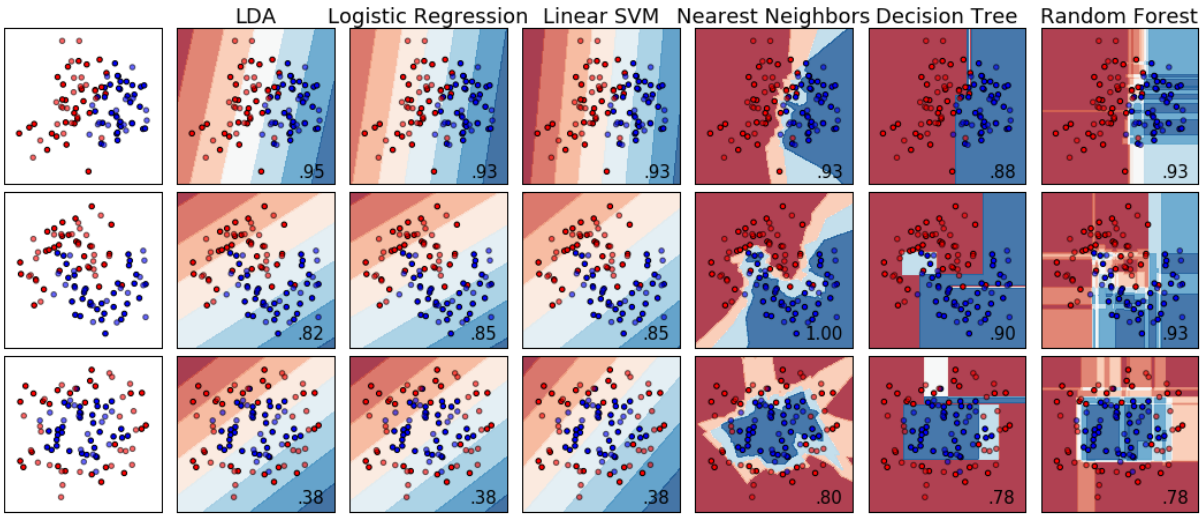


Figure 2.4: Simple comparison of different classifier models on simulated 2D data. The dots represent data samples from two classes while the shaded area indicates the classifier's estimated prediction probability. The classification score is shown in lower right. This plot is best viewed in colour and was generated using code modified from the scikit-learn python toolkit [Pedregosa et al., 2011].

[Vapnik, 1998].

Other types of classifiers naturally handle both multiclass and non-linearly separable data such as k Nearest Neighbour (kNN) [Duda et al., 2001] and Decision Tree [Breiman et al., 1984] classifiers. as the name suggests kNN classifiers estimate the class of a test sample by considering the distances and frequency of labels among its k nearest training samples. kNN however has demanding computational and storage requirements as it needs to store and search the entire training data which define the complex decision boundary. Decision Trees on the other hand are compact and efficient binary tree structure that chooses a variable at each step that best splits the dataset such the final leaf nodes contain mostly items of the same class. The branching nodes are usually learnt by selecting from a random subset of variables the one with the highest Gini impurity [Breiman et al., 1984] or information gain [Quinlan, 1986]. Finally, multiple decision trees can be used in an ensemble known as the random forest [Breiman, 2001b] where the mode of the outputs are used to make the final prediction.

A comparison of these supervised classifiers and their decision boundary are illustrated in Figure 2.4. This comparison also demonstrates that when the data is linearly separable (first row) any of the basic linear classifiers are suitable. As the data becomes less linearly separable (last row) the performance of the linear classifiers drop while the more complex classifiers are hardly effected. When comparing the classification score across the different classifiers, the nearest neighbour based approach appears to perform favourably in all cases, however this

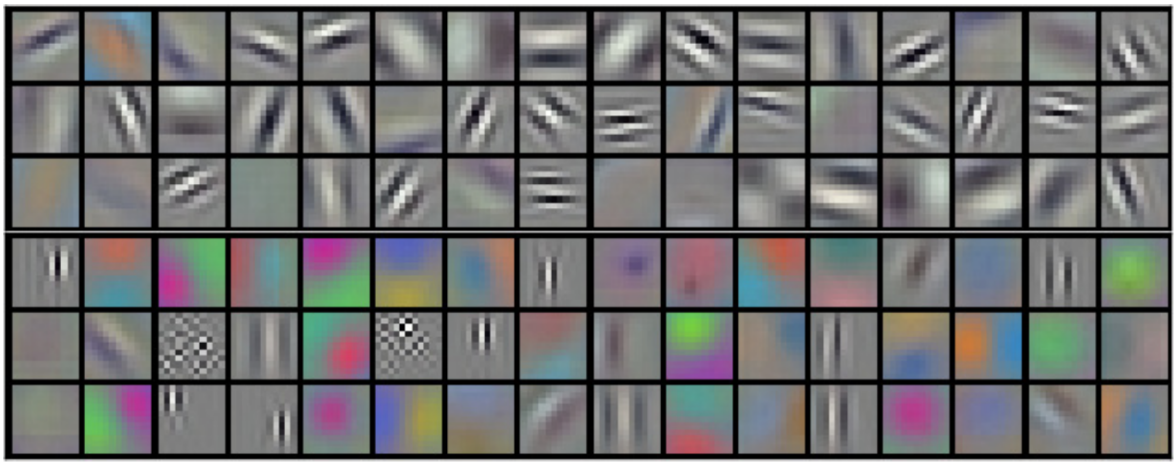


Figure 2.5: Visualisation of the 96 filters with kernel size of 11x11 from the first layer of a DCN which was trained on ImageNet. Reproduced from [Krizhevsky et al., 2012].

outcome is subject to having sufficient training samples covering the expected data distribution to be observed at test time. In practice, if the data is not linearly separable, non-linear transform such as the kernel trick [Hofmann et al., 2008, Mika et al., 1999] can be used to potentially transform the data into a more separable space. However, it is important to note that real world data is typically very high dimensional, particularly raw image data and visual features can be extremely high dimensional making it easier to separate linearly. Finally, these classifier models can be extended to multiclass classification by training multiple classifiers using a one-vs-rest, one-vs-one or using a softmax cross-entropy loss [Bishop, 2006].

Deep Convolutional Network

Over recent years DCN architectures [LeCun et al., 1989] have made an astonishing impact on both the machine learning and computer vision communities [Donahue et al., 2013, Dosovitskiy et al., 2014, Farabet et al., 2013, Krizhevsky et al., 2012, Razavian et al., 2014] particularly in the area of image recognition. A DCN consists of multiple layers, each performing a transformation of their input data in the form of a linear projection followed by a differentiable, non-linear activation function to produce an output response. A set of weights and biases govern the linear projection step in each layer which essentially performs an inner product with the input and the weight parameters. The output responses for each layer forms the input for the following layer in a feed-forward fashion, where the first input is the data and the last represents a score for each class.

The baseline architecture of Krizhevsky et al. [2012] consists of eight layers in total with five being convolutional (conv1-conv5) and three fully-connected (fc6-fc8). In the convolutional layers, the weights only connect to a small receptive field in the input, acting like a 2D filter. Figure 2.5 shows the first layer filters of [Krizhevsky et al., 2012] learnt on real-life images. Note that these filters contain both colour blob filters and texture like filters similar to the methods discussed in 2.1.1. These filters are shifted across the lateral dimensions of the input to perform a 2D convolution. The fully-connected layers on the other hand, remove the spatial structure of the data by flattening the transformed input before applying an inner product with their weights to project the entire input into a new high dimensional space. The intuition of this DCN architecture is that the convolutional layers transform the input into low level visual descriptors representing local object parts, while the fully-connected layers are responsible for the high-level task putting the parts together to classify the entire image [Azizpour et al., 2015].

The [Krizhevsky et al., 2012] architecture consists of around 60 million parameters across all eight layers which were optimised for classification via *deep learning*. Deep learning is the process of first applying a each layer transformation in turn to the input data to produce the network output. The classification error of the network (or loss) is then used to adjust the weights by an amount proportional to the error with respect to the layer input. As each layer is differentiable, the chain-rule enables the error to be propagated back to update the parameters in all layers. This training process is usually performed with Stochastic Gradient Descent (SGD) optimisation. Given the high number of parameters, DCNs are prone to over-fitting the training data. To combat this issue Krizhevsky et al. [2012] employed a regularisation in the form of weight decay and novel model averaging technique known as Dropout [Srivastava et al., 2014]. Additionally, this models ability to generalise to unseen images is particularly due to having an extensive dataset containing 1.3 million labelled images for training [Domingos, 2012]. This has also lead to further improvements in image classification by increasing the number of layers [Szegedy et al., 2015] or both the number of layers and parameters per layer [Simonyan and Zisserman, 2015].

Due to the hierarchical nature of the features learnt in DCNs, the intermediate responses have been shown to be very informative for a range of visual tasks beyond the original training dataset Razavian et al. [2014]. This approach to feature extraction is increasingly being used to replace hand-crafted features such as SIFT in the closely related field of recognition [Ge et al., 2015, Sünderhauf et al., 2015]. These approaches do not require retraining of the DCN

parameters but instead use the discriminative features produced to efficiently lookup a nearest neighbour or train a simple linear SVM, making them applicable to potential online adaptation. Similarly, a process referred to as *fine-tuning* is increasingly being used to speed up DCN training where a pretrained model is reused but the last layer has been replaced to reflect the specific task. Yosinski et al. [2014] showed that fine-tuning networks boosts the generalisation performance as the network retains patterns learnt from a broader range of images.

Datasets and Dataset Bias

Over the years, a number of datasets have been proposed for creating standardised benchmarks for evaluating different classification techniques. However, when the scope of variability in these datasets is small or restricted to a single context, the ability of top performing methods to generalise to real world scenarios is questionable [Pinto et al., 2008]. While the computer vision community aims to increasingly grow the size of annotated image datasets, the data itself may contain biases known as *dataset bias*. Torralba and Efros [2011] showed that across six different online benchmark datasets, the model trained on a dataset's training images was also the best preforming model on that dataset's testing images.

Several computer vision datasets are compiled using the vast amounts of images taken from the internet [Deng et al., 2009, Everingham et al., 2009, Fei-Fei et al., 2007, Torralba et al., 2008] which all share a common source of dataset bias. That is, when a human is in control of the camera when taking the image they typically only take photos of what is interesting, novel or artistic. This is illustrated with one of the most popular and largest image dataset known as ImageNet [Deng et al., 2009] (shown in Figure 2.6). Namely, selection bias is indicated by observing that examples of cars are mostly vintage or race cars. The capture bias is in the form of having the objects centred in the image and covering the majority of the image. This object focused capture bias also frequently results on only a single object visible in most images. The datasets of [Dollár et al., 2009b, Geiger et al., 2013] where not collected using internet search phrases or a hand held camera but rather a camera mounted to a vehicle in urban scenarios. However, [Dollár et al., 2009b] is only annotated for pedestrians and not vehicles while [Geiger et al., 2013] contains annotations for people and vehicles but with significantly fewer annotations.

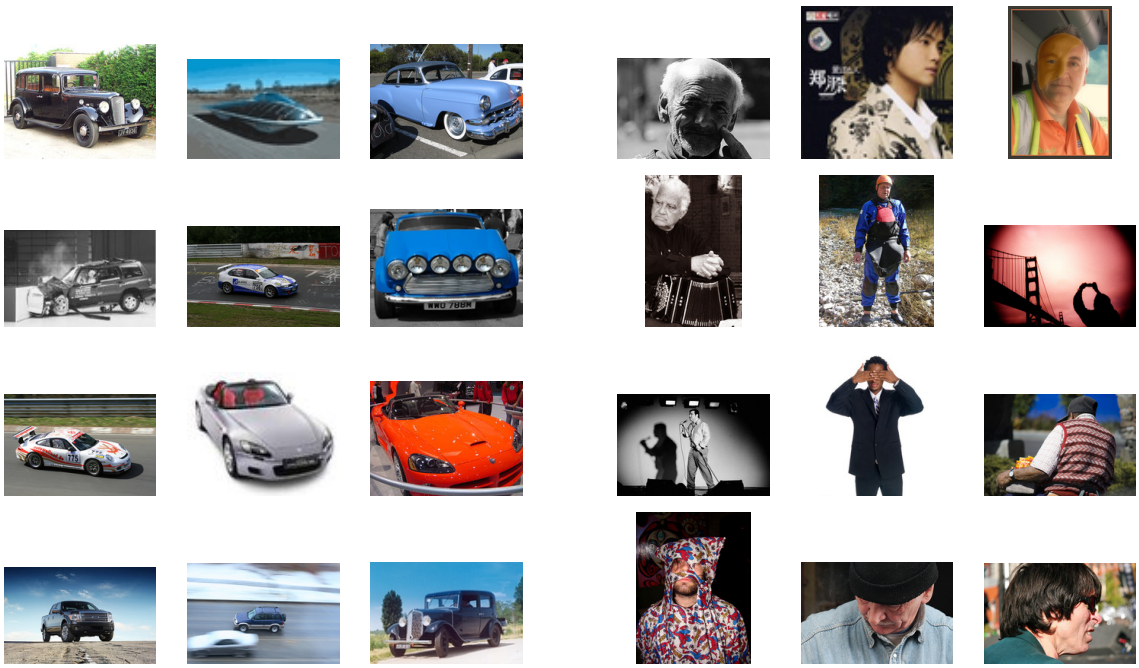


Figure 2.6: A subset of images from [Deng et al., 2009] used in the detection task of the ImageNet Large Scale Visual Recognition Challenge 2014 to illustrate the selection and capture dataset bias.

Self-Supervised Learning

Obtaining a sufficiently large, relevant and unbiased dataset is often impractical, an alternative solution is to bootstrap a discriminative classifier with unlabelled data. This is desirable as there is significantly more unlabelled data available than labelled and relevant data can be collected from the deployed environment directly. However, choosing which side of the decision boundary to position the unlabelled data is equivalent to classifying the data directly.

Literature related to this topic used several terms such as *semi-supervised*, *active learning*, *co-training* and *self-supervised*. Semi-supervised learning typically combines an unsupervised clustering of unlabelled data before using labelled data to assign each cluster a class based on its label distribution [Rosenberg et al., 2005, Zhu, 2008]. Active learning is a related area where a pretrained machine learning based detector is continuously supervised by asking for clarification in the form of labels for samples with low confidence [Grimmett et al., 2015, Triebel et al., 2013]. However, this requires that a human oracle is always in the loop. Co-training replaces the human supervisor with another classifier that is learnt from the different features that were generated from the same source [Blum and Mitchell, 1998]. This however relies on that the features for each classifier are independent from each other to prevent drift,

which is considered a sub-optimal assumption Kalal et al. [2012], Pierce and Cardie [2001].

This thesis focuses on self-supervised learning where a supervisory signal is enforced through exploiting either geometrical or temporal constraints on the objects within the scene. Unlike active learning this does not require a human oracle and compared to co-training it is robust to drift as the constraints are fixed for the life of the learner. Self-supervised learning has shown success in similar applications, particularly in road or free space estimation [Achar et al., 2011, Lookingbill et al., 2007, Zhou and Iagnemma, 2010]. These methods draw upon scene based assumptions such as road driving or river scene structure to gather and label samples for training an appearance based classifier.

2.2 Visual Object Detection

In contrast to recognition which assumes that the images contains one of a known set of object categories, object detection is the task of locating potentially many objects within the image or identifying when no objects are visible. Given a set of input features extracted from the visual data, object detection techniques aim to predict the location of the target object. This section looks at techniques used to locate the objects and also object recognition techniques for when the location is given. Additionally, approaches that consider motion to segment background from foreground are also reviewed as an alternative to appearance based methods.

2.2.1 Locating Objects

Traditionally, object detection is formulated as a classification problem in the well known *sliding window* paradigm (previously discussed in Section 1.1) where the classifier is evaluated over an exhaustive list of positions, scales, and aspect ratios. This exhaustive list can contain millions of sub windows that need to be evaluated sequentially. When using a sophisticated classifier, the computational cost of this approach quickly becomes prohibitive for realtime applications.

To reduce this computational burden many researchers have looked at how to reduce the number of windows evaluated by analysing features that capture scale independence [Benenson et al., 2012, Dollár et al., 2010, Gavrila and Munder, 2006]. Cascaded approaches [Kalal et al., 2010, Viola and Jones, 2001] are frequently used to speed up detection. The classical Viola

and Jones [2001] approach would only compute features incrementally based on the responses previously evaluated simple classifiers within the AdaBoost framework [Freund and Schapire, 1997]. Kalal et al. [2010] used a three stage cascade employing a more sophisticated classifier after the majority of candidates were rejected in the previous stage. Additionally, geometric context of the scene can be exploited to focus compute on the most likely locations and scales to contain pedestrians [Benenson et al., 2012, Gavrila and Munder, 2006].

In recent years, addressing the reduction of windows to consider in detection has formed a sub field of computer vision referred to as detection proposals (a.k.a. objectness, object proposals or region proposals). Detection proposals are considered as a pre-processing step reducing the millions of candidate windows per image down to hundreds or a few thousand. Several object proposal techniques are based on the merging of low level segmentation techniques [Alexe et al., 2012, Carreira and Sminchisescu, 2012, Manen et al., 2013, Rantalankila et al., 2014, Uijlings et al., 2013]. Since these techniques employ image segmentation such as [Felzenszwalb and Huttenlocher, 2004] as a preprocessing step, their utility as a computational reduction technique comes with added overhead.

Other techniques, combine simple features with a sliding window based approach that is designed for speed by taking advantage of efficient data structures such as integral images [Ziming et al., 2014, Zitnick and Dollár, 2014]. Zitnick and Dollár [2014] observed that objects are typically characterised by the number of contours wholly enclosed by a bounding box. Ziming et al. [2014] introduce a binarised normed gradient feature which is efficiently computed on modern computing architectures and used with a simple linear SVM classifier. A detailed comparison of various detection proposal techniques was recently evaluated in [Hosang et al., 2014]. Apart from the computational benefits, detection proposals were recently shown to also improve detection performance [Girshick, 2015].

2.2.2 Object Recognition Models

In supervised detection the types of objects are known a priori, such as pedestrians or cars. These techniques are typically designed for a single image where each location is inspected for the presence of the object of interest. Given a sub-window either selected via a detection proposal technique or with a sliding window, a fixed length descriptor is then extracted for object or background classification. As these methods are based on the object's visual appearance

(colour, texture or shape) in a single frame, they avoid many issues related to motion modelling for moving cameras. In a supervised framework, positive images of the target object and negative background images are presented to a classifier which learns a mapping from the feature space to the label space. The classifier encodes the decision boundary in appearance space that specifies if a newly presented window contains the target object of interest.

One of the first successful object detectors is the Viola-Jones detection framework applied to face detection [Viola and Jones, 2001] and later generalised to detect other object classes [Viola and Jones, 2003]. Their approach used AdaBoost [Freund and Schapire, 1997] to select a set of weak decision tree classifiers based on Haar wavelet features and order the decision responses to reject a hypothesis in a cascade structure – focusing computation on regions most likely to be the object of interest. Dalal and Triggs [2005] harness the expressive power of the HOG descriptor to train a linear SVM classifier which enjoys an order of magnitude reduction in false positives at the same detection rate over the Viola-Jones detector (as reported by Dollár et al. [2009b]). More recently, Zhang et al. [2009] also proposed an AdaBoost architecture using LDA base classifiers with covariance features [Tuzel et al., 2007].

A major limitation of these object recognition based techniques is the requirement of comprehensive labelled training and test sets, which captures high variance in visual appearance caused from viewpoint, deformation and occlusion. For example, Junior et al. [2009] used around 9600 positive and a 10000 negative labelled images for training a pedestrian detector with an additional 9800 images for testing. Moreover, once trained, the boosted classifier cannot adjust to the particular scenario in which it is employed [Javed et al., 2005]. To deal with the viewpoint issue Rybski et al. [2010] train multiple classifiers, one for each 45^o viewing angle of a car. The deformation and occlusion issues have recently been address with part based models which describe an object as a collection of parts or visual words.

To overcome the issues of occlusion and non-rigid objects, several deformable part based models have been proposed [Agarwal et al., 2004, Andriluka et al., 2008, Azizpour and Laptev, 2012, Bouchard and Triggs, 2005, Felzenszwalb et al., 2008, 2010, Leibe et al., 2004]. These methods capture the spatial relationships between different parts. For example, a persons *torso* is positioned below their *head* and between their left and right *arms*. The joint detection of multiple parts and their spatial relationship to each other can be used to improve the detection performance of the higher level object detections [Felzenszwalb et al., 2010].

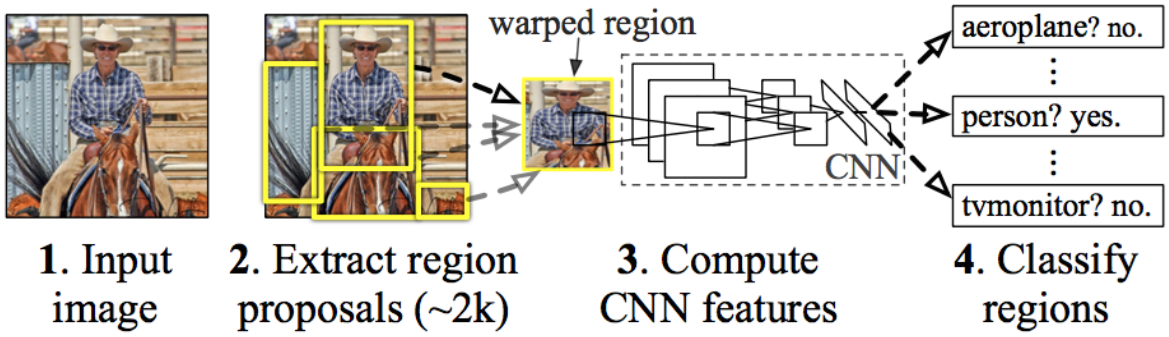


Figure 2.7: An overview of the Region CNN pipeline. Reproduced from [Girshick et al., 2014].

Leibe et al. [2004] pioneered this field with the Implicit Shape Model (ISM) which augments a codebook of local visual appearance models with a spatial probability distribution defining the relative offset of the entire specific object class. Wu and Nevatia [2007] learn multiple human part based classifiers based on edge features before combining the parts into a whole human detector within Maximum a Posteriori (MAP) estimation framework. Felzenszwalb et al. [2008] formalise the Deformable Part Model (DPM) as a hierarchical structure by extracting features at two scales with the course scale representing objects and the finer representing parts. Zhu et al. [2010] extend this framework to a three layer hierarchy and show that the deeper structure outperforms the two layer DPM and also demonstrating that simpler part structures are sufficient to obtain strong results.

Another critical issue for object detection is computational cost which limits their value for realtime detection from mobile platforms. The detector with the best performance run-time trade-off is Dollár et al. [2010] which replaces multi-scale feature extraction with a set of multi-scaled classifiers. Other techniques limit the search to specific Regions of Interest (ROI) in the image by utilising stereovision with the ground plane assumption [Bajracharya et al., 2009, Benenson et al., 2012, Gavrila and Munder, 2006, Howard et al., 2007].

Given the recent success of the DCN based image classification, they have also been applied to object detection. Sermanet et al. [2014] apply the convolutional layers to multiple scales of a test image and average the detection scores. Girshick et al. [2014] proposed the R-CNN method that combined the detection proposals of [Uijlings et al., 2013] with the DCN image classifier of [Krizhevsky et al., 2012] (shown in Figure 2.7). This framework has become a baseline technique for many recent visual object detection frameworks [Girshick, 2015, He et al., 2014]. Similarly, Oquab et al. [2014] use the [Krizhevsky et al., 2012] as the object classifier in a sliding window fashion to localise objects and generate a heat map over the image for a set

of semantic classes. They extract patches with various sized windows before upsampling the window back to 224x244 for the input to [Krizhevsky et al., 2012]. This approach has a number of inefficiencies, each scaled patch is run through the network end-to-end, while the up-sampling also generates extra data for small patches without added information.

Farabet et al. [2013] uses a more efficient approach where subsampled versions of the input image is applied and the sliding window is performed intrinsically by the convolutional layers of their model. To fuse the information from multiple scales they then upsample the resulting DCN feature maps to the original input size and concatenate the features before classification. He et al. [2014] attempt to handle scale by directly concatenating different pooling window sizes in the final convolutional layer within a spatial pyramid framework. Additionally, this allows the entire input image to be passed through the convolutional layers just once with a sliding window based approach used to pass the feature vector through the fully connected layers for final classification at multiple locations, scales and aspect ratios. Similarly, Girshick [2015] later proposed a method which also carries out the convolution over the entire image and combines the regions produced by a detection proposal method to perform the final classification.

2.2.3 Background Modelling

For moving object detection, motion segmentation techniques can also be applied to identify moving objects without restriction of a specific semantic class or prior knowledge of their visual appearance. Contrary to recognising previously modelled objects of interest, Grimson et al. [1998] showed that detection can be achieved by first modelling the background scene and then identifying regions in the image which do not fit this model. As this approach focuses on modelling the background, the variations in the object appearance from changing view point, occlusion or deformation become irrelevant as any object which contravenes with the background model will be detected. Their approach works by continuously comparing the current frame to a model of the background so that regions where there is a significant difference are highlighted as foreground. This straight forward approach is generally described as *background subtraction*.

The majority of the literature [Grimson et al., 1998, Heikkilä; et al., 2004, Heikkilä and Pietikäinen, 2006, Piccardi, 2004, Singh et al., 2009] on background subtraction is concerned

with detecting novel foreground objects from a stationary video camera as surveillance applications has previously been the key driver for research in this area. In such applications this is a mature technology with hundreds of techniques proposed [Bouwmans, 2011], yet exclusively for a static camera. While some methods [Borges, 2013, Elgammal et al., 2002, Reddy et al., 2013, Zivkovic and van der Heijden, 2006] can handle gentle dynamic situations such as waving trees or slight oscillation of the camera due to wind, they are not suited for the level of dynamic scenes experienced from a camera mounted to a mobile platform.

Techniques designed to handle camera motion generally entails the computation of optical flow followed by either ego-motion estimation or motion clustering on the distribution of the extracted flow vectors. Ego-motion is the estimation of the camera's relative motion to the rigid static environment and has long been used in Structure from Motion (SfM). The effects of camera motion can be observed using the optical flow and estimated using geometric constraints such as the epipolar or trifocal constraint [Kim et al., 2012]. As the optical flow generated by independently moving objects do not fit the general scene motion an outlier robust estimator such as RANSAC [Fischler and Bolles, 1981] or LMedS [Rousseeuw, 1984] are generally employed to estimate the camera motion model [Kitt et al., 2010]. Guizilini and Ramos [2013] use the output of a RANSAC based approach [Hartley, 1997] to select candidate points for the online learning of a non-parametric model based on Gaussian process classification to better identify static and dynamic parts of the scene. Similarly, Sheikh et al. [2009] proposed to model the trajectory basis of salient background points which are identified using RANSAC before learning a pixel-wise segmentation within a Markov Random Field (MRF) framework. However this method is both computationally slow and also requires up to 30 frames for estimating the trajectory basis.

Beyond the background/foreground segmentation problem few works have employed clustering techniques to segregate multiple foreground objects in the scene. MacLean et al. [1994] describe the scene motion as a GMM using motion features and utilise the EM algorithm to estimate the mixture portions of data samples. However, they assume the number of moving objects is known. Lenz et al. [2011] detect any number of objects by connecting sparse interest points using Delaunay Triangulation and then cluster the points by removing edges with dissimilar stereo and motion features.

Table 2.1: Detection response confusion matrix.

	Actual object	No object
Detection	*TP*	*FP*
No Detection	*FN*	*TN*

2.2.4 Detection Evaluation Metrics

In a binary decision problem (*i.e.* the target object is presented or not), a classifier assigned either a positive or negative label to each presented data sample. When considering the performance of such a system, two types of errors are possible. **Type I** error or false positive (*FP*) occurs when a detection does not correspond to any actual object, while a **Type II** error or a false negative (*FN*) is the failure to detect the true presence of the target object. Table 2.1 illustrates the potential combinations of the system responses and the ground truth presence of an object.

A dataset with known true labels are used to evaluate the performance of a detection system or classifier algorithm. The true labels are generally acquired through manual hand labelling each sample by a human expert in the field. The predicted response from the classifier for all samples in the evaluation set are compared to the true labels. Given the confusion matrix in Table 2.1 the true positive rate (TPR) and false positive rate (FPR) metrics are defined as follows:

$$TPR = \frac{|TP|}{|Total\ Actual\ Positives|} = \frac{|TP|}{|TP| + |FN|},$$
$$FPR = \frac{|FP|}{|Total\ Actual\ Negatives|} = \frac{|FP|}{|FP| + |TN|}.$$

The TPR measures the fraction of correctly classified positive samples (from total actual positives), while the FPR measures the fraction of negative samples that are misclassified as positive (from total actual negatives). Therefore, a perfect classifier should have 100% TPR and 0% FPR. Note that these metrics work in conjunction rather than isolation as it is trivial to score perfectly on either individually at the cost of the other. For example if a classifier was to always return a positive response then all the actual positive samples could be correctly identified resulting in $TPR = 100\%$, while all actual negative samples are misclassified as positive resulting in $FPR = 100\%$. Similarly both metrics would be 0% if the classifier always

returned negative.

The performance of a given classifier in terms of TPR and FPR is generally illustratively expressed using the Receiver Operator Characteristic (ROC) space. This space is defined as *xy*-axis plot with FPR on the *x*-axis and TPR on the *y*-axis. If a classifier outputs a continuous score (i.e. probability) then this value can be thresholded at various levels to generate a curve in the ROC space. This enables the reduction of the two metrics into a single performance score by setting a desired TPR or FPR and measuring the performance on the other metric. Fixing the TPR or FPR is generally application specific, for example if an obstacle detector formed part of a driver assistance system then a low FPR is desirable, while if used as the primary sensor in an autonomous system then high TPR is desired. Alternatively, the overall performance can be evaluated by considering the total area under the ROC curve.

A common alternative for quantitatively evaluating the performance of a binary detector classifier is to use the *precision* and *recall* metrics. In this metric space it is desirable to maximise both metrics as precision and recall are defined as follows:

$$Precision = \frac{|TP|}{|TP| + |FP|} ,$$
$$Recall = \frac{|TP|}{|TP| + |FN|} .$$

Here recall is the same as TPR, while precision is the fraction of positives correctly classified out of the total number of predictions made. Precision and recall are generally favoured over ROC metrics when evaluated on a highly skewed dataset. A detailed view of the relationship between these two methods are provided by Davis and Goadrich [2006]. F-measure or F-score is used to measure the overall performance of a classifier by evaluating the harmonic mean of precision and recall as:

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall} ,$$

where β is a weight used to emphasise either precision ($\beta > 1$) or recall ($\beta < 1$). Typically the $F1$ score is used to evaluate a detection system where equal importance is given to both precision and recall. If the detection method is capable of estimating the likelihood or a confidence score for each detection, the precision and recall can be used to generate a curve

by adjusting the detection threshold which can also be used to find the appropriate balance between precision and recall. Moreover, in the multiple class setting, Mean Average Precision (mAP) is often used to aggregate the score by computing the mean area under the precision-recall curve for each class. Finally, in the case of high recall, the log-average miss rate has become popular [Dollár et al., 2009a,b].

2.3 Visual Object Tracking

Tracking is an important topic in computer vision and it has been studied for several decades. Despite the extensive research in this area, issues arising from varying illumination, changing view point, partial occlusion and object deformation along with complex object and camera motion rankle the field. Several publications survey the significant and recent contributions to this art [Jalal, 2012, Li et al., 2013, Wu et al., 2013, Yang et al., 2011, Yilmaz et al., 2006]. This section describes the key concepts behind the most notable (in addressing the aforementioned challenges) contributions from the single target visual tracking literature before exploring the multi-target extensions.

2.3.1 Single Object Tracking

A visual tracker is typically initialised with an observation of the object of interest in a single frame and the tracker is responsible for maintaining knowledge of the location and extent of the object in subsequent frames. The initial observation is generally represented as a bounding box containing the object of interest selected either by a user or one of the automatic detection methods presented in Section 2.2. The main functional blocks for constructing a visual tracker are: motion estimators, object and context description models, decision mechanism (i.e. classifier) and more recently a model adaptation strategy (i.e. online learning).

Recent surveys of the most notable techniques in visual tracking were published in [Wu et al., 2013, Yang and Wang, 2011] with [Salti et al., 2012] showing performance comparisons under different conditions. While the majority of the methods in these surveys are for single target tracking they offer valuable insights into effectively updating object appearance models online. Furthermore, state-of-art visual trackers are designed with the following considerations [Yang et al., 2011]:

Robustness to clutter, occlusion, changes in illumination and complex motion.

Adaptive to change in both context and object appearance.

Computational Efficiency enabling real-time processing of live video streams.

These considerations are addressed in the choice of the algorithms used in the main functional blocks of a visual tracker.

Bradski [1998] proposed an adapted version of the mean-shift [Fukunaga and Hostetler, 1975] algorithm by encoding the hue value of a target object as a histogram which is continuously updated with each frame. More recently, He et al. [2013] propose a novel locality sensitive histogram where floating point values is added to multiple spatially local bins which exponentially decays based on distance to pixel location. This alteration has shown significant improvement in visual tracking.

Increasingly, machine learning is being employed to incrementally train an appearance based detector to capture the evolving appearance of the specific object instance over time. [Kalal et al., 2012] explicitly use the predicted trajectory to identify miss detections and similarly spatial structure is used to identify false positives. Any identified errors in the detector output are used to correct the detector through online learning. Methods based on low dimensional subspaces [Ross et al., 2007] or structured SVMs [Hare et al., 2011] have also been used to incrementally update appearance models for the tracked object online. While these approaches have the desired property of learning online to improve tracking over time, the constraints used to identify false detection in [Kalal et al., 2012] and the structured output of [Hare et al., 2011] do not extend to multiple objects. Over the past few years a number of tracking challenges [Kristan et al., 2013, 2014, 2015, 2016] were established to better quantify and accelerating progress in visual tracking. However these challenges focus only on the case of single object tracking for any arbitrary class, while this thesis is aimed more towards multiple object tracking.

2.3.2 Multiple Object Tracking

This section summarises studies directly related to the work presented in this thesis; particularly vision based multiple object tracking or MOT. One of the most difficult components of MOT, is in combining frame-by-frame detections to estimate the most likely trajectories of an unknown number of targets, including their entrances and departures to and from the scene [Berclaz

et al., 2011, Maggio and Cavallaro, 2009]. Classical approaches for target tracking typically combine a data association framework with an optimal state estimator to evaluate the trajectories of different objects and predict their state ahead of time. Alternative methods generally use sophisticated learning algorithms for *tracking-by-detection* utilising rich visual representations of the objects to solve the data association problem [Jalal, 2012].

Data Association

Multi-object tracking can be achieved by detecting objects in individual frames and then linking detections across frames. Such an approach can be made very robust to the occasional detection failure: If an object is not detected in a frame but is in previous and following ones, a correct trajectory will nevertheless be produced. By contrast, a false-positive detection in a few frames should be ignored. However, when dealing with a multiple target problem, the data association step results in a difficult optimisation problem in the space of all possible families of trajectories. Greedy search or sampling based methods outlined in this section address this problem finding a solution to the global optimum.

When multiple targets are to be tracked, assigning the appropriate measurement to the corresponding object trajectory can become a challenging task known as *data association*. Traditionally this has been solved using Multiple Hypothesis Tracking (MHT) [Cox and Hingorani, 1996, Reid, 1979] or the Joint Probabilistic Data Association Filters (JPDAF) [Bar-Shalom, 1987, Schulz et al., 2003], which both delay making hard decisions while objects are in close proximity. In their pure form, the combinatorial complexity of these approaches is exponential in the number of tracked objects making them infeasible for real-time applications in highly dynamic environments with many targets. However, by incorporating appearance based heuristics in the assignment cost [Kim et al., 2015] and appropriate approximations to the optimisation problem [Rezatofighi et al., 2015], it was recently shown that both MHT and JPDAF remain competitive with only a fraction of the previous computational requirements.

When considering only one-to-one correspondences modelled as bipartite graph matching, globally optimal solutions such as the Hungarian algorithm [Kuhn, 1955] can be used [Huang et al., 2008, Perera et al., 2006]. Approximate methods such as Markov Chain Monte Carlo Data Association (MCMCDA) have showed promising results in tracking a varying number of objects [Oh et al., 2004, Yu et al., 2007] while tolerating miss detections and false positives. Ge

and Collins [2008] propose a hierarchical Bayesian model to infer both the optimal parameters and tracklet partitions from unlabelled data within the MCMCDA framework. Recently, Zamir et al. [2012] incorporate the whole temporal span to solve the data association problem for one object at a time by modelling the inter-frame associations of a single object as a generalized minimum clique graph.

Motion Prediction Estimation

When assigning detections to existing tracked objects within a Dynamic Bayesian Network (DBN) framework (detailed in Appendix A), it is beneficial to consider the predicted state of individual targets using a recursive Bayesian estimator. Possibly the most well known estimator is the Kalman Filter [Kalman, 1960], which estimates the true state of a linear system from a series of observations which contain Gaussian noise and a known motion model. Both the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) [Julier and Uhlmann, 2004] variants can be applied to non-linear motion and measurement models by linearising around the current estimate or sampling around the expected mean to estimate the true mean and covariance. These methods rely on a Markov assumption and carry the associated danger of drifting away from the correct target. Particle filters [Gordon et al., 1993, Isard and Blake, 1998] can handle non-Gaussian noise by using a set of '*particles*' to simulate random perturbations of a state governed by the motion model. These predictive filters are generally applied to estimate the state of each individual object independently. However, as the particle filter has the ability to deal with multi-modal distributions, multiple objects as well as multiple hypotheses can easily be tracked simultaneously [Khan et al., 2005, Koller-Meier and Ade, 2001].

Ding et al. [2008] showed that it is possible to use dynamics to compare tracks and disambiguate between targets without assuming a motion model a priori. Recently, Dicle et al. [2013] model the underlying dynamics of each moving objects with linear auto-regressors and formulate a generalised linear assignment framework that merges detections with the least combined motion complexity. These predicted proximity and motion affinity measures are considered complementary to appearance similarity based approaches to data association.

2.3.3 Alternative Visual Tracking

Due to the recent improvement of object detection algorithms many visual trackers adopt a *tracking-by-detection* paradigm. Aeschliman et al. [2010] showed that segmentation plays a critical role in robustness and performance of tracking, particularly for tracking multiple overlapping targets. Similarly, Chen and Corso [2010] combine both appearance and optical flow for propagating labels between frames in a video sequence.

Wu and Nevatia [2007] directly associate detections using information from combining part based detectors and resorting to a mean-shift tracker when no data association is found. Zhang and van der Maaten [2013] incorporate spatial constraints to preserve the scene structure between frame via a pictorial-structures framework. Felzenszwalb et al. [2010], jointly training individual object classifiers and updating the structural constants with online SVM learning. Part-based approaches are prone to over-fitting due to the flexibility of the model. Yao et al. [2013] address this problem using a two stage training framework comprised of part tracking step followed by the estimation of object and part correlation parameters.

The problem of multi-target tracking can also be decomposed as both a discrete and continuous optimisation problem Andriyenko et al. [2012], Milan et al. [2013]. Assigning detections to either new/existing tracklets or identify as false alarm is intrinsically in the discrete domain while optimal estimation of the target states (such as position, size, and velocity) is inherently a continuous state space estimation problem. Andriyenko et al. [2012] alternate between fitting piecewise polynomial trajectory models to target hypotheses and updating the data association taking into account global trajectory and label costs. Milan et al. [2013] introduce a mixed discrete-continuous Conditional Random Fields (CRF) representation to simultaneously evaluate the data association and trajectory estimate while also imposing two physical constraints. The first ensures that two continuous trajectories should not overlap in both space and time. The second is a mutual exclusion constraint which generally defines that only a single detection should be associated with a single track hypothesis. Together, these constraints result in plausible trajectories.

2.3.4 Tracking Evaluation Metrics

Bernardin and Stiefelhagen [2008] formulate a pair of metrics measuring the performance of

multiple object trackers that focuses on the precision in object location estimation, accuracy in recognising object configuration and consistency in propagating object labels over time. The Multiple Object Tracking Accuracy (MOTA) combines all false positives, false negatives, and label switch errors into a single number, while Multiple Object Tracking Precision (MOTP) measures the average displacement between the ground truth position and the tracker output. As these metrics combine the multiple label association errors into a single number, with a separate number for position precision they are widely used for evaluating the performance of multi-target tracking systems [Jalal, 2012, Milan et al., 2013].

The main metric used to compare trackers would have to be the MOTA metric as it combines three common sources of error in one measure. The MOTA is computed as:

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSw_t)}{\sum_t GT_t} \tag{2.1}$$

where t is the frame index, FN_t and FP_t are detection errors as described in 2.1 while GT_t is the total number of ground truth object in frame t. The $IDSw_t$ is any error in switching the identity of a target object in frame t. Note the MOTA is commonly represented as a percentage $(-\inf, 100]$, which can also be negative in the cases where the number of errors exceed the number of ground truth objects.

The MOTA score defines a detection with a 50% overlap criterion which does not capture the localisation precision of the tracker. For a more detailed measure of how well a tracker locates each target the MOTP is used:

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} \tag{2.2}$$

where $d_{t,i}$ is the bounding box overlap with the assigned true object i and c_t is the number of matched objects in frame t.

Another set of common tracking metrics that capture track consistency are the Mostly Tracked (MT) and Mostly Lost (not tracked) (ML) quality measures [Li et al., 2009, Wu and Nevatia, 2006]. The MT measures the portion of true objects that are tracked for at least 80% of their lifespan. While the ML denotes the portion of true objects that are recovered for less than 20% of their lifespan. It is desirable for a tracker to have high MT and low ML.

2.4 Summary and Implications

Detection and tracking are two well studied research topics and both are increasingly making use of model representations based on machine learning. However, the majority of methods employ supervised classifiers as the appearance model which are typically trained in a batch process. Beyond requiring significant amounts of training data, prior investigations showed that learning models in this way is biased to the original source of the data (Section 2.2.2). This motivates the main focus of this thesis which is the need to gather samples for adapting appearance models during deployment.

The following observations and gaps identified in the literature provide an opportunity to resolve the first research question: "*How to discover objects which may have appearance characteristics unknown by the detector?* "

- Motion based detection methods discussed in Section 2.2.3 use a static camera to model the background and segment moving foreground objects. This appearance model-free approach does not require pretraining and is able to identify previously unseen moving objects. However these approaches rely on various forms of background subtraction making them unsuitable for cameras with significant ego-motion as experienced on a vehicle.
- The recent method of Guizilini and Ramos [2013] used epipolar constraints to account for camera ego-motion and directly train a classifier to learn the appearance of dynamic vs static parts of the scene. While this approach was able to learn during deployment, it was limited to binary classification as opposed to segmenting individual objects.

From the literature, the following observations can be made towards addressing the second research question: "*How to adapt an appearance based detector for deployment in new and novel environments with different background characteristics to the training data?* "

- A useful characteristic of the background subtraction methods is that they would adapt to the deployed environment making them robust to distracting novel background patterns. However, these techniques are sensitive to camera motion preventing them from modelling background appearances beyond a single static scene. To overcome this problem we seek an appearance representation which is not anchored to a specific location in the image.

- Detection proposals are aimed towards selecting likely object candidates to focus attention for object recognition. Some of these techniques were found to consistently identify candidate regions [Hosang et al., 2014] in a class agnostic manor. As these methods shift the attention to be centred onto regions with general appearances of an object, it should be possible to learn a general appearance model for background distractors while also improving efficiency over a sliding window approach.
- Supervised discriminative methods such as deep convolutional networks reviewed in Section 2.1.3 provide a flexible framework that can be retrained for different recognition tasks if an annotated dataset is available. Additionally, their intermediate features are shown to encode multiple cues such as colour, texture and shape that could prove beneficial to modelling the background.

The following observations and gaps identified in the tracking literature provide an opportunity to resolve the third research question: "*How to adapt the assignment cost using data collected at deployment?* "

- Online learning has been used for single object tracking (reviewed in Section 2.3.1) where a detector is trained to identify a single instance of an object [Kalal et al., 2012]. However, the constraints used to gather training samples during deployment assume only a single object of interest exists in the image making it non-trivial to extend to MOT problems.
- Geometric constraints applied to self-supervised scene segmentation in Section 2.1.3 provide a level of robustness to model drift. While many of these constraints rely on specific knowledge of the scene structure, some of the constraints used in visual object tracking (e.g. mutual exclusion Milan et al. [2013] in Section 2.3.2) are generally applicable and could be used as a source of self-supervision to adapt an appearance based tracker at deployment.

The following chapters address these gaps identified in the literature review. In particular, Chapter 3 addresses the first research question by exploring a combination of motion based detection with online learning to continuously discover new objects and update a classifier during deployment. Later in Chapter 4, recent developments in DCNs are used to address the second research question by extracting DCN features for background modelling within both unsupervised and semi-supervised frameworks. Finally, Chapter 5 takes the intuition that constraints can be used to guide self-supervised techniques and is applied to object tracking for

addressing the third research question. Together, these methods utilise unsupervised and self-supervised techniques to facilitate the adaptation of detection and tracking during deployment.

Chapter 3

Discovering Unknown Moving Objects

The literature review in the previous chapter identified motion based techniques as a class independent tool for detecting moving objects. The first published paper in this thesis [Bewley et al., 2014] takes inspiration from two of the few techniques presented in the literature that are suited for a moving camera. Particularly, the use of epipolar constraints as a supervisory signal for learning a classifier [Guizilini and Ramos, 2013] is explored along with motion based clustering to separate the dynamic components into individual objects as inspired by [Lenz et al., 2011]. This paper combines these ideas together to learn new objects as they appear in video frames. Additionally, the paper proposes a technique for associating motion clusters to known object instances which is then used to improve the classifier within a self-supervised framework. This framework enables the vision system to identify new objects with potentially unknown visual appearance effectively addressing the first research question of: *How to discover objects which may have appearance characteristics unknown by the detector?*

This chapter presents a motion based detector that identifies moving objects by analysing the displacement of features matched to the previous frame which are clustered using DBSCAN [Ester et al., 1996], while the classifier compares the appearance encoded in the clustered features to stored appearance models. This complementary arrangement of motion clustering and appearance classification, enables the detection of both known objects and unknown objects that are currently moving in the scene. The inter-frame motion can also bootstrap tracking of individual objects to both reduce incorrect detections and estimate the velocity of the detected objects. By tracking object detections over multiple frames, the single frame detected motion regions can be aggregated to eliminate false positives, identify misdetections and captures the

changes in an objects appearance over time.

The proposed method was quantitatively evaluated and compared to the earlier work of [Guizilini and Ramos, 2013] by treating all objects as a single instance. The multi-instance performance was evaluated on a range of datasets which differ in environmental context, including video from the benchmark [Geiger et al., 2012] and a novel mining sequence. These experiments demonstrated the systems ability to learn across multiple environments without relying on any pretrained appearance models.

RESEARCH STUDENTS CENTRE
Examination Enquiries: 07 3138 1839
Email: research.examination@qut.edu.au

Statement of Contribution of Co-Authors for Thesis by Published Paper

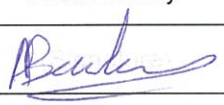
The following is the format for the required declaration provided at the start of any thesis chapter which includes a co-authored publication.

The authors listed below have certified* that:

1. they meet the criteria for authorship in that they have participated in the conception, execution, or interpretation, of at least that part of the publication in their field of expertise;
2. they take public responsibility for their part of the publication, except for the responsible author who accepts overall responsibility for the publication;
3. there are no other authors of the publication according to these criteria;
4. potential conflicts of interest have been disclosed to (a) granting bodies, (b) the editor or publisher of journals or other publications, and (c) the head of the responsible academic unit, and
5. they agree to the use of the publication in the student's thesis and its publication on the QUT ePrints database consistent with any limitations set by publisher requirements.

In the case of this chapter:

Publication title and date of publication or status:
"Online Self-Supervised Multi-Instance Segmentation of Dynamic Objects" (published May, 2014)

Contributor	Statement of contribution
Alex Bewley  8th April 2016	Wrote the manuscript, designed and conducted experiments and performed data analysis
Vitor Guizilini	Provided input into experimental design, scope of paper and proofreading/editing
Fabio Ramos	Provided input into experimental design, scope of paper and proofreading/editing
Ben Upcroft	Provided input into experimental design, scope of paper and proofreading/editing

Principal Supervisor Confirmation

I have sighted email or other correspondence from all Co-authors confirming their certifying authorship.

Ben Upcroft		15/4/16
Name	Signature	Date

Online Self-Supervised Multi-Instance Segmentation of Dynamic Objects

Alex Bewley[1], Vitor Guizilini[2], Fabio Ramos[2] and Ben Upcroft[1]

***Abstract*— This paper presents a method for the continuous segmentation of dynamic objects using only a vehicle mounted monocular camera without any prior knowledge of the object's appearance. Prior work in online static/dynamic segmentation [1] is extended to identify multiple instances of dynamic objects by introducing an unsupervised motion clustering step. These clusters are then used to update a multi-class classifier within a self-supervised framework.**

In contrast to many *tracking-by-detection* based methods, our system is able to detect dynamic objects without any prior knowledge of their visual appearance shape or location. Furthermore, the classifier is used to propagate labels of the same object in previous frames, which facilitates the continuous tracking of individual objects based on motion.

The proposed system is evaluated using recall and false alarm metrics in addition to a new multi-instance labelled dataset to measure the performance of segmenting multiple instances of objects.

Fig. 1: Example detection of multiple dynamic objects discovered using our proposed method that corresponds to the input image sequence shown above. The different colours (hue) in the output image represents the multiple object instances detected, while the intensity denotes the likelihood of the assigned class at each pixel. These objects were detected using only the motion of the scene and not any off-line models describing the visual appearance of the objects.

I. Introduction

This paper addresses the problem of detecting and segmenting multiple dynamic objects simultaneously from a monocular video sequence, where the camera itself is moving within the scene. Motion segmentation remains as one of the fundamental computational challenges and is a critical perceptive capability for several robotic tasks, such as collision avoidance and path planning in dynamic environments. The approach taken in this paper uses a combination of unsupervised motion based clustering methods to supervise a multi-class classifier with training examples collected online.

Much research effort is being expended on object recognition based methods which use various supervised classifiers to train a predictive model off-line with a (preferably) large manually labelled training dataset [2], [3], focusing on the detection of a single class of object [4], [5], [6]. The performance of these methods is highly dependent on having a comprehensive training dataset, which contains sufficient number of labelled examples of objects of interest along with negative examples. It is costly and often impractical to obtain such a training set, where each class is known and completely represented for different view-points and lighting conditions. Instead, we collect training examples online in a self-supervised framework, without any prior knowledge of the object's shape, location or visual appearance.

The work presented here falls within the self-supervised classification category, however we restrict ourselves to the detection of only independently moving objects in the scene. Here we are not concerned with assigning semantic labels such as 'car' or 'human' to image regions. Rather we assume that any dynamic object is an obstacle and needs to be tracked in a dynamic motion planning framework. In order to predict where each dynamic object will likely be located in the future, it is desirable to separate the dynamic pixels into independent groups with similar motion. To achieve this, we go beyond the self-supervised binary classification of [1] to identifying multiple independent motion regions within an image as shown in Fig. 1.

This paper addresses several challenges absent in the binary case. Firstly the input sequence need to be first segmented into multiple spatially consistent motion segments. Each motion segment must be assigned a temporally coherent class label before used to train a multi-class classifier. Finally multi-class classification is inherently more difficult than binary classification, which is further exacerbated by the non-stationary nature of video data.

The paper is organised as follows: In the next section we review relevant literature, followed by a brief overview of the proposed system in section III. In section IV we describe the static segmentation and motion clustering of sparse optical flow. In section V we detail the online process for using the sparse motion clusters to update a non-parametric model enabling temporally consistent inference over the entire image sequence. Section VI shows some results before conclusions and outlook to future work in section VII.

[1]A. Bewley and B. Upcroft are with the School of Electrical Engineering and Computer Science, Queensland University of Technology, Australia {aj.bewley,ben.upcroft}@qut.edu.au

[2]V. Guizilini and F. Ramos are with the School of Information Technologies, The University of Sydney, Australia {vgui2872,fabio.ramos}@sydney.edu.au

II. Relevant Literature

Over the last decade many algorithms have been developed to detect obstacles from a moving platform using vision [7]. These methods often combine object recognition and visual ego-motion estimation with occupancy maps. A tracking-by-detection framework is commonly employed utilising advancements in visual object recognition and can further be improved by utilising 3D tracking from stereo images [8]. However this method only detects pedestrians as it is trained off-line for detecting humans.

The Random Sample Consensus (RANSAC) [9] paradigm has been extended to model multiple motions from monocular vision simultaneously [10], however it is restricted to only fitting rigid objects. Kitt *et al*. use a similar two stage approach (RANSAC with an ensemble of extremely randomised decision trees) [11] however their method requires off-line training with hand-labelled training examples of all likely situations. Another approach is to estimate the full structure-from-motion of the camera with a robust estimation of the ground plane [12].

An alternative to learning appearance models of the target objects is to build statistical models the background appearance and temporal shifts online [13], [14]. Recent advancements in compressive sensing [15], [16] led to new approaches for extracting dynamic objects from monocular video sequences, however these approaches have only been demonstrated for the case of static or nominal camera motion.

The objective of this work is to segment regions of an image sequence corresponding to individual moving objects, observed from a mobile camera, without using offline training or prior knowledge of the location, shape or appearance of the target objects. To robustly model the camera motion we estimate the epipolar geometry of matched points within a RANSAC framework mirroring the initial binary classification step in [1]. We also take inspiration from [17] for clustering different motion regions before modelling the spatial location and colour of dynamic objects in addition to the static background.

III. System Overview

The work presented here builds on the binary dynamic classification work by Guizilini and Ramos [1] by extending it to segment multiple objects. An overview of the proposed system is illustrated in Fig. 2 and can be described using the following pipeline:

1) As in [1] sparse optical flow is computed by matching key-points from the current and previous frames as new images are acquired.
2) The optical flow vectors corresponding to the static environment are identified by fitting a motion model within a RANSAC framework to account for outliers.
3) The outliers of the previous step undergo density based clustering to identify independent motion in the scene and remove mismatched key-points.
4) The static and dynamic clusters are then used to incrementally update a multi-class non-parametric kNN model by matching each cluster to either an existing class or a new class.
5) This non-parametric model is then used to infer the object instance for any pixel in the image.

IV. Dynamic Object Discovery

The online detection of dynamic objects begins with an initial segmentation of the sparsely matched key-points from two consecutive frames. These motion segments provide a continuous source of training examples to update the dynamic object classifier described in the next section with new objects and new views of existing objects. The motion clustering happens in two stages, firstly separating static and dynamic points followed by clustering the dynamic points into groups representing consistent motion.

The sparse optical flow is computed by first detecting salient image features using both 'SURF' [18] and 'Good Features to Track' [19]. Using a combination of feature detectors provides reasonable coverage of the image space, including corners, edges and textured areas. The optical flow of these features is extracted by computing the 'BRISK' descriptor [20] of each point and compared to the features detected in the previous frame. This essentially extracts a sparse sampling of the optical flow across the image, providing a basis for motion segmentation described in this section.

A. Static and Dynamic Classification

The process of detecting dynamic objects begins with the binary classification of static and non-static key-points. In this step the global image motion is estimated to account for optical flow generated by the camera motion itself. When the camera is moving, the optical flow from static points in the world are constrained by the epipolar geometry of the two viewpoints.

We classify static points using the epipolar constraint that describes the motion of key-points from two viewpoints using the fundamental matrix [21]. A suitable technique for this is the RANSAC algorithm robust to outliers from the dynamic objects and has been demonstrated as a basis for online classification of static and dynamic points [1]. With sufficient key-points for the static environment, the RANSAC algorithm should elect the fundamental matrix that best represents the camera motion. Therefore, any matched key-point that lies on the epipolar line defined by both the estimated fundamental matrix and the corresponding key-point in the previous frame should belong to a static object. Point matches that do not fit the epipolar geometry are further processed as described in the next section.

B. Dynamic Point Clustering

So far the point correspondences have only been separated into static and dynamic binary classes. We extend this to include multiple instances of moving objects within the image, by grouping similar and separating dissimilar dynamic points based on motion. This grouping task can be formulated as a

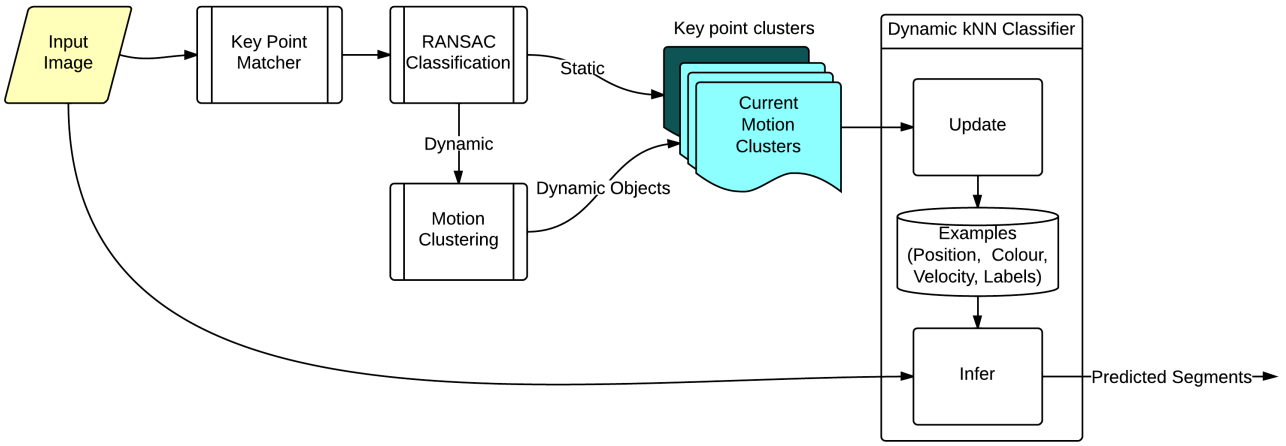


Fig. 2: An overview of the object extraction and learning process used in this paper.

clustering problem for an unknown number of clusters with noise introduced by mismatched key-points.

For characterising the motion of key-points, both the image location and optical flow velocity of the non-static features are used to represent the image motion of the corresponding object. By only considering the non-static features for clustering, we limit the computation required along with providing a larger separation between points. Let the motion feature vector w corresponding to each dynamic key-point describing both the image position(u, v) and inter-frame displacement $(\dot{u}, \dot{v})$ as:

$$w = [u, v, \dot{u}, \dot{v}]^T.$$

To cluster the dynamic points we chose an algorithm which doesn't require the number of clusters to be known *a priori*, suitable for non-rigid objects and can identify outliers which correspond to mismatched features. For this we choose to use the Density Based Spatial Clustering Analysis with Noise (DBSCAN) algorithm [22]. DBSCAN is governed by two parameters: a radius ϵ and the minimum number of points to form a cluster $minPts$ which essentially defines the minimum local density of a cluster.

The motion features for the set of dynamic points ($w_i, i \in DynamicSet$) are used as the input to the DBSCAN algorithm. Using the average key-point density in the image we can evaluate suitable neighbourhood density parameters (ϵ and $minPts$) for grouping features corresponding to the moving objects in the scene. The density is estimated using the average key-point density in the 2D image and assuming that adjacent key-points on the same object will share similar inter-frame displacements as opposed to if they lie on different dynamic objects. For example if two points w_p and w_q on the same object have similar motion *i.e.* $\|w_p - w_q\| \approx \|(u_p, v_p) - (u_q, v_q)\|$ and the main difference lies in their relative image position. If p and q are from different objects or different parts of the image we expect high separation in position and motion space respectively thus reducing the likelihood these points would share a common neighbourhood.

Fig. 3: Sparse optical flow vectors grouped by motion. Static points are shown in black, while the dynamic points are coloured by their cluster assignment. Points with low density in motion space (considered noise) are marked in white.

Using the average matched key-point density ($n/(width \times height)$) of the current frame we can automatically set the ϵ radius to be:

$$\epsilon^2 = \frac{minPts \cdot width \cdot height}{n\pi}, \quad (1)$$

where n is the number of matched key-points including static and dynamic in the frame. The $minPts$ parameter is explicit set to the minimum number of points we expect in a cluster. We found that setting $minPts$ to 10 is a good compromise between the number of false clusters and missing clusters.

Points with fewer than $minPts$ in their ϵ neighbourhood are considered as noise within the DBSCAN framework, unless on the boundary of a dense cluster. This attribute of DBSCAN essentially eliminates non-static points caused by mismatches in optical flow as observed in Fig. 3.

V. Dynamic KNN Classification

While the motion clustering method described in the previous section can identify both individual dynamic objects in the current frame and previously unseen objects, it has limited use for object tracking as it does not maintain any memory of which objects were previously identified. Here we introduce a self-supervised classifier for associating currently detected clusters with previously found objects. Knowledge of previous objects can be maintained for short durations if temporally occluded or when an object is missed due to the

number of matched key-points dropping below the $minPts$ threshold required by DBSCAN.

The k-nearest neighbour (kNN) classifier provides a suitable mechanism for this task, as it's capable of representing complex decision boundaries and naturally supports multi-class classification problems, while being simple to implement [23]. As the name suggests, the k nearest neighbour algorithm assigns a class label to an unlabelled test point by considering the distance to and frequency of labels amongst the k nearest neighbours in the model. This enables the kNN classifier to represent complex and non-Gaussian decision boundaries defined by the representative data. For finding the k nearest neighbours, we use the randomised kd-trees method described in [24] to achieve efficient search time with precision guaranteed in Euclidean space.

The kNN classifier is trained with a set of input feature vectors x_i that describe the image location and colour of the each key-point along with the assigned cluster number k_i. The input feature vector x is defined as:

$$x = [u, v, S \cdot cos(H), S \cdot sin(H), V]^T,$$

where H, S, V are the hue, saturation and intensity value (HSV) of the pixel located at (u, v) in the image. The HSV colour space is chosen over the RGB colour to limit sensitivity of lighting to a single channel.

For clarity, in this section we refer to *clusters* as the output of the unsupervised approach for the current frame and *class labels* as the temporally consistent moving object identifier stored in the kNN classifier. Also note that $k = 0$ represents the static cluster from RANSAC while $k = 1...K$ is a unique identifier for the individual dynamic clusters found using DBSCAN for the current frame.

This classifier is initialised with the initial clusters found in the first pair of frames and then incrementally updated there after. Clusters found in subsequent frames are used to continuously update the kNN non-parametric model allowing it to adapt to the changing dynamics of the scene using a continuous supply of training examples. However, the cluster numbers of the dynamic objects are arbitrary in the unsupervised framework and need to be matched to the class numbers representing previously discovered objects.

In the remainder of this section, we detail the inference method used for predicting new class labels, address the issues of cluster to class association and manage the growth of the model through selective updating and structured forgetting of uninformative points.

A. Inference

A drawback of the standard majority voting classification in this application arises due to the static class being exceptionally more frequent than any dynamic class and essentially dominates the feature space near the decision boundaries. To help alleviate this problem we use a probabilistic soft-max weighting function to evaluate the conditional probability of assigning the label c to test point x as follows:

$$p(c|x, N^k(x)) = argmax_c \left(\frac{\sum_{i \in c \cap N^k(x)} e^{-\|x_i - x\|^2}}{\sum_{i \in N^k(x)} e^{-\|x_i - x\|^2}} \right), \tag{2}$$

where $N^k(x)$ are the kNN points from x in the non-parametric model. This essentially weights the votes from each neighbour by their proximity to the test point.

B. Cluster Association

As subsequent frames provide new key-point examples of dynamic objects, these clusters need to be associated to previously seen objects or assigned to a new object. Evaluating the appropriate and temporally consistent class label for a given cluster is critical for the classifier to continuously build on its knowledge of a specific object. Here the classifier itself is used to predict the class labels of the new training clusters and resolve inconsistencies through information filtering.

Cluster association is achieved through an initial step of inferring the class label of each supplied training point and comparing its overlap ratio of class labels and cluster numbers. This is equivalent to assigning each cluster to the class with the highest Jaccard similarity coefficient [25]. Given all key-points of cluster K and the set of key-points classified with label C (denoting an existing object in the non-parametric model), the Jaccard coefficient is computed as:

$$J(K, C) = \frac{|K \cap C|}{|K \cup C|}. \tag{3}$$

The clustering result of the current frame is compared to the output of the dynamic kNN classifier to evaluate new objects and update existing objects. Given the cluster labels and predicted class assignments for each key-point provided in the current frame's training set, we take a greedy approach by remapping the entire cluster to the class number with the highest Jaccard coefficient; a point p_{ck} jointly assigned to class c and cluster k is reassigned to the c^* that has the highest Jaccard coefficient voted by all point in the same cluster.

The discrepancies between the predicted class labels and the consensus/remapped class labels can be further used to identify anomalies in the temporal consistency of the unsupervised clustering and identify clusters representing new/unseen objects. For example, in the top row of Fig. 4, a non-static cluster found corresponding to the entering car on the left, doesn't match any existing dynamic labels signifying the need for a new class label (represented as red in the output image). If this cluster was a false positive, it is expected that the cluster is not temporally coherent and that current and future static points located in the same region of the input space will rectify this scenario in the forgetting step.

C. Updating

Using the assigned labels from the cluster association, kNN learning is as simple as adding an example point to the

Fig. 4: Examples of temporal coherent label output of the proposed method. Top row shows the immediate detection of a car as it enters the scene denoted by the red cluster introduced in the middle frame. The bottom row demonstrates correct label assignments are maintained over multiple frames, even during temporary occlusion of the pedestrian on the right of the image.

non-parametric model. To minimise the unbounded growth rate of the model, new points are filtered before being added to the model. New points are only inserted if the inferred class at the feature location differs from the reassigned cluster label or if the local density of the input point is low.

Additionally, as the state of dynamic objects naturally change over time, it is desirable to reflect this behaviour in the classification process. This has many advantages over a stationary kNN classifier, particularly in regions of occlusion and dis-occlusion (see bottom row of Fig. 4). As we are already computing the sparse optical flow for our learning examples we can re-use this computation to set the temporal state of each individual training sample.

The kNN is updated by evaluating x_{t-1} at each key-point location in the previous image and x_t in the current image using the optical flow correspondence. The temporal partial derivative of each point is approximated as:

$$\delta x/\delta t = x_t - x_{t-1}.$$

After new examples are added to the kNN database the entire non-parametric model is updated using the partial derivatives:

$$x_{t+1} = x_t + \delta x/\delta t.$$

This keeps the learnt input values relevant as points move though the image and gradually change colour as lighting conditions change. At this point, we rebuild the kd-tree structure for efficient kNN inference on the updated point set.

D. Forgetting

As more points are added to the non-parametric model in the learning phase the speed of inference degrades. As the scene evolves over time many points added to the model become irrelevant and overcrowded by points with mixed labels. This has a detrimental effect on the speed of inference as the size of the model continues to grow. Unlimited growth is restricted in a number of ways:

1) Firstly, we actively search for irrelevant data points by performing inference on each point in the model and removing points classified with a different class to their assigned labels. This is a common outlier detection method used for kNN and is analogous to the Gaussian Process feature filtering component of [1].
2) Secondly, as we propagate points in our model, points with non-zero velocity eventually exit the input volume bounded by the image dimensions. This input boundary is evaluated and expanded by checking the minimum and maximum values of each input dimension. Any point in the model propagated outside the boundary by a user selected margin is considered irrelevant and is discarded. The default value of this margin is set to the maximum average velocity defined by the optical flow evaluated in a similar fashion to the boundary itself.
3) Finally, as misclassified points are removed in the first step there is the potential the area would be over represented by points with uniform labels. These points are removed by limiting the maximum density within the non-parametric model with uniform class labels.

VI. Experimental Results

To evaluate the proposed method we first consider its performance in segmenting dynamic from static objects and compare to other online learning methods. Then we measure the performance of continuously segmenting dynamic objects on a multi-instance basis using a hand annotated sequence taken from the KITTI dataset [3].

A. Online static vs dynamic classification

For comparison to other static/dynamic binary classification methods we use the receiver operator characteristic

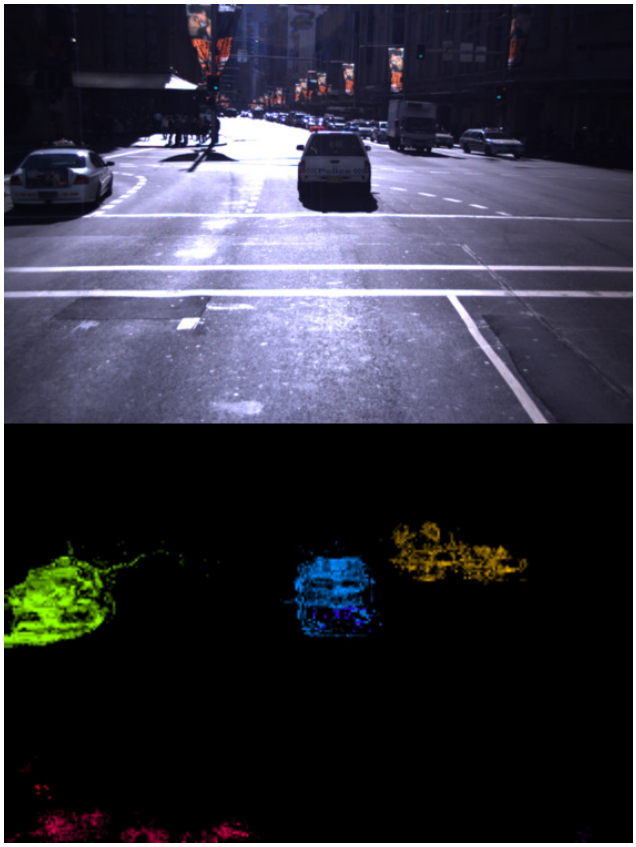


Fig. 5: Detected dynamic objects from the Sydney dataset. In this frame the truck and car on the opposite side of the intersection are considered as a single object as they share similar motion. The red object is a false positive due to the lack of texture on the road. It is expected that this system would be combined with a ground plane estimator to remove these false positives and shadows if deployed specifically for road applications.

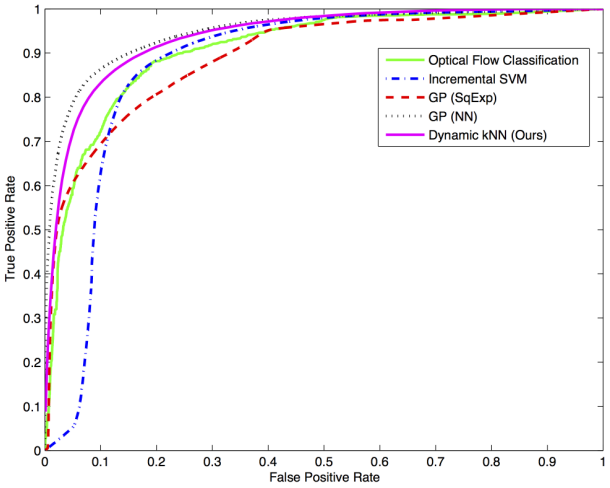


Fig. 6: Receiver Operator Characteristic (ROC) curves over 100 frames from the Sydney dataset and comparison to other online dynamic object segmentation algorithms including: an optical flow based classification of [26], Incremental SVM [27] and the Gaussian Process methods from [1]. Our method outperforms all other techniques except for [1]. However, it must be noted that all these methods are binary classifiers whereas ours includes multiple instances. The effectiveness of multi-instance classification is evaluated in the following experiments.

(ROC) curve, generated by varying the discriminative probability threshold on $1-p(static|x)$ (of equation 2). A larger area under this curve indicates a better overall performance in all threshold levels. This provides a graphical illustration of the true positive rate vs. the false positive rate. To evaluate the ROC curve performance for the proposed system we use the same ground truth dataset of [1], consisting of 100 frames taken from various sections of a 1500 frame dataset taken from a moving vehicle around the city of Sydney. This dataset contains a significant variance in lighting conditions, as the camera moves in and out of building shadows creating high contrast in the visual appearance of dynamic objects (see Fig. 5). With the exception of [1], Fig. 6 shows that our proposed method outperforms several other online techniques including an optical flow based classification of [26]. However, it should be noted that all these other methods are binary classifiers and their *one-vs-all* multi-class equivalent would generally require a single instance classifier for each cluster found as opposed to the kNN classifier which naturally handles multi-class data.

Fig. 7 shows how the accuracy varies in terms of area under the ROC curve for different values of k. The improvement shown by increasing k can be contributed to the smoothing effect of sampling more neighbours, which is beneficial in texture-less areas with a high false positive rate. This gain is ultimately limited as k becomes significantly larger than the number of samples on small and distant object leading to miss detections.

B. Online motion clustering

Since the Sydney dataset was originally produced for static / dynamic classification, it only contains binary labels in the form of an image mask. This paper is mostly concerned with not only detecting a moving object, but also discriminating between each dynamic object. Due to the complexity of labelling for multiple instances of dynamic objects, existing hand labelled dataset with pixel-wise semantic labels are not suited to evaluate the effectiveness of our algorithm. We address this by hand labelling the bike image sequence from [3] with pixel-wise annotations of the individual objects and use the V-measure proposed in [28] as a guide of how well the system can segment these objects. The V-measure is chosen as it combines two desirable aspects of clustering, homogeneity (each cluster contains only members of a single class) and completeness (all members of the same class are contained in a single cluster), without explicitly assigning semantic labels to each cluster. Our online motion clustering system attains a V-measure of 0.24 comprised of a completeness score of 0.31 and homogeneity score of 0.19 as defined in [28].

Since the system doesn't go as far as assigning semantically meaningful labels to each cluster, we evaluate the system's performance in handling multiple instances by visualising the cluster purity and completeness in Fig. 8. This plot shows the true object composition spread across the static cluster (left most column) and the ten largest dynamic

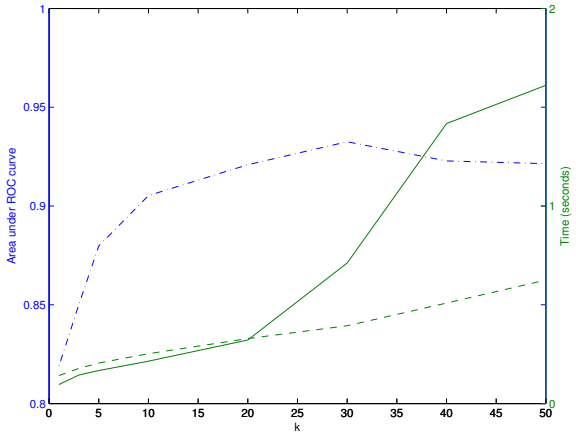


Fig. 7: Comparing performance (dotted dashed blue) along with the update (solid green) and inference (dashed green) computation times for different number of k neighbours on the Sydney dataset. The performance is measured by the area under the ROC curve such that larger values indicate a better overall performance in terms of true positive and false positive rates for all threshold levels. This experiment shows that increasing k improved the overall performance up to $k = 30$, however, this improvement is at the cost of longer computation times.

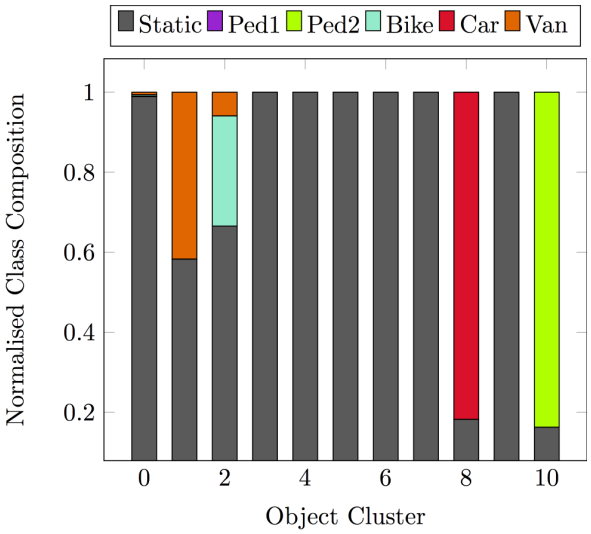


Fig. 8: Visual representation of the class/cluster composition over the KITTI Bike Sequence with individually pixel-wise annotated objects. The columns represent the clusters found by the algorithm with cluster 0 the static cluster and the colours represent the ratio of true class labels for pixels of that cluster.

clusters (in terms of the number of pixels). The colours in each bar represents the true object instance label with the hight of each interval denoting the ratio of overlap between that true class and the object cluster. The bar magnitudes have been normalised to aid visibility of object instances which are only present in a few frames, such as the car and pedestrians.

This plot can be interpreted as: the first pedestrian was not detected, the other true classes are generally contained within a single object cluster. The van and bike classes have a significant portion of the static background within this cluster as the white van shares similar colour to the nearby saturated road surface and for the bike a few key-points around the wheels tend to have the same colour as the road making the colour spread to the texture-less road where there are few true static key-points to correct this. After the system has had time to sufficiently sample the static background the introduction of new dynamic objects such as the car (see top row of Fig. 4) and second pedestrian tend to be more homogeneous. This is largely due to colour change in the localised region input space near the new object tends to be under represented by the static class. We have also run this system on different image sequences to consider it's generality and in environment where there is good contrast between the colour and the environment we observe that the system can accurately segment different dynamic objects (see Fig. 9).

Fig. 9: Moving segments of mining dataset. This shows all moving parts of the scene has been accurately detected. The truck undergoing loading is stationary.

C. Computational cost

A prototype of this algorithm was implemented in C++, making extensive use of the OpenCV 2.4 library, and was deployed on an Intel i7 machine with 8GB RAM. The total time from image acquisition to training and then inferring the presence of dynamic objects across the entire image takes 3.5 seconds, with the breakdown times shown in Table I. This was evaluated by taking the average frame processing time for a typical image sequence from the Sydney [1] and KITTI [3] datasets, with resolution shown in pixels and time in seconds. The two main components which consume the majority of the time are the key-point matching and the kNN training. There are many alternative point trackers which can supply sufficient number of key-points correspondences that when implemented on parallel hardware is capable of real-time performance (e.g. [29]). Furthermore, other forms of efficient indexing, such as Locality Sensitivity Hashing [30], could potentially be a faster choice for nearest neighbour searching in various components of this application, requiring further investigation.

Dataset	Sydney (640 x 480)	KITTI (1242 x 375)
Point Matching	0.35	0.98
RANSAC Fitting	0.09	0.19
Motion Clustering	0.02	0.02
kNN Training	0.63	2.01
Dense kNN Inference	0.11	0.31
Total	1.20	3.51

TABLE I: Average execution times by section (seconds per frame) with $k = 30$.

VII. Conclusion

In this paper we have proposed a method for combining unsupervised motion clustering in a self-supervised framework, for the purpose of segmenting multiple dynamic objects from a monocular image sequence. Furthermore, the segments from each frame are made temporally coherent through the continuous inference, remap and update learning cycle. We have evaluated this approach quantitatively using the original 100 frame dataset of [1], in addition to qualitatively evaluating the multi-class performance on a range of datasets which differ in context. While the binary dynamic segmentation of [1] has better detection performance, we believe this to be an important step towards multiple object tracking and ultimately instance based object characterisation in an unsupervised framework.

In future work, we plan to optimise the individual system components further with respect to run-time and performance. Additionally, the ability to segment whole objects in an unsupervised framework opens many opportunities in object tracking and visual appearance learning.

Acknowledgment

This research has been supported by the Australian Coal Association Research Program (ACARP). The authors would also like to give thanks to Lionell Ott for providing access to a custom cluster assignment plotting tool.

References

[1] V. Guizilini and F. Ramos, "Online self-supervised segmentation of dynamic objects," in *International Conference on Robotics and Automation*, 2013.

[2] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[3] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, Jun. 2012, pp. 3354–3361.

[4] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, Jun. 2008.

[5] X. Wang, T. X. Han, and S. Yan, "An HOG-LBP human detector with partial occlusion handling," in *International Conference on Computer Vision*, no. Iccv. IEEE, Sep. 2009, pp. 32–39.

[6] P. E. Rybski, D. Huber, D. D. Morris, and R. Hoffman, "Visual classification of coarse vehicle orientation using Histogram of Oriented Gradients features," in *2010 IEEE Intelligent Vehicles Symposium*. Ieee, Jun. 2010, pp. 921–928.

[7] S. Wangsiripitak and D. Murray, "Avoiding moving outliers in visual SLAM by tracking moving objects," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, May 2009, pp. 375–380.

[8] A. Ess, B. Leibe, K. Schindler, and L. V. Gool, "Moving obstacle detection in highly dynamic scenes," in *International Conference on Robotics and Automation*. IEEE, 2009.

[9] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[10] K. Ni, H. Jin, and F. Dellaert, "GroupSAC: Efficient consensus in the presence of groupings," in *International Conference on Computer Vision*. IEEE, Sep. 2009, pp. 2193–2200.

[11] B. Kitt, F. Moosmann, and C. Stiller, "Moving on to dynamic environments: Visual odometry using feature classification," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Oct. 2010, pp. 5551–5556.

[12] K. Yamaguchi, T. Kato, and Y. Ninomiya, "Vehicle Ego-Motion Estimation and Moving Object Detection using a Monocular Camera," *International Conference on Pattern Recognition (ICPR)*, pp. 610–613, 2006.

[13] K. Patwardhan, G. Sapiro, and V. Morellas, "Robust foreground detection in video using pixel layers," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 30, no. 4, pp. 746–751, Apr. 2008.

[14] T. Bouwmans, "Recent advanced statistical background modeling for foreground detection: A systematic survey," *Recent Patents on Computer Science*, vol. 4, no. 3, pp. 147–176, 2011.

[15] R. Sivalingam, A. D'Souza, M. Bazakos, and R. Miezianko, "Dictionary learning for robust background modeling," in *International Conference on Robotics and Automation (ICRA)*. IEEE, May 2011, pp. 4234–4239.

[16] C. Lu, J. Shi, and J. Jia, "Online robust dictionary learning," in *Computer Vision and Pattern Recognition*. IEEE, Jun. 2013, pp. 415–422.

[17] D. Almanza-Ojeda, M. Devy, and A. Herbulot, "Active method for mobile object detection from an embedded camera, based on a contrario clustering," *Informatics in Control, Automation and Robotics*, vol. LNEE 89, pp. 267–280, 2011.

[18] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding (CVIU)*, vol. 110, no. 3, pp. 346–359, 2008.

[19] J. Shi and C. Tomasi, "Good features to track," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE Comput. Soc. Press, 1994, pp. 593–600.

[20] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *International Conference on Computer Vision (ICCV)*. IEEE, Nov. 2011, pp. 2548–2555.

[21] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[22] M. Ester, H.-p. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *International Conference on Knowledge Discovery and Data Mining*, 1996.

[23] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. Wiley-Interscience, 2001.

[24] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *International Conference on Computer Vision Theory and Application*. INSTICC Press, 2009, pp. 331–340.

[25] R. Real and J. M. Vargas, "The probabilistic basis of Jaccard's index of similarity," *Systematic Biology*, vol. 45, no. 3, pp. 380–385, 1996.

[26] C. Liu, "Beyond Pixels: Exploring New Representations and Applications for Motion Analysis," Ph.D. dissertation, Massachusetts Institute of Technology, 2009.

[27] C. Diehl and G. Cauwenberghs, "Svm incremental learning, adaptation and optimization," in *Proceedings of the International Joint Conference on Neural Networks*, vol. 4. IEEE, 2003, pp. 2685–2690.

[28] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, no. June, 2007, pp. 410–420.

[29] M. Garrigues and A. Manzanera, "Real time semi-dense point tracking," *Image Analysis and Recognition*, vol. 7324, pp. 245–252, 2012.

[30] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *Symposium on Computational Geometry*, 2004.

Chapter 4

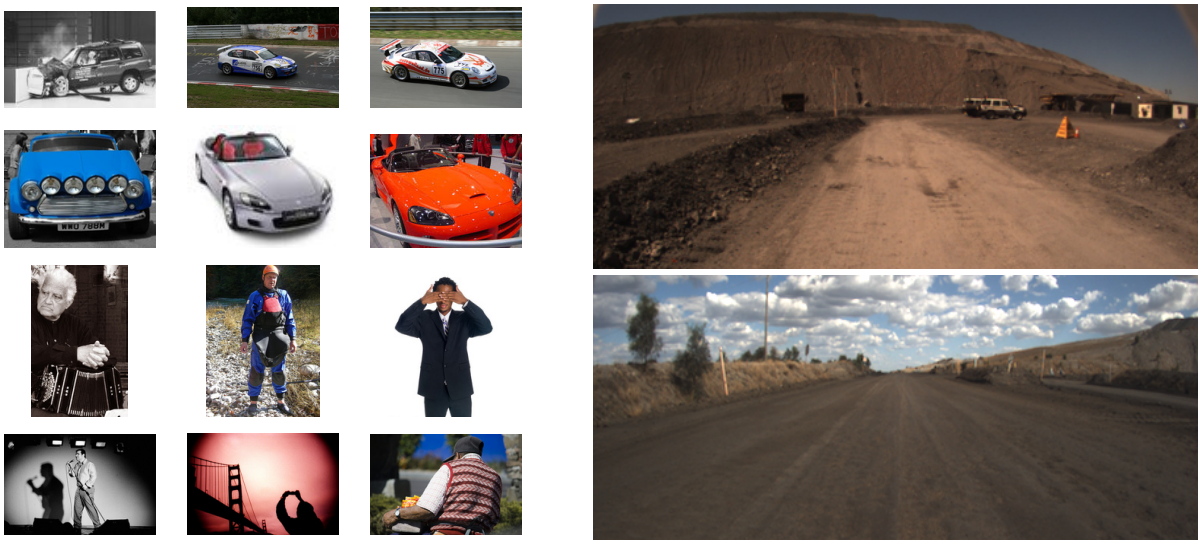
Background Modelling for Adapting to the Deployed Environment

---

This chapter continues the theme that a pretrained object detector may perform poorly in environments which are visually dissimilar to the training data. In contrast to learning new objects by their motion as performed in the previous chapter, this chapter presents a method for detecting objects based on their appearance dissimilarity to the background. This is demonstrated by applying the state-of-the-art appearance based Region Convolutional Neural Network (RCNN) detector of [Girshick et al., 2014] in a surface mining environment. As this environment contains substantially different visual characteristics to the typical internet based images (for example see Figure 4.1) used in training the RCNN detector, a high rate of errors are encountered.

This chapter presents a solution to the second research question of: *How to adapt an appearance based detector for deployment in a new and novel environment with different background characteristics to the training data?* Compared to the background subtraction methods presented in the literature review, background modelling for a moving camera needs to be abstract and motion invariant. To this end, background patches are modelled using unsupervised learning by grouping common appearances into different clusters to represent the different categories of background.

This work investigates the combination of two philosophies; namely, use of a general Deep Convolutional (Neural) Networks (DCN) object detector with background modelling techniques which specialises in identifying if a region only contains background. Such an approach is



(a) ImageNet [Deng et al., 2009] training images. (b) Images collected during deployment.

Figure 4.1: Sample images illustrating visual dissimilarity between internet based and vehicle mounted images collected during deployment.

motivated by the difficulty in retraining a DCN on imbalanced datasets where the availability of background data dwarfs the number of labelled deployment specific object training samples. By checking for novelty against a separate background model the detector itself can be used either independently without retraining or retrained on a balanced subset of the deployment specific data.

A method for gathering background patches is presented that is specific to the deployed environment where only a simple inspection of a video sequence is required to avoid intensive object labelling. Additionally, this chapter also takes advantage of the recent developments in DCNs, in particular, the expressive power of intermediate DCN features [Razavian et al., 2014] which are used for describing the background patches. By reusing the computed intermediate features, this method efficiently validates the DCN detections by comparing to the gathered corpus of background patches.

This chapter contains a journal paper [Bewley and Upcroft, 2016] which extends the preliminary work of Bewley and Upcroft [2015]. Namely, the journal paper includes further analysis of the use of region proposals, details an improved variant of the background model and presents a Bayesian fusion framework for combining the estimates from the DCN and the background model. As there is redundancy between the two papers, the extended journal version is presented first, leaving the option to pass over the preliminary conference version. Furthermore, this chapter includes a comparative study between the proposed weakly supervised background

model and a pure DCN model trained with significantly more labelled mining data.

RESEARCH STUDENTS CENTRE
Examination Enquiries: 07 3138 1839
Email: research.examination@qut.edu.au

Statement of Contribution of Co-Authors for Thesis by Published Paper

The following is the format for the required declaration provided at the start of any thesis chapter which includes a co-authored publication.

The authors listed below have certified* that:

1. they meet the criteria for authorship in that they have participated in the conception, execution, or interpretation, of at least that part of the publication in their field of expertise;
2. they take public responsibility for their part of the publication, except for the responsible author who accepts overall responsibility for the publication;
3. there are no other authors of the publication according to these criteria;
4. potential conflicts of interest have been disclosed to (a) granting bodies, (b) the editor or publisher of journals or other publications, and (c) the head of the responsible academic unit, and
5. they agree to the use of the publication in the student's thesis and its publication on the QUT ePrints database consistent with any limitations set by publisher requirements.

In the case of this chapter:

Publication title and date of publication or status:
"Background Appearance Modelling with Applications to Visual Object Detection in an Open Pit Mine" (submitted to JFR, 2016)

Contributor	Statement of contribution
Alex Bewley  8th April 2016	Wrote the manuscript, designed and conducted experiments and performed data analysis
Ben Upcroft	Provided input into scope of paper and proofreading

Principal Supervisor Confirmation

I have sighted email or other correspondence from all Co-authors confirming their certifying authorship.

Ben Upcroft		15/4/16
Name	Signature	Date

Background Appearance Modelling with Applications to Visual Object Detection in an Open Pit Mine

Alex Bewley
School of Electrical Engineering and Computer Science
Queensland University of Technology,
Brisbane, Australia.
alex.bewley@hdr.qut.edu.au

Ben Upcroft *
ARC Centre of Excellence for Robotic Vision,
Queensland University of Technology,
Brisbane, Australia.
ben.upcroft@qut.edu.au

Abstract

This paper addresses the problem of detecting people and vehicles on a surface mine by presenting an architecture that combines the complementary strengths of deep convolutional networks (DCN) with cluster based analysis. We highlight that using a DCN in a naïve black box approach results in a significantly high rate of errors due to the lack of mining specific training data and the unique landscape in a mine site. In this work we propose a background model that exploits the abundance of background-only images to discover the natural clusters in visual appearance using features extracted from the DCN. Both a simple nearest cluster based background model and an extended model with cosine features are investigated for their ability to identify and suppress potential false positives made by the DCN. Furthermore, localisation of objects of interest is enabled through region proposals, which have been tuned to increase recall within the constraints of a computational budget. Finally, a soft fusion framework is presented to combine the estimates of both the DCN and background model to improve the accuracy of the detection. Our system is tested on over 11km of real mine site data in both day and night conditions where we were able to detect both light and heavy vehicles along with mining personnel. We show that the introduction of our background model improves the detection performance. In particular, soft fusion of the background model and the DCN output produces a relative improvement in the F1 score of 46% and 28% compared to a baseline pre-trained DCN and a DCN retrained with mining images respectively.

1 Introduction

While the mining industry pushes for greater autonomy, there still remains a need for human presence on many existing mine sites. This places significant importance on the safe interaction between human occupied and remotely operated or autonomous vehicles. In particular, for reliable collision avoidance it is necessary to maintain sufficient situational awareness. To improve the situational awareness in regards to collision avoidance, this work investigates computer vision based techniques for detecting other vehicles and personnel in the workspace of heavy vehicles such as haul trucks.

Traditionally, methods for detecting other vehicles and personnel from heavy mining equipment have relied on radio transponder based technologies. Despite transponder based sensors being mature and reliable for ideal conditions, in practise their reliability is circumvented by practical issues around their two way active

*http://www.roboticvision.org/

nature, portable power requirements, limited spatial resolution and human error. In contrast, computer vision techniques for object detection aim to recognise the presence of other objects by observing a video camera feed. This offers a unique alternative that operates passively and can utilise the available cameras on existing vehicles. Furthermore, it is expected that reliable collision avoidance systems would utilise multiple sensing technologies to provide a high integrity solution with redundancy.

Detecting and recognising objects has long been a topic of research within the computer vision community, which has advanced tremendously in recent years as measured by standard benchmarks (Deng et al., 2009; Lin et al., 2014). These major advancements can be largely attributed to both the availability of huge annotated datasets (Fei-Fei et al., 2007; Torralba et al., 2008; Deng et al., 2009; Lin et al., 2014) and developments in data driven models such as deep convolutional networks (DCN) (Krizhevsky et al., 2012; Sermanet et al., 2013). A DCN effectively models visual appearance through a huge set of parameters which are tuned by training on a large set of annotated images with relevant objects of interest. In this work we utilise the DCN of (Krizhevsky et al., 2012) which has shown astonishing performance on the ImageNet recognition benchmark (Deng et al., 2009) and repurpose it toward recognising personnel and vehicles from the background in an open pit mining environment. However, naïvely applying an off-the-shelf DCN to images collected in a mining environment results in a significant number of false positives due to the differences in appearance between the training set and the target domain. See Figure 1.

Adapting DCNs to different domains typically requires a large training set relevant to the target domain (Zeiler and Fergus, 2013; Yosinski et al., 2014). When the amount of training data is small, data driven approaches tend to over-fit the training samples and not generalise to unseen images. In this work we utilise a pre-trained DCN using millions of images from ImageNet and experiment with both a naïve remapping of ImageNet to mining classes and compare this to retraining the network with limited data.

With a vehicle mounted camera the focal length is fixed and the viewing angle is rigidly coupled to the vehicle's orientation. This distinguishes it from the ImageNet recognition problem where typical images collected were implicitly pointed at regions of interest and appropriately zoomed. Additionally, due to the wide field of view the majority of the images are background with zero to potentially multiple objects of interest visible in any given frame. To address these multi-scale and object localisation issues, we employ a similar strategy to (Girshick et al., 2014) by applying an initial step for finding likely object locations through a region proposal process before performing object recognition with the DCN.

Given that the majority of images collected on a mine site contain zero objects of interest, we can efficiently collect a huge amount of background data suitable for training a linear classifier. Using this newly trained classifier in conjunction with the DCN ensures robustness and drastically reduces spurious detections. This classifier is based on k-means clustering offering a convenient way to implicitly partition the background data into different categories. This approach accurately captures the characteristics of the background, enabling the discovery of novel non-background objects. In light of this, two approaches are presented for using this background model to either suppress false background detections or to correct the detection confidence.

Building off the preliminary work in (Bewley and Upcroft, 2015), this article presents an alternative fusion framework where hard decisions on suppressing background are replaced with a soft probabilistic model. This enables a soft fusion between the likelihood a sample was generated from the novel background model and the class prediction probability produced by the DCN classifier. Furthermore, various stages of the pipeline proposed in (Bewley and Upcroft, 2015) have been optimised to increase either the recall or precision in order to improve the overall detection performance. Additionally, a new method for discriminating between background and novel objects is proposed and incorporated to the fusion framework. Finally, a thorough examination of each component both individually and jointly is performed to highlight the major contributing factors and gain insight into how the vision based detector behaves in a mine site environment.

This paper is organised as follows: Section 2 provides a short review of related literature in the areas of mine-site sensing, pedestrian and general object detection, and information fusion. Section 3 describes the various steps in our approach including analysis for region proposals, adapting DCNs, formulation of the background

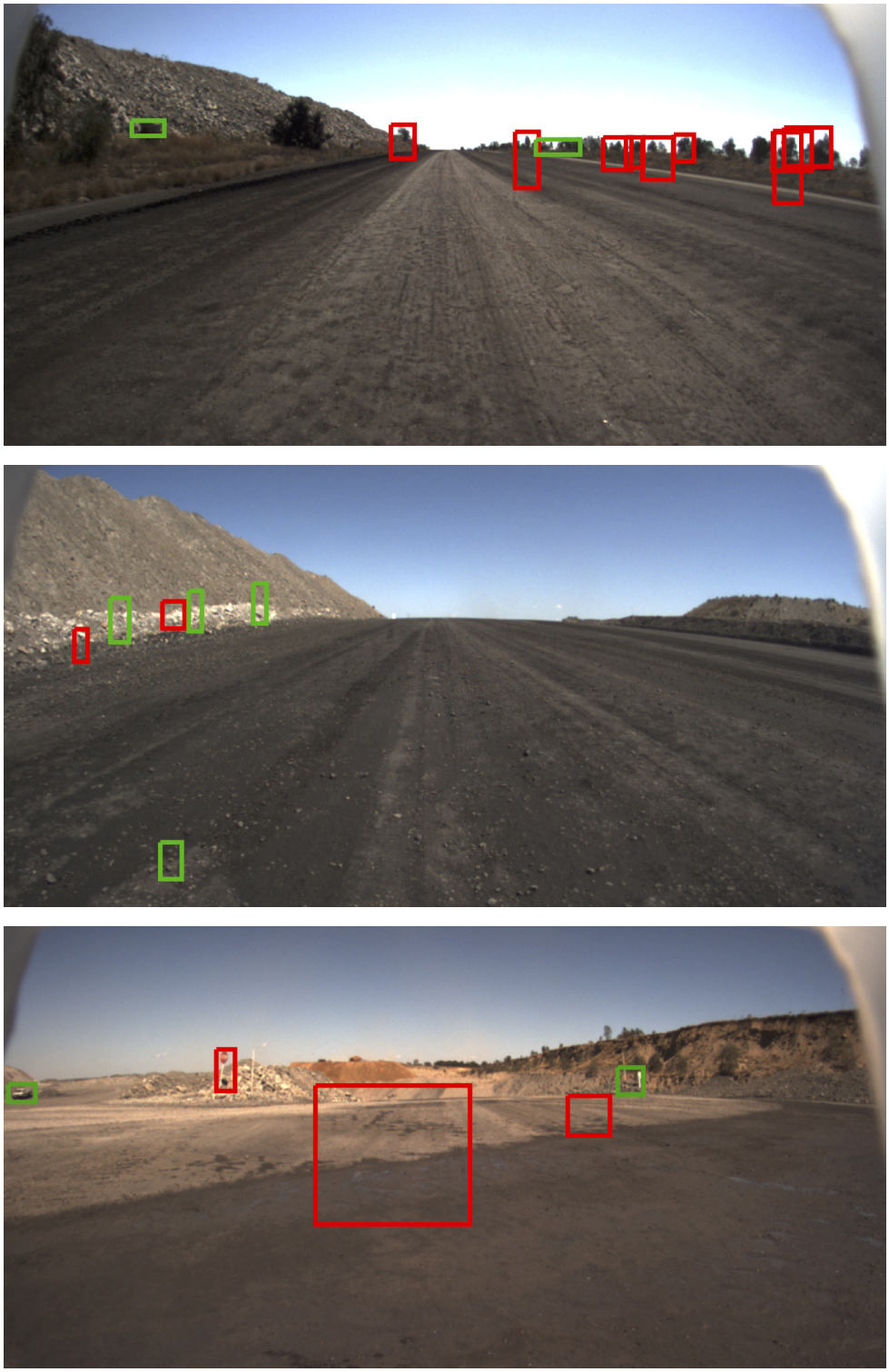


Figure 1: An example of the number of false positives when applying a DCN off-the-shelf in a mining environment. The colours represent different classes, e.g. person (*red*) and vehicle (*green*). The only true positive is the light vehicle on the left in the bottom image. Best viewed in digital form.

novelty detector, and information fusion. In Section 4 the detection performance of the proposed method is analysed on a challenging set of mining videos. Finally, Section 5 concludes the paper with a summary of the learnt outcomes and discussion for future improvement.

2 Related Work

Here we briefly review object detection methods that are not reliant on two way communication before covering some related work using DCN for generic object detection. Early work has focused on range based techniques such as LiDAR (Roberts and Corke, 2000; Marshall and Barfoot, 2008) commonly used for mapping fixed obstacles such as buildings or underground tunnel walls. Applying these sensors to detecting personnel and vehicles fitted with retro-reflectors is found to be sensitive to the dynamics of the sensor platform (Phillips et al., 2013). In this work we focus specifically on detecting potentially dynamic obstacles including vehicles and particularly people from vision based data. To this end, the more relevant prior work is that of (Mosberger and Andreasson, 2012) which exploits the standardised requirement for personnel on mine sites to wear high-visibility clothing equipped with retro-reflector patches. This enables a single IR camera with active flash to highlight personnel in view which can then be used for tracking (Mosberger et al., 2013). In this work, we do not restrict the detection system to specifically finding retro-reflectors, but rather formulate the problem as a recognition task focusing on the overall appearance of personnel and mining vehicles.

Pedestrian detection using computer vision techniques is an actively researched topic (Dalal and Triggs, 2005; Dollár et al., 2009; Benenson et al., 2013; Dollar et al., 2014; Zhang et al., 2015). These techniques take a sliding window approach, where for each image location, multiple rectangles with different scales and aspect ratios are examined for the presence of a pedestrian. This examination is generally characterised by extracting features specifically designed for pedestrians (Dalal and Triggs, 2005) before using a binary classifier (Dalal and Triggs, 2005; Dollár et al., 2009; Benenson et al., 2013) or cascade of classifiers (Dollar et al., 2014; Zhang et al., 2015) to determine if the rectangle represents a pedestrian.

Recent popularity of big data and deep learning have dominated the object recognition problem. Among these data driven approaches, deep convolutional networks (DCN) with recognition performance quickly approaching human levels (Krizhevsky et al., 2012; Donahue et al., 2013; Razavian et al., 2014; Sermanet et al., 2014) are selected for use in this work. DCNs themselves have been used for over 20 years (LeCun et al., 1989) for tasks such as character recognition. Over recent years DCNs have made an astonishing impact on the task of object recognition within the computer vision community (Krizhevsky et al., 2012; Farabet et al., 2013; Razavian et al., 2014; Girshick et al., 2014; Donahue et al., 2013) largely contributed to the availability of huge labelled image sets such as ImageNet (Deng et al., 2009). In this work, we use a network based on the work of (Krizhevsky et al., 2012) that was pre-trained on ImageNet which we re-purpose for the mine-site environment with limited label data and large sequences of unlabelled background data. While even deeper network architectures (Simonyan and Zisserman, 2014; Szegedy et al., 2015) have recently surpassed the (Krizhevsky et al., 2012) model for the ImageNet task, we continue to use the (Krizhevsky et al., 2012) model for efficiency and argue that a comparison of network architectures is beyond the scope of this work.

Recognising what objects are in an image is only half of the object detection problem. The other half is locating the objects within the image. Sermanet et al. (Sermanet et al., 2014) sample over multiple scales and exploit the inherently spatially dense nature of the convolutions within DCNs to identify regions with high responses. Similarly, (Farabet et al., 2013) also perform convolutions over multiple scales and combine the responses over superpixel segmentation (Felzenszwalb and Huttenlocher, 2004). Another popular approach and the one that we base this work off is the region convolutional neural network (RCNN) of (Girshick et al., 2014). The RCNN framework efficiently combines the DCN of (Krizhevsky et al., 2012) with an object proposal method: selective search (Uijlings et al., 2013). Generic object proposal methods aim to efficiently scan the entire image at different scales and aspect ratios to reduce potentially millions of search windows

down to hundreds (Hosang et al., 2014) of the most likely candidates. In this work we use `edge box` object proposals (Zitnick and Dollár, 2014) as the accuracy is higher and significantly faster according to a recent survey of object proposal methods (Hosang et al., 2014).

In this work we make use of information fusion for combining information from both the background model and the DCN output. In (Bewley and Upcroft, 2015) simple logic was used for combining the output of both systems, such that if the DCN responded with a `car` *or* `person` *and* the sample was *not* `background` then it would indicate a detection. This approach creates a hard decision where a failure in either system results in a miss detection. In contrast, probablistic approaches allow for a softer fusion of information (Dempster, 2008). In this work, we utilise probablistic variants of the DCN and background model in order to formulate the fusion of information in a classical Bayesian inference framework (Durrant-Whyte and Henderson, 2008).

3 Methodology

In this section, we describe our approach to vision based object detection where we highlight similarities and differences to the inspiring RCNN pipeline (Girshick et al., 2014). Our method consists of three key phases: 1) Region proposals with non-maximum suppression (NMS), 2) DCN recognition and finally, 3) Detections are validated by checking for novelty against the background model. See Figure 2 for a high-level overview of this pipeline. We bypass the problem of over-fitting on a small dataset by using a pre-training DCN and map its output to mining relevant classes.

The detection method is then extended with the incorporation of background modelling techniques to improve and adapt to the mining environment. From this background model we investigate two different novelty measures approximating the probability a detection was not generated by a background sample. This novelty measure produces a single value on the interval $[0, 1]$ where low values represent a high similarity with the background model and the high values represent that the sample is novel with respect to the background model. Finally, we describe how information from both the DCN prediction and the background model are combined using a probabilistic fusion framework.

3.1 Region Proposals

The aim of region proposals is to efficiently scan the image to eliminate millions of potential windows, keeping only the regions that are likely to contain an object of interest. We use the `EdgeBoxes` region proposal method (Zitnick and Dollár, 2014) over the `selective search` (Uijlings et al., 2013) used in the original RCNN work as this method is orders of magnitude faster with comparable accuracy. For a detailed comparison of region proposal methods we refer the reader to (Hosang et al., 2014).

Region proposal methods are regarded as a generic object proposal technique that is independent of object class (Hosang et al., 2014). However, in practice these methods are sensitive to parameter tuning, requiring data relevant to the target domain for optimal performance (McMahon et al., 2015). The default parameters for `EdgeBoxes` were adjusted to return a fixed 1000 proposals and an additional step of non-maximum suppression (NMS) is applied and described next.

3.2 Non-maximum Suppression

The region proposals provided by the `EdgeBoxes` method are further reduced through a process of NMS. The NMS process considers the score produced by the `EdgeBoxes` (EB) method and the overlap with other bounding boxes. As the name suggests it then greedily suppresses all but the maximum scoring proposal for all adjacent overlapping regions where the intersection-over-union (IOU) area is greater than a set threshold. In contrast to applying NMS after the DCN (Girshick et al., 2014), this way we can speed up the detection

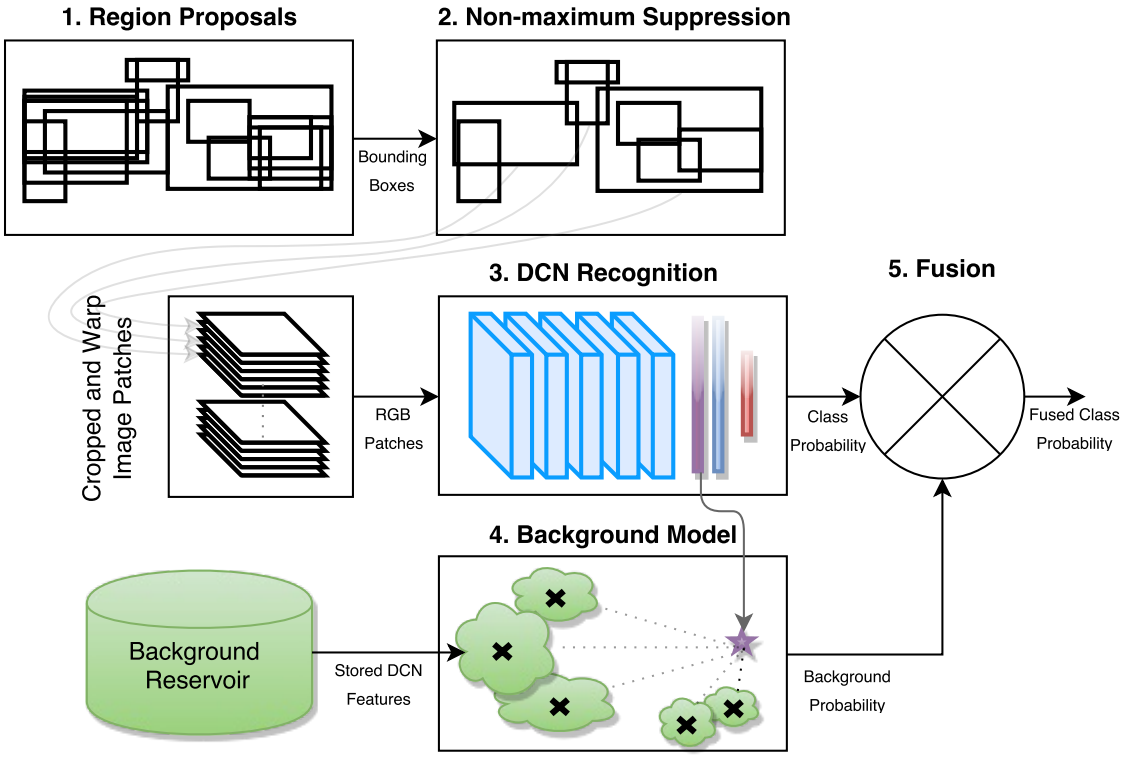


Figure 2: An illustration of the detection pipeline used in this work. When a new image is provided, the detection pipeline begins with the region proposal step. The system parameters are highlighted in blue and green which are learnt offline from an off-the-shelf network and background only images respectively. Purple represents the responses from an intermediate layer (also depicted as the purple star), that is compared with the cluster centers of the background model indicated with black crosses. The red output layer of the DCN contains scores for each class type. The output from the DCN and background model are fused to produce the systems final output.

pipeline by reducing the number of proposals going into the DCN while maintaining comparable coverage over the image.

To better understand the effect of applying NMS to the EB scores, the trade-off between the number of proposals and its effect on the recall is further investigated. Ideally, it is preferable to have as few proposals needed to cover all visible objects in order to minimise both the computational load of the later DCN and the overall rate of false positives. Since object proposals is used as the first step in the pipeline, it is paramount to have a good coverage of the true objects in the image since missed objects will never be recovered (Hosang et al., 2014). Figure 3 shows a comparison between applying NMS to the region proposals and reducing the number of proposals using the EB score. The EB+NMS curve shows that applying small amounts of NMS rapidly reduces the number of proposals while maintaining high recall. However, when suppressing proposals with overlaps lower than 0.3 IOU, results in lower performance compared to simply thresholding the EB score. The S shape of the EB+NMS curve shows that there exist a non-linear relationship between the localisation accuracy around true objects and the number of proposals. The overlap threshold in this work is set at 0.5 to produce approximately 260-300 proposals per frame where there is a considerable reduction in proposals (possibilities for false positives) for a moderate drop in recall.

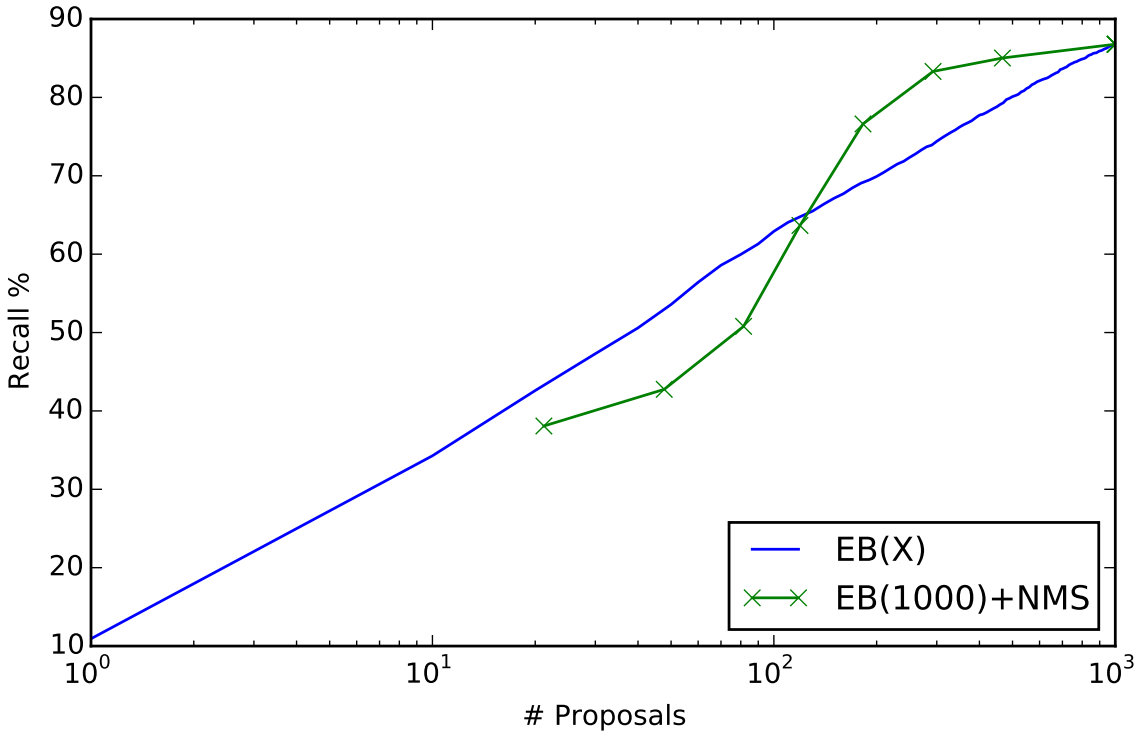


Figure 3: Performance of EdgeBoxes (EB) region proposal method along with non-maximum suppression applied with overlap thresholds set to $[0, 0.1, 0.2, ..., 0.7]$ shown with crosses. Note that the EB+NMS curve converges with the EB curve at NMS threshold of 0.7 since that is the limit used in the EB default parameters.

3.3 Region Classification

Having selected regions of the image that have the general characteristics of an object, we now perform object recognition to distinguish the object category. For this we apply the DCN from RCNN (Girshick et al., 2014) which is based on the winning architecture (Krizhevsky et al., 2012) for the ImageNet Large Scale Recognition challenge in 2012. For this work, we used the RCNN implementation provided with the Convolutional Architecture for Fast Feature Embedding (`caffe`) (Jia et al., 2014) framework out-of-the-box and apply it to classifiying the content of the object proposals generated as previously described.

Since the work in this paper largely gravitates around the recent success of DCNs, we briefly describe their workings. A DCN consists of multiple layers, each performing a transformation of their input data in the form of a linear projection followed by a differentiable, non-linear activation function to produce an output response. A set of weights and biases govern the linear projection step in each layer which essentially performs an inner product with the input and the weight parameters. The output responses for each layer forms the input for the following layer in a feed-forward fashion, where the first input is the data and the last represents a score for each class.

The architecture of (Krizhevsky et al., 2012) used in RCNN consists of eight layers in total with five being convolutional (conv1-conv5) and three fully-connected (fc6-fc8). In the convolutional layers, the weights only connect to a small receptive field in the input – acting like a 2D filter (or kernel) – that is shifted across the lateral dimensions of the input to perform a 2D convolution. The fully-connected layers on the other hand, remove the spatial structure of the data by flattening the transformed input before applying an inner product with their weights to project the entire input into a new high dimensional space. The intuition of

this DCN architecture is that the convolutional layers transform the input into low level visual descriptors representing local object parts, while the fully-connected layers are responsible for the high-level task putting the parts together to classify the entire image (Azizpour et al., 2015).

The RCNN architecture consists of around 60 million parameters across all eight layers which were optimised for classification via *deep learning*. Deep learning is the process of first applying a each layer transformation in turn to the input data to produce the network output. The classification error of the network (or loss) is then used to adjust the weights by an amount proportional to the error with respect to the layer input. As each layer is differentiable, the chain-rule enables the error to be propagated back to update the parameters in all layers. Given the high number of parameters, this network was trained using the large ImageNet dataset consisting of 1.3 million labelled images. Training a model on this scale is enabled through the use of Stochastic Gradient Descent (SGD).

The original detection task for the `caffe` RCNN model was to predict one of 200 classes that represent common objects found in images taken from the internet. For this application we are only interested in distinguishing between four high level categories, namely: `background`, `person`, `light vehicles(LV)` and `heavy vehicles (HV)`. Using a DCN model trained on ImageNet in a mining context raises several issues that need addressing:

1. The majority of the 200 ImageNet classes are indoor/domestic related objects including Food (apple,burrito, etc.), Musical Instruments (accordion, saxophone, etc.) or various animals (bird, camel, etc.)
2. How to associate mining classes with ImageNet classes?
3. Semantically, the `background` is significantly different from many of the existing object specific classes.

To gain some insight, we use a small validation set of 200 mining related images to investigate the output of the DCN out-of-the-box. This set is made up of cropped mine-site images containing the classes `person`, `LV` and `HV` along with 90 interesting region proposals extracted from background only images taken on mine sites. In Figure 4 we show the results of naïvely applying the pre-trained RCNN model to this image set. To better visualise the output we applied a soft-max transform to approximate the output class prediction as a probabilistic estimate. [1]

Not surprisingly, the `person` and `LV` classes are well represented and can be directly mapped from the `person` and `car` ImageNet classes used to train the original DCN. On the other hand, the `background` closely resembles uniform random sampling of classes as there are no relevant classes in the existing model such as trees, buildings, or road signs etc. Similarly, the `HV` class prediction also mostly resembles a uniformly random distribution with a slight bias towards the ImageNet classes `snowplow`, `cart` and `bus`.

3.3.1 Remapped Model

While the `person` or `car` class outputs can be simply mapped to `person` and `LV`, distinguishing between `background` and `HV` from the output is without avail. From a practical perspective, detecting `person` and `LV` is of higher importance for a passive computer vision system since `HV` are equipped with active protection devices. With this in mind, we simply assign all 198 non `person` or `car` outputs as background and accept that not detecting `HV` is a limitation of this remapping approach.

With this simple class mapping approach and assuming that falsely picking one of the positive classes is in fact uniformly random, we expect to eliminate 99% of all the proposed background regions. However, when

[1] It is important to note that this is for visualisation purposes only and that the y-axis does not represent the true probability since the finalsupport vector machine (SVM) layer of RCNN was not calibrated for probabilistic outputs.

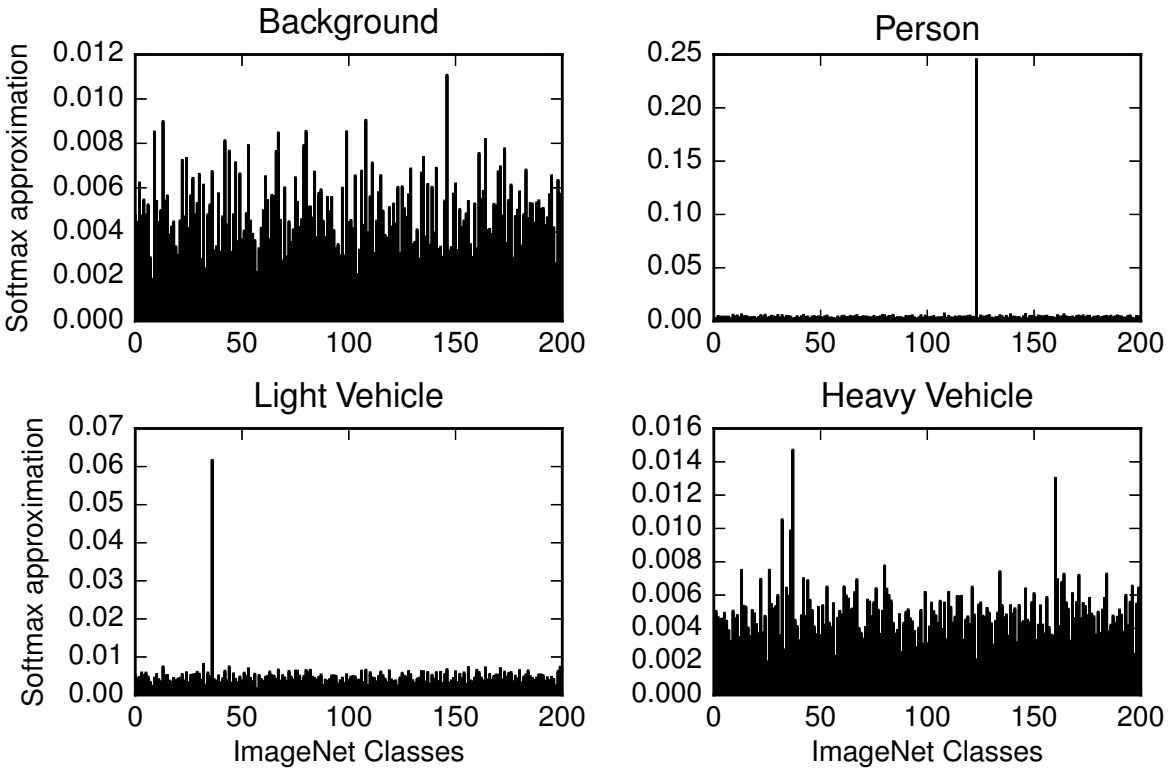


Figure 4: The average class estimate for a set of mining related images. Notice that person (class 123) and light vehicle/car (class 36) are existing classes for the pre-trained network and can be used directly. The background and the heavy vehicle classes are novel and show a wider spread as they are not modelled with the pre-trained DCN.

processing around 100 proposals per frame, the expected false positive rate is once per frame. Later, we propose a simple background model that reuses the DCN computation to provide a background likelihood estimate for reducing this false positive rate.

3.3.2 Retrained Model

An alternative is to train the DCN specifically on mining images for the target classes, including the `HV` class. This does raise the issue of learning the DCN parameters from the limited mining specific training data available. When using a small training set, the DCN can settle into a state that fails to recognise the wide variety of patterns that occur in the real world. This is largely caused by the massive number of connections inside a network that tend to converge on the few unique patterns in the training set. To overcome this effect of over-fitting the training data, many researchers (Ahmed et al., 2008; Aytar and Zisserman, 2011; Oquab et al., 2014; Yosinski et al., 2014; Azizpour et al., 2015) choose to take a network pre-trained on a massive dataset such as ImageNet and retrain only the last few layers for the task of predicting a set of domain specific classes. The intuition behind this is that the visual knowledge gained from the larger dataset is transferred to the target task to maintain diversity in low level feature extraction while the higher level classification task is allowed to adapt to the new domain.

3.3.3 Remap vs Retrain

In this work, we took the off-the-shelf DCN with 200 ImageNet outputs and remapped the mining classes referred to as the *remapped* DCN. Additionally, we train another model by replacing the last layer entirely and retrain the network keeping all but the last two layers fixed to prevent over-fitting. This network was trained until the loss reached zero on the set of 200 labelled mining specific examples. This model is referred to as *retrained*. Since the number of labelled examples used for training is small, Figure 5 shows the *retrained* model has negligible performance improvement over the *remapped* version which does not require any training. However, retraining the network with a softmax loss, effectively trains the model to accurately produce a probabilistic distribution over the target classes in contrast to using an SVM loss for the last layer as in (Girshick et al., 2014). As we will later show, this is important when fusing the DCN output with the proposed background model. Additionally, by retraining the DCN with outputs responsible for predicting mining related classes, it can be trained to also detect `HV` with negligible computational overhead.

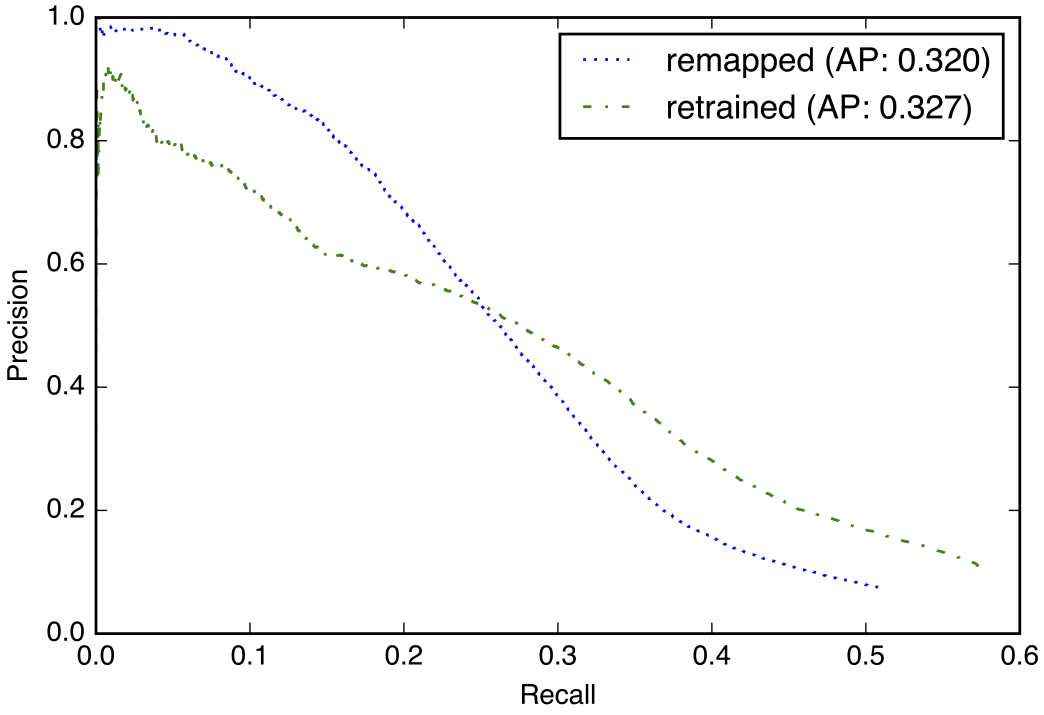


Figure 5: A comparison between simply remapping the ImageNet classes to Mining classes based on the activations shown in 4, versus retraining the network with mining images for classifying the target mining related categories directly. The average precision (AP) is provided in the legend.

3.4 Background Modelling

While the landscape on a mine-site constantly changes over time in a geometric perspective, the bleak visual appearance of the background remains predominantly consistent. The aim of this model is to capture the visual appearance of common background regions which are poorly characterised by the DCN used for classifying object proposals. At test time a proposal's similarity to this model provides an additional measure of confidence to be fused with the prediction output of the DCN model.

For a given background region, it generally belongs to one of an arbitrary set of categories, such as the

semantic categories of rock, sky, tree etc. Rather than using supervised techniques that require a set of manually annotated images, we instead partition the background data without explicit semantic labels. To do this, we exploit the assumption that intra-category samples generally appear visually similar to each other, yet may be distinctively different to other background categories. Put another way, the background regions form natural clusters enabling us to employ unsupervised techniques to model their visual appearance. See Figure 6 for an illustration of the natural background clusters found by applying a clustering approach to a mining dataset.

To describe the visual appearance of each region, the intermediate layers of the DCN provide a free and compact representation suitable for this task. Additionally, these features have been shown to be robust against lighting and viewpoint changes without any re-training (Sünderhauf et al., 2015). We refer the interested reader to (Krizhevsky et al., 2012) for an illustration of the DCN's inner workings. In general, the first layer of a DCN extracts simple colour and texture features in the first layer, and through subsequent layers, these features eventually transition to the learnt specific task (Yosinski et al., 2014) such as classifying the 200 ImageNet classes. Along the way irrelevant visual information for the original task (e.g. features describing sky) are lost once it reaches the final layer. With this intuition, we reuse the transformed data from one of the DCN's intermediate layers as an appearance descriptor for input to our background model.

To learn this cluster based model, a reservoir of negative samples is required. Gathering background data is a relatively simple task since only inspection for the presence of target objects is necessary. Specifically, any image sequence not containing any of the target objects can be used to build an extremely large reservoir by extracting proposals from each frame.

Ideally this background reservoir should be large enough to contain sufficient diversity to capture the variation in visual appearances, while also small enough for efficient nearest neighbour querying. To achieve this, a process called "hard"-negative-mining (Felzenszwalb et al., 2010) is utilised to focus only on the most informative image regions. Specifically, we run the detection pipeline with the pre-trained RCNN model of (Girshick et al., 2014) over these background only sequences and keep only regions that falsely classified as either a `person` or `car`. A sufficiently large reservoir can be built by also including near false positive regions that fall within the SVM margin of the pre-trained RCNN model. In doing this we are left with a collection of background region patches that are challenging in the sense that the DCN is unable to confidently classify them as background.

3.4.1 Nearest K-means Cluster Model

The first background model simply consists of a non-parametric model where the Euclidean distance to the nearest background example is used as a measure of novelty. Since the background reservoir is large, potentially redundant, and computing Euclidean distances for high dimensional data is slow, it is desirable to represent the background samples as a few exemplar points. With the intuition that the background forms natural clusters we choose to represent the background appearance as a set of clusters. Using the response output from an intermediate layer of the DCN as a visual feature $\mathbf{f}$ for a region proposal, the k-means algorithm is applied to group the reservoir points into clusters. This facilitates the selectable size of the background model M where the background sample points are the cluster centroids computed as follows:

$$\mathbf{m}_i = \frac{1}{|k_i|} \sum_j \mathbf{f}_j, j \in k_i, \tag{1}$$

where $\mathbf{f}_j$ is the visual feature of sample j which is a member of the cluster k_i.

At test time, each `person` or `LV` predicted patch is verified by measuring the Euclidean distance between its intermediate feature $\mathbf{f}_t$ and the nearest cluster mean

$$d_t = min_i(\|\mathbf{f}_t - \mathbf{m}_i\|). \tag{2}$$

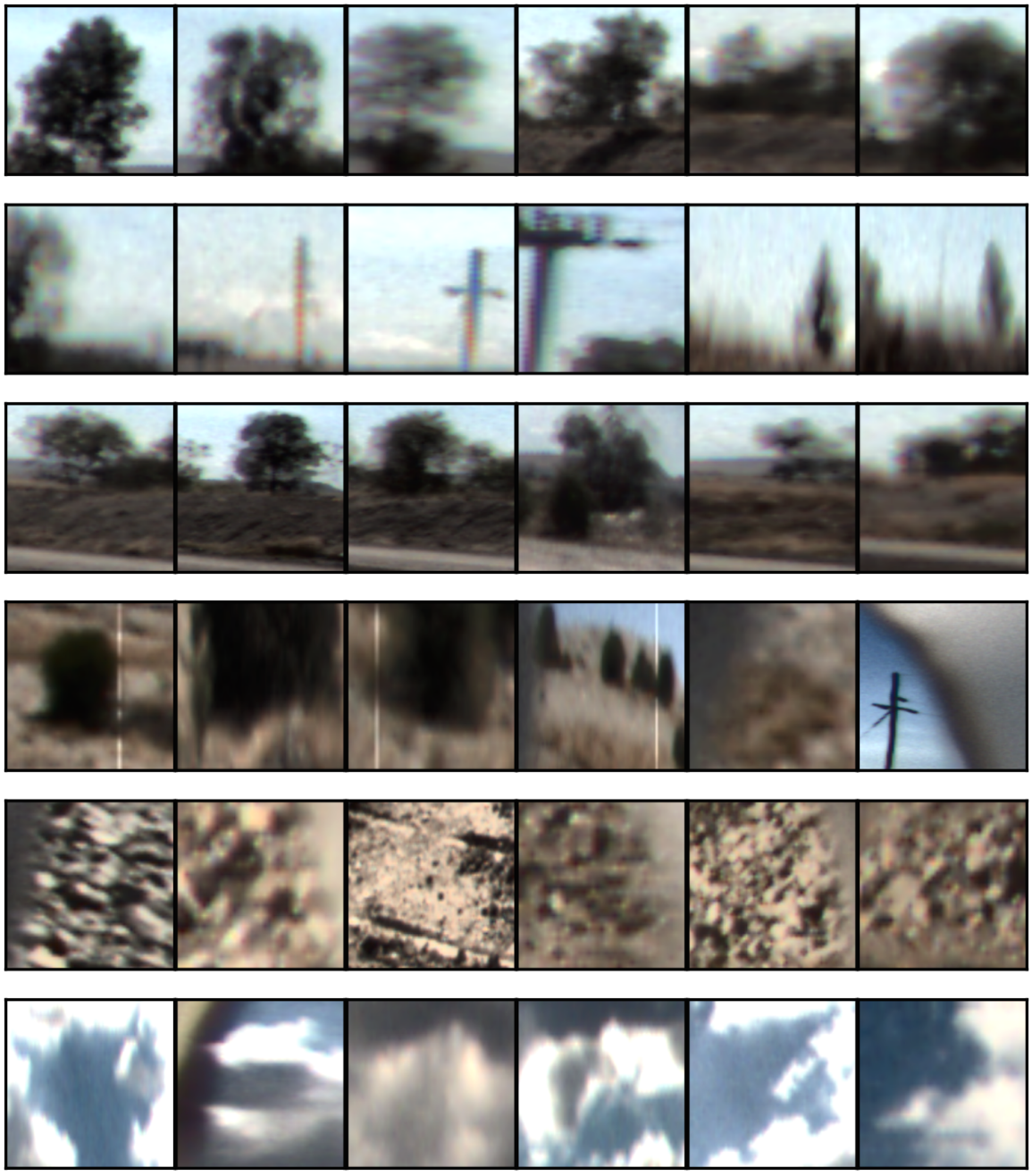


Figure 6: An illustration showing six of the most common types of background region proposals out of a total of 128 clusters. The rows represent different clusters while the columns show a random background region which is a member of the associated cluster. Each cluster gathers samples with similar visual appearance such as centred on a tree (top row) or centred on sky with an adjacent vertical structure (second row). Qualitatively, k-means clustering naturally formed mostly pure clusters with the exception of the forth row which also covers a number of patches occurring with relatively low frequency.

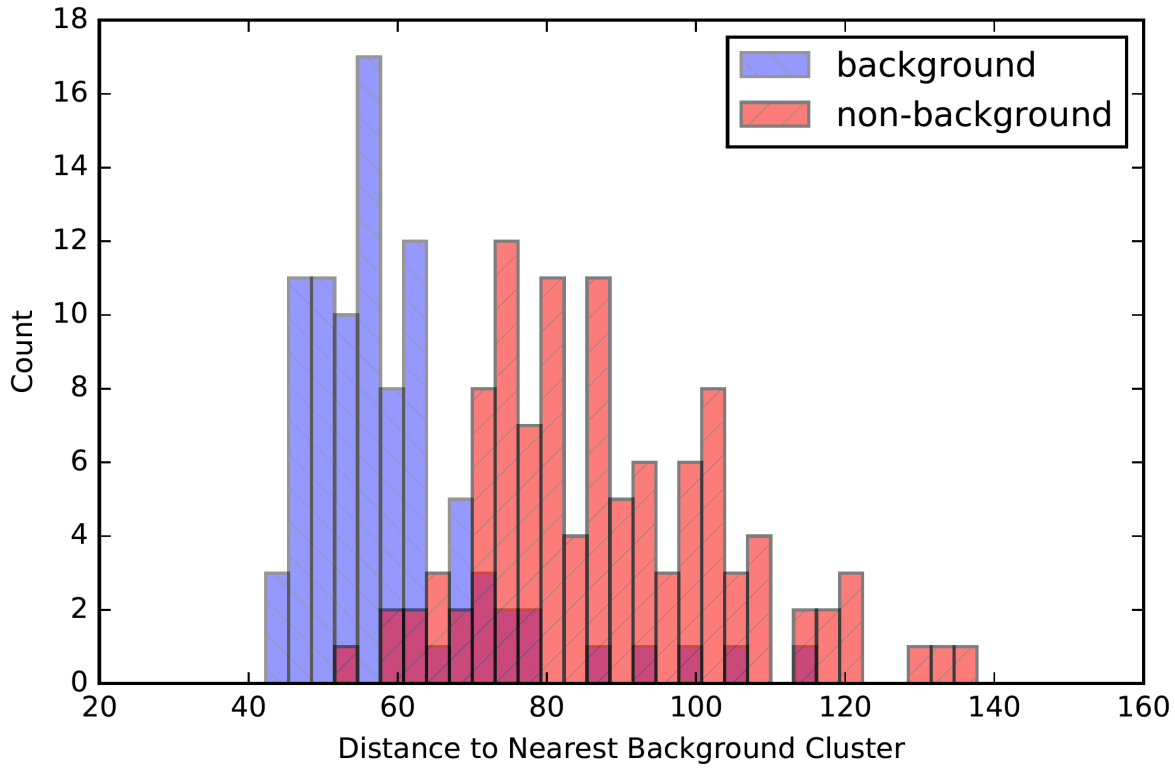


Figure 7: Detailed view of the distance to nearest cluster distribution for validation images composed of background and non-background images.

If the nearest background cluster is close in this feature space, i.e. is visually similar, then the likelihood that the test patch is background should increase. Conversely, it is expected that a patch corresponding to an object of interest should be dissimilar to the background model, which is signified by a greater distance to the nearest cluster centre. This expectation is corroborated with empirical evaluation on a small set of held out background patches and 110 images of non-background objects as shown in Figure 7.

While a hard threshold on the distance between a test patch and the nearest cluster centre can be used to suppress background detections (Bewley and Upcroft, 2015), a probabilistic version is favoured to enable soft fusion with other information such as the DCN output. To achieve this, the large collection of background samples is modelled using a parametric distribution. Figure 8 shows three different parametric distributions overlaid on the empirical histogram of distances between the background patches and their corresponding cluster centre. The *gamma* distribution best fits this data and is therefore used to approximate the likelihood of distances conditioned on a background sample being supplied. The cumulative density function is then used as a surrogate to the probability a test sample was not generated by the background as it has the property of monotonically increasing with the minimum distance to any background cluster. This soft probabilistic estimate of the test feature $\mathbf{f}_t$ being a non-background sample is computed using the Euclidean distance d_t to the nearest cluster as:

$$P(y_{BGM}|c \neq BG) = \frac{\gamma(\alpha, \beta d_t)}{\Gamma(\alpha)}, \tag{3}$$

where γ and Γ are the lower and upper incomplete gamma functions following the standard cumulative distribution function of a *gamma* distribution. The parameters α and β represent the shape and rate

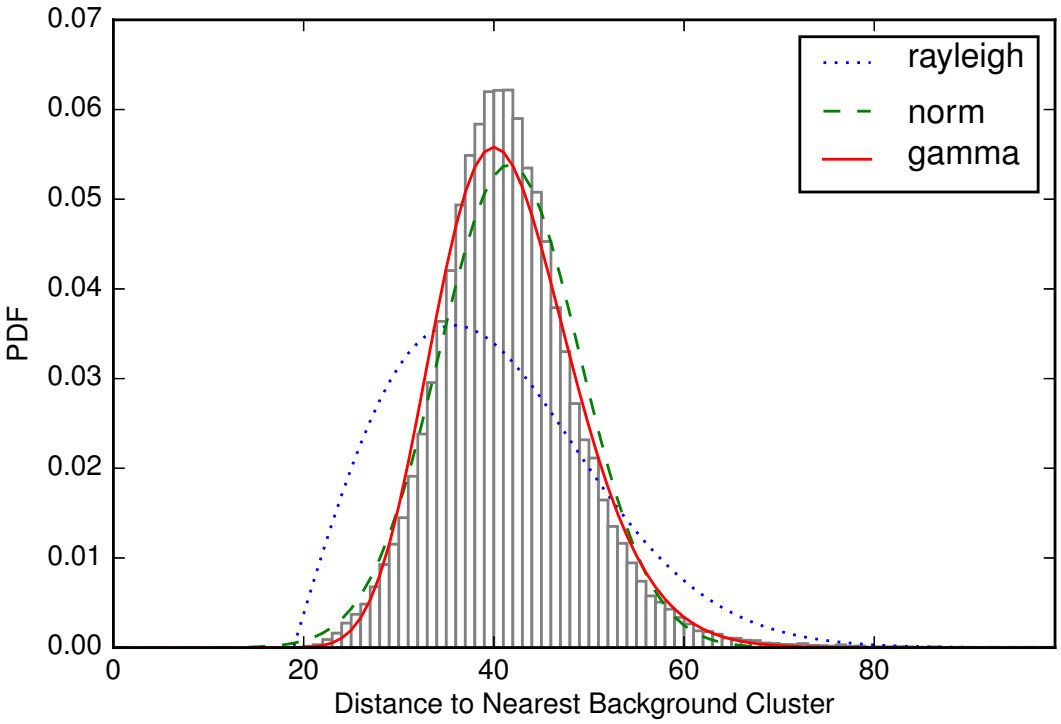


Figure 8: Distribution of background samples as a distance from nearest cluster centre.

parameters of the gamma distribution. These parameters can be estimated by fitting to the reservoir data as shown in Figure 8.

3.4.2 Joint Cosine Similarity Model

This alternate approach considers the similarity of all background clusters jointly rather than only the most similar cluster. Here, the similarity from a test sample to all cluster centres from the k-means model is used to discriminate between background and non-background samples. This enables a non-uniform weighting for each of the clusters effectively transforming the extracted DCN feature into a new feature space. This weighting is learnt by formulating the background model as a binary classification where the goal is to discriminate between background and non-background samples.

Here we reconsider the use of the Euclidean distance for measuring novelty from the background clusters. Particularly, given the high dimensionality of the DCN features, the Euclidean distance is easily compromised by the *curse of dimensionality* (Aggarwal et al., 2001). This would also explain the peculiar property visible in Figure 8 where the distance from the background samples to their nearest cluster centroid is distributed far from zero, with low variance. To investigate this, an alternative similarity measure is introduced based on the *cosine* distance to the cluster centres, which is considered to be more robust in high dimensions. In this new background model a weighted sum of all similarity measures between the test sample and the clusters is used to form a new feature representation.

Algorithm 1, details how the transformation of DCN features into a joint similarity representation along with learning the weight and bias parameter for a logistic regression. The $k \times n$ matrix S is used to store and represent a set of labelled features F in their joint similarity form. This transformation essentially

Algorithm 1 Logistic Cosine Background Model Learning

Input: M ▷ Set of all cluster means (Eqn. 1)
Input: F ▷ DCN features of example labelled regions.
Input: $\mathbf{l}$ ▷ Label vector corresponding to F.
Output: θ, b ▷ Weights and bias for logistic regression model.

1: **function** $(M, K, \mathbf{l})$
2: $S = \mathbf{0}_{k \times n}$ ▷ Initialise similarity matrix
3: **for** $\mathbf{f}_j \in F$ **do**
4: **for** $\mathbf{m}_i \in M$ **do**
5: $s_{i,j} = cosine\_similarity(\mathbf{f}_j, \mathbf{m}_i)$
6: $(\theta, b) = cross\_validation\big(train\_LR(), 10, (S, \mathbf{l})\big)$ ▷ Perform 10 fold cross validated learning
return θ, b

compares all n samples in F with each cluster mean vector using the standard cosine similarity measure (lines 3-5). Finally, a logistic regression (LR) classifier is learnt on this transformed data using the labels $\mathbf{l}$ provided (line 6). The weight and bias parameters; θ and b respectively, are optimised using SGD with $k$fold cross validation to select an appropriate amount of regularisation. At test time, the probability that a DCN feature $\mathbf{f}_t$ represents a non-background patch is computed by first transforming into the similarity form $\mathbf{s}_t$, and then fed into the LR classifier as follows:

$$P(y_{BGM}|c \neq BG) = \frac{1}{1 + exp^{-(\theta^T \mathbf{s}_t + b)}}, \tag{4}$$

where $\mathbf{s}_t$ is the joint similarity representation vector of the test sample computed as in lines 4 and 5 of Algorithm 1.

This LR classifier is trained using a held out validation set containing positive and negative patches, where the learnt separation is shown in Figure 9. In contrast to the k-means model where each cluster is treated with equal variance, this joint similarity based model captures the intuition that some clusters may have higher importance when discriminating background for non-background. Additionally, the LR classifier produces a probabilistic estimate enabling it to naturally fit within a Bayesian framework for fusing the background model estimate with the DCN class prediction.

3.5 Fusing Measurements

So far we have described how we can adapt a DCN for making predictions of the mining related class corresponding to each detected proposal. Additionally, a method for extracting features from the DCN, coupled with a clustering approach forms a good representation for identifying background samples. Here, we derive how these two measures can be combined with the goal of producing better estimates than either method individually.

Given the output predictions from both the DCN y_{DCN} and the background model y_{BGM}, we can treat these two predictions as two separate decision makers. While y_{DCN} and y_{BGM} are not independent i.e. $P(y_{DCN}, y_{BGM}) \neq P(y_{DCN})P(y_{BGM})$, they are conditionally independent on the object class c expressed as follows:

$$P(y_{DCN}, y_{BGM}|c) = P(y_{DCN}|c)P(y_{BGM}|c). \tag{5}$$

This independence enables us to fuse the DCN output, which has good target discrimination capabilities, while the background model is good at suppressing non-target proposals.

Using Bayes' Rule the class probability conditioned on the joint output of both the background model and

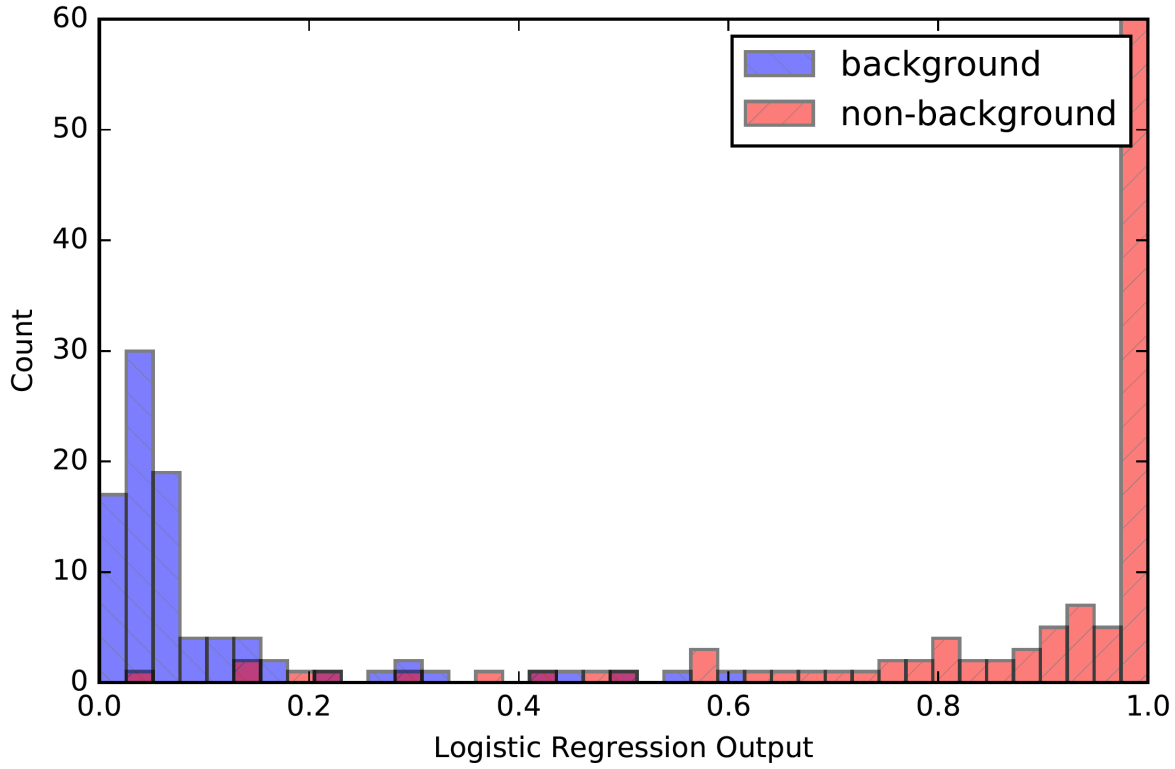


Figure 9: An illustration of the separation learnt using a logistic regression model on a small set of 90 background and 110 non-background samples.

the DCN is expressed as:

$$P(c|y_{DCN}, y_{BGM}) = \frac{P(y_{DCN}, y_{BGM}|c)P(c)}{P(y_{DCN}, y_{BGM})}, \tag{6}$$

where $P(c)$ is the class prior and $P(y_{DCN}, y_{BGM})$ is the joint probability of the observation. When selecting the most likely class, the joint probability of observations can be ignored since it is common for all classes c_i.

$$c^* = argmax_i[P(y_{DCN}, y_{BGM}|c_i)P(c_i)]. \tag{7}$$

4 Experiments

4.1 Mining Dataset

The dataset use for evaluating this work was collected from a camera mounted to a light vehicle operating in an active mine-site, see Figure 10. While the motivation is to put vision based sensing on a heavy vehicle, a light vehicle is more practical for gathering a diverse set of visual sequences. The cameras used in this study were 0.9 MP BlackFly cameras produced by PointGrey. The frame-rate was fixed to 10 Hz and maximum exposure time set to 20 ms to prevent blur. The lens focal length was set to 1.8 mm, translating a to 94 degree horizontal field of view. The dataset contains both static and dynamic instances of a `person`, `LV` or `HV`.

Figure 10: The experimental dataset gathering vehicle with cameras mounted to the bullbar. Note: all images used in this paper were captured from the camera on the left hand side of the vehicle.

Continuous video was gathered with and without the camera in motion and on various haul roads and a few light vehicle only zones to capture variation in the environment. This video data was captured at 10 Hz and partitioned into various sequences. In this work we use 5 sequences where no people or vehicles are visible to build our background model. Collectively these background sequences make up 8952 frames in total (approximately 14 km). These sequences are referred to as Train 1-5 in Table 1.

To evaluate the performance we use another 6 sequences with several instances of `person`, `LV` or `HV`, that were manually annotated using the tool developed by (Vondrick et al., 2012). These annotated sequences contain 11150 frames in total (approximately 11 km) and referred to as Test 1-6 in Table 1.

In addition to the train and test sequences captured in the field, we made a small validation set of 200 manually cropped images containing mining objects. These validation images were collected from various sources including the internet along with a few captured at night from the same mine site but in different locations to the test sequences. This validation set was used throughout the design process to generate Figures 4, 7, 9, 11, and Figure 12. Table 2 lists the parameters of the system with links to the empirical studies performed in this work and the dataset sequences used. Note that the study performed for selecting the NMS did use the test sequences since they are the only sequences available with substantial set of annotated mining object classes in real life scenarios. However, since the same NMS is applied to all the classifier models we consider the later comparisons in this experimental section to remain valid.

Table 1: Experimental Image Sequences

Name	Total Frames	Purpose	Environment	Conditions	Content
Train 1	2433	Learning Clusters	Haul road between pits	Morning, Sunny	BG Only
Train 2	1975	Learning Clusters	In-pit haul road	Midday, Sunny	BG Only
Train 3	1500	Learning Clusters	In-pit haul road	Afternoon, Semi-overcast	BG Only
Train 4	1669	Learning Clusters	Haul road between pits	Afternoon, Semi-overcast	BG Only
Train 5	1375	Learning Clusters	Haul road between pits	Evening, Overcast	BG Only
Validation	200	Parameter Tuning	Hand cropped mining images from various sources[a]	Day and night	BG, People, LV, HV
Test 1	1463	Evaluation	Store yard and processing plant	Morning, Sunny	People, LV
Test 2	2951	Evaluation	In-pit, haul road and LV area	Midday, Sunny	People, LV, HV
Test 3	600	Evaluation	Haul road between pits	Midday, Sunny	People, LV
Test 4	2827	Evaluation	In-pit and haul road	Midday, Sunny	LV, HV
Test 5	1569	Evaluation	LV only area	Evening, Overcast	People, HV
Test 6	1740	Evaluation	LV/HV parking area	Night, Overcast	People, LV, HV

[a] The 200 validation images are made up from both internet images and several challenging background and night images which were collected in the field but outside of the test sequences 1-6.

4.2 Background Model Validation

Here we describe the experiments performed to design our background modelling system explained in the previous section. From the 5 background sequences, we applied the region proposal and remapped DCN detection framework to find challenging region proposals from every tenth frame. While some of the false objects may be observed in multiple frames, the time difference is sufficient to capture a variety of view points for these distracting objects. We lowered the detection threshold to collect region proposals if the remapped DCN predicted either a `person` or `car` in the top 5 out of 200 ImageNet class responses. With this configuration we collect around 8000 hard negatives for our background reservoir. We held out 90 of the most interesting background regions and added them to the validation set.

To address the design decisions for the background cluster model, we perform an empirical study using the reservoir containing only negatives and the validation set with both negative and positives. We jointly test

Table 2: System Parameters

Parameter(s)	Values	Tuning Method	Data Used
EB Max Count	1000	Assumed	None
NMS Overlap	0.5	Empirical (see Fig. 3)	Test 1-5
DCN Remapping	*	Empirical (see Fig. 4)	Validation
DCN Retraining	*	SGD using classification loss	Validation
BGM Clusters (k)	128	Empirical k-means (see Fig. 11)	Train 1-5
BGM Euclidean (α, β)	(0.015,0.111)	Empirical (see Fig. 8)	Validation
BGM Cosine (θ, b)	(*,-38.9)	SGD with 10 fold cross validation	Validation
Fusion prior ($BG, Person, LV, HV$)	(0.7,0.1,0.1,0.1)	Assumed	None

* indicates a high dimensional parameter set.

different combinations of DCN layer features and number of clusters by evaluating their performance on the validation set. For the distance threshold we set this to the distance corresponding to a 95% recall on the positive set. With the recall fixed, the overall performance of the background model is measured by the precision at which it can identify a true negative.

In all experiments we use the DCN structure of (Krizhevsky et al., 2012), which is an eight layer network with five convolutional layers, followed by three fully connected layers. The last layer was adapted to the detection task (Girshick et al., 2014) with 200 outputs corresponding to the ImageNet detection dataset. For extracting a feature to describe the background appearance, we consider the response of layers in the network. Figure 11 shows the relative performance of the different DCN layers when sweeping over different background model sizes measured by the number of clusters used. This experiment shows that the `fc6` layer exhibits substantially higher precision across all model sizes. This could be put down to its position in the network, where it is the first layer to incorperate the global visual information, as opposed to the convolutional layers. These findings are in agreement with a recent related study for investigating the performance of transfering layers where the 6th layer regularly achieves the highest performance on the target task (Azizpour et al., 2015).

Figure 11 shows that each layer produces a dog-leg shaped curve with a common turning point at 128 clusters. This signifies a point where the diversity of the environment is sufficiently represented. While increasing the number of clusters beyond 128 clusters continues to improve precision, the rate of improvement is small. E.g. for 16x more clusters the improvement is only 1% (89% for 128 versus 90% for 2048 clusters) compared to a 3% drop when using 16x fewer clusters (86% for 8 clusters). In terms of processing time, both the nearest neighbour lookup for the k-means Euclidean model and the cosine similarity model is linear in the number of clusters [2]. Since the `fc6` layer with 128 provides a reasonable trade-off between precision and computation speed, these parameters are used in all remaining experiments. A detailed view of the distances between the validation samples and the nearest cluster centre using these parameters can be seen in Figure 7.

Figure 12 illustrates the samples in this validation set with the largest error. The false negatives are mostly night images which can be put down to the fact that similar images are rare if not non-existent in the ImageNet samples used to train the DCN. For the false positives, these are mostly signs which make up a minority of the scene. From these samples we can describe our background model as a form of novelty detection where interesting parts of the scene such as signs are distinguished from the general background. This finding along with the unsupervised clustering shown in Figure 6 are a testament to the DCN's expressive capabilities in representing visual similarity.

For the cosine based background model, `fc6` features were used and the number of clusters was set to 128, matching the Euclidean k-means model. The logistic regression was trained using stochastic gradient descent optimisation with regularisation penalty selected though 10-fold cross validation. The resulting learnt separation between background and non-background was previously shown in Figure 9.

4.3 Detection Evaluation

The system is evaluated on a set of five daytime sequences and one night sequence, where the task is to detect and locate both people and vehicles within each frame. In this evaluation we follow the same criteria as described in (Bewley and Upcroft, 2015), where we consider a true detection if at least 50% of the detection region is covered by a single ground truth object. This differs from the intersection-over-union (IOU) definition of overlap, as we accept detecting a `person`'s head and shoulders without their whole body while IOU would count this as both a miss detection and a false positive. Additionally, if multiple detections overlap a single ground truth instance, we count this as a single true positive and neither of the overlapping detections are false. A concrete example would be if a person's head is covered by a single detection and their body another. In such a scenario, both are considered valid detections but only count as a single true

[2] Efficient data structures like kd-trees where considered but due to the high dimensionality the practical improvement was negligible.

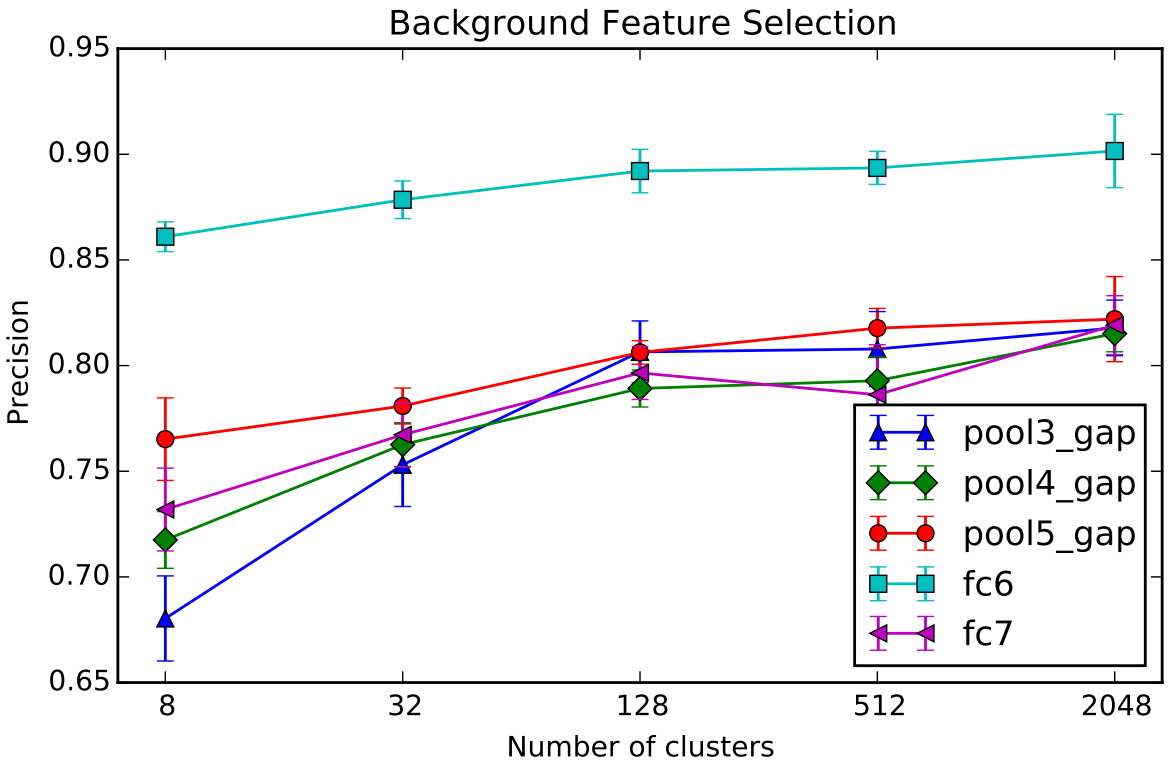


Figure 11: Cross-validation precision at a fixed recall of 95% for different DCN layers and the number of clusters used to represent the background. Each point shows average of 5 trials. Pool 3-5 are derived from convolutional layers 3-5 respectively while the fc6 and fc7 are direct copies of the fully connected layer responses. Note: the x-axis is in log_2 scale.

positive per object. It should be also noted that any detection or miss detection of an object labelled as partially occluded in the ground truth is ignored in this evaluation. Only the retrained model and the fused models are capable of detecting `HV` so were omitted from the valuation across the different models for fair comparison. To accommodate this, any detections which overlap with `HV` objects are considered as neither true or false and are excluded from the evaluation.

Figure 13 shows both a failure example on the left and valid detections on the right for each sequence. Sequence 1 (top row), is located near a store yard and a processing plant where there are a number of regions with varying appearance making this sequence particularly challenging for the background model. This sequence has a lower precision as there are a number of novel regions in this environment that mitigate the benefits provided from the background model trained on common background sequences. The *remapped* DCN model is less effected by this as some of the 200 ImageNet classes contain sufficient diversity to compete with the `person` and `LV` class responses. The second sequence was captured around an in-pit go-line, where mine workers transition between operating LV and HV. In this environment, the fused model performs exceptionally well in detecting `person` and `LV`, however, `HV` would occasionally be missed or misclassified as `LV`. The third and forth sequences were captured along main haul roads, between pits and in the pit (respectively), representing the typical operating environments for `HV`. In this environment the fusion of the background model nearly elimintates all the false positives caused by trees and other common road side objects. However, novel objects such as signs and cones are sometimes detected (e.g. in the fourth row of Figure 13 a distant "Reduce Speed" sign is mistaken for a `person`). The fifth sequence was taken in the late afternoon in semi-overcast weather conditions and consists of multiple mine workers in view of the

Figure 12: Validation samples where the nearest k-means background cluster model failed. Images are shown in their warped form, representing the DCN input. The four right false negatives were collected at night.

camera. The final sequence was shot at night around the mine's LV parking area where significant activity was observed – coinciding with a shift change. Under these conditions the fused model is able to detect both people and LV while it also appears to have learnt to distinguish between yellow lights mounted to vehicles and white lights such as the bright spot corresponding to a saturated retro-reflective sign to the left of the distance LV in the bottom right image of Figure 13.

Table 3 shows a detailed breakdown in the performance of all individual components and the fused method. Here we have separated the performance of each component to isolate the behaviour of the DCN and the background models (BGM). For all models the detection probability threshold was set to 0.5, except the (Bewley and Upcroft, 2015) model which was based on a Euclidean distance to achieve 95% recall on the validation set. The DCN output generally has a high recall but low precision as the DCN does not consider the class prior probability.

The two BGM models behave considerably differently from each other. Particularly the model based on the Euclidean distance to the nearest neighbour presents high recall as this was by design in the threshold selection. However it is interesting that the performance in both precision and recall significantly dropped compared to the held out validation set used in tuning the distance distribution parameters. On the other hand, the Cosine based BGM has significantly improved the recall, resulting in an overall F1 score that is compatible to the DCN output. Despite using a roughly assumed class prior, the soft fusion between the retrained DCN and cosine BGM, enables the fused model to take advantage of both the high DCN recall and the BGM's precision to produce an overall top F1 score.

5 Conclusions and Future Work

Visual object detection has the potential to make a significant impact to improving safety in the mining industry. The goal of this work is to address the gap in situational awareness of heavy mining vehicles to sense other vehicles and personnel without active transponders. Towards achieving that goal, this paper presented a passive, vision-only detection system that takes advantage of recent developments in computer

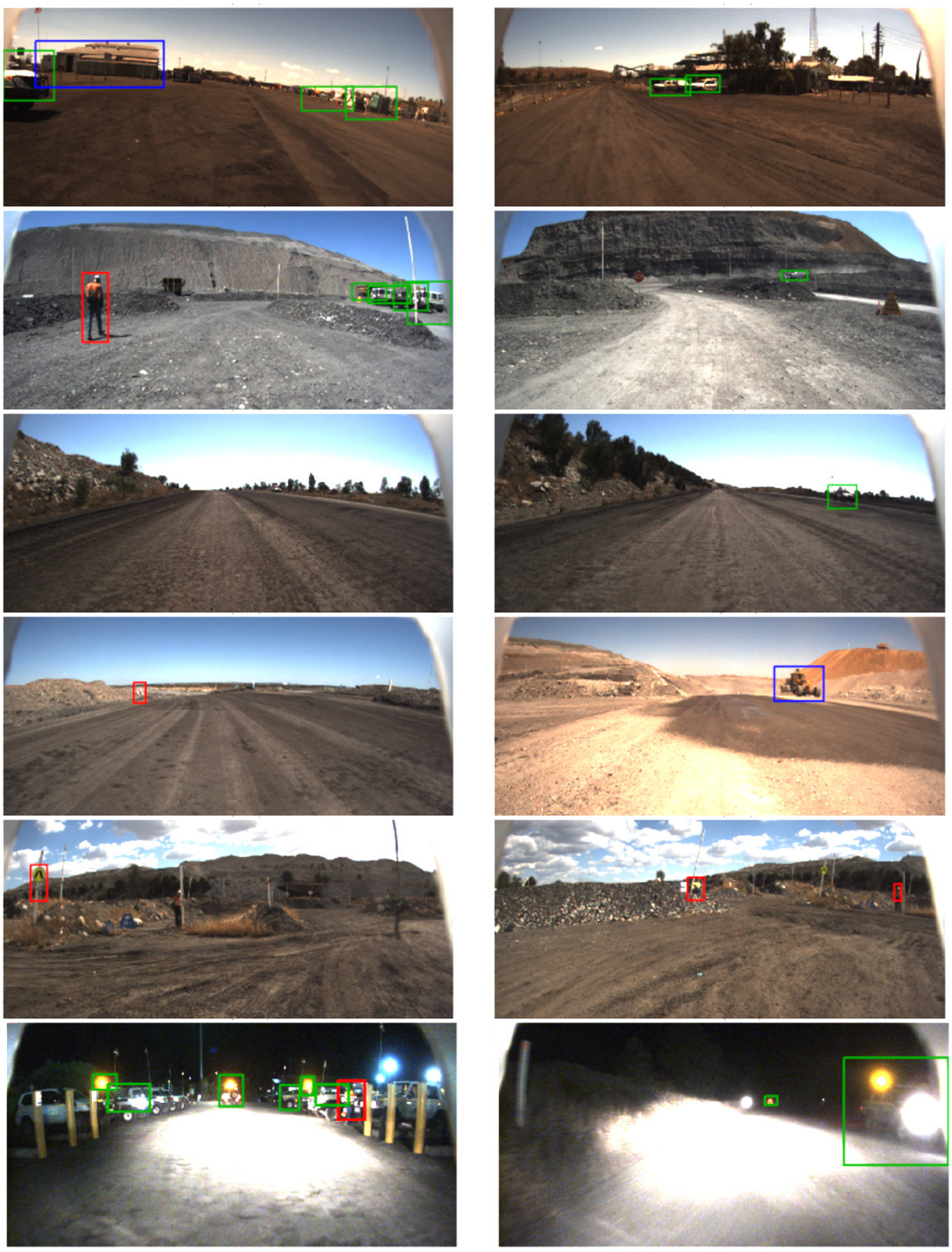


Figure 13: Exemplar images of *fused* model with images containing at least one false positive or miss detection in the left column and only true positives on the right. Each row represents the sequence order 1 to 6 corresponding to the results in Table 3. The colours denote the predicted class of the object with: *red* for Person, *green* for LV and *blue* for HV. Note: detecting and recognising HV is attributed to the *retrained* component of the *fused* model.

Table 3: Detection Performance on Mining Sequences
F1 (Precision, Recall)

Sequence (frames)	DCN Remapped[a]	DCN Retrained	BGM Euclidean	BGM Cosine	Fused
Test 1	**0.33** (**0.24**,0.54)	0.17 (0.10,**0.72**)	0.07 (0.04,0.67)	0.23 (0.14,0.57)	0.27 (0.17,0.67)
Test 2	0.81 (0.72,0.93)	0.80 (0.68,**0.98**)	0.22 (0.12,0.92)	0.73 (0.69,0.78)	**0.90** (**0.87**,0.94)
Test 3	0.04 (0.02,0.46)	0.09 (0.05,**0.68**)	0.01 (0.01,0.55)	0.12 (**1.00**,0.06)	**0.69** (0.90,0.56)
Test 4	0.26 (0.15,0.89)	0.42 (0.27,**0.96**)	0.10 (0.05,0.93)	0.56 (0.46,0.73)	**0.68** (**0.58**,0.82)
Test 5	0.43 (0.33,0.62)	0.72 (0.58,**0.93**)	0.25 (0.15,0.89)	0.60 (0.79,0.48)	**0.83** (**0.88**,0.78)
Test 6	0.57 (**0.54**,0.60)	**0.61** (0.49,0.83)	0.47 (0.32,**0.86**)	0.60 (0.48,0.80)	0.59 (0.49,0.75)
Overall	0.475 (0.39,0.75)	0.541 (0.42,**0.90**)	0.202 (0.12,0.86)	0.550 (0.56,0.66)	**0.692** (**0.65**,0.80)

[a] This is a soft probabilistic version of the method presented in (Bewley and Upcroft, 2015).
Bold indicates the best performance across models.

vision and machine learning to detect both personnel and mining vehicles. This sensing approach was evaluated in an active open-pit mine site environment across six different parts of the mine and at both day and night. Challenges around over-fitting on a small dataset were addressed by exploring a fusion framework incorporating a pre-trained DCN and exploring both remapping and retraining the final layers for adapting to the mining environment.

The experiments show that the in-pit environment is suitable for vision based detection utilising object proposals and DCNs, along with background modelling techniques. The experiments also show that the amount of NMS applied to the proposed regions can increase the proposal recall compared to the same number of proposals selected using the proposal score. However, it was also shown that this improvement has limited range in the amount of NMS applied. This characteristic, to a degree, could be contributed to the differences in the mining environment compared to the typical internet based images used in the development of the `EdgeBox` proposal method used in this work.

When applying an off-the-shelf DCN pre-trained for the ImageNet detection task to mining imagery we highlight that it performs poorly on both the background and the mining specific `HV` class. While this method has the advantage of retaining knowledge of the large amounts of data from ImageNet for the common classes, it is restricted in its ability to distinguish new and novel classes. On the other hand, retraining the final layers using mining images overcomes this limitation allowing the `HV` class to be distinguished from the `background`, `person` and `LV` classes. However, with the limited amount of labelled training data available, this retrained model is comprable to simply remapping the classes from the pre-trained ImageNet model.

The use of a DCN alone for visual object detection is shown to perform poorly when presented with typical mining imagery such as a haul road environment. To overcome this limitation, two variants of a background model are presented in this work for the purpose of suppressing the false positives produced by the DCN. The soft probabilistic variant of the background model when fused with the DCN output offers superior performance over the logic based combination previously used for background suppression in (Bewley and Upcroft, 2015). Quantitatively, the fused model presented achieves a relative improvement in F1 score of 46% over a pre-trained DCN and 28% over a DCN retrained with mining images.

This article also presents an improved variant of the background model shown to have several desirable attributes. Firstly, the visual feature descriptor is already computed as part of the DCN recognition. Secondly, it only required the bare minimum labelling effort to build a large reservoir of background only samples. It is made robust to the high dimensional nature of the DCN features by using the cosine similarity followed by logistic regression. All its parameters can be estimated from a small set of labelled images. Also the cosine variate provides superior background discrimination over the simpler k-means model, achieving a detection accuracy comparable to the best DCN baseline model on its own. Moreover, the higher precision of the cosine background model makes it complementary to the high recall DCN when used in a fusion framework. However, in sequences where the background differed significantly from the background only training sequences its performance dropped considerably. This combined with the use of a static offline trained background limits this approach to having either a comprehensive training set or restricting the detection to similar environments, e.g. only haul roads. Future work in addressing this issue could include learning the background cluster online by incorporating additional information from other sources.

The Bayesian fusion framework for combining the outputs of the DCN and the background model is basic and assume independence. Despite its simplicity, the improvement in F1 score of the fused model over its counterparts suggest there is value in combining these methods. Future work could include alternative fusion techniques such as cascade architectures to improve computational efficiency or introducing other sensor information.

Another characteristic of this work is that it does not retain any temporal information. In particular the feed-forward architecture of the DCN treats each frame in the video like it is seeing it for the first time. An advantage of this is that the framework is robust to motion experienced by either the object or camera by performing the detection of each frame independently. The disadvantage of this is that temporal memory is useful in video as objects move gradually in the image which can be exploited through tracking techniques to improve recall. Also, while this work is only concerned with single camera based sensor data we see many opportunities to combine techniques incorporating, motion segmentation (Bewley et al., 2014), odometry (Hawke et al., 2015), stereo (Bewley and Upcroft, 2013) or range-based sensors for improved robustness. Additionally, as more labelled mining image data becomes available we expect to be able to design and fine-tune a DCN that performs better in this domain than the existing network designed for the ImageNet benchmark.

Acknowledgments

This research was funded by the Australian Coal Association Research Program (ACARP). The authors would also like to acknowledge Anglo American for allowing data collection at the Dawson operation. Acknowledgement also goes to the high performance computing group at Queensland University of Technology for both support and use of their services when conducting the experiments in this paper.

References

Aggarwal, C. C., Hinneburg, A., and Keim, D. A. (2001). On the Surprising Behavior of Distance Metrics in High Dimensional Space. In *International Conference on Database Theory*, pages 420–434. Springer Berlin Heidelberg.

Ahmed, A., Yu, K., Xu, W., Gong, Y., and Xing, E. (2008). Training Hierarchical Feed-Forward Visual Recognition Models Using Transfer Learning from Pseudo-Tasks. In *European Conference on Computer Vision (ECCV)*, pages 69–82. Springer Berlin Heidelberg, Berlin, Heidelberg.

Aytar, Y. and Zisserman, A. (2011). Tabula Rasa: Model Transfer for Object Category Detection. In *International Conference on Computer Vision (ICCV)*.

Azizpour, H., Razavian, A. S., Sullivan, J., Maki, A., and Carlsson, S. (2015). From Generic to Specific

Deep Representations for Visual Recognition. In *Computer Vision and Pattern Recognition (CVPR) Workshops*.

Benenson, R., Mathias, M., Tuytelaars, T., and Van Gool, L. (2013). Seeking the strongest rigid detector. In *Computer Vision and Pattern Recognition (CVPR)*.

Bewley, A., Guizilini, V., Ramos, F., and Upcroft, B. (2014). Online Self-Supervised Multi-Instance Segmentation of Dynamic Objects. In *International Conference on Robotics and Automation*, pages 1296–1303, Hong Kong, China. IEEE.

Bewley, A. and Upcroft, B. (2013). Advantages of Exploiting Projection Structure for Segmenting Dense 3D Point Clouds. In *Australian Conference on Robotics and Automation*.

Bewley, A. and Upcroft, B. (2015). From ImageNet to Mining : Adapting Visual Object Detection with Minimal Supervision. *Proceedings of the 10th International Conference on Field and Service Robotics (FSR)*.

Dalal, N. and Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. Ieee.

Dempster, A. P. (2008). A Generalization of Bayesian Inference. In *Classic Works of the Dempster-Shafer Theory of Belief Functions*, pages 73–104. Springer Berlin Heidelberg, Berlin, Heidelberg.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Dollar, P., Appel, R., Belongie, S., and Perona, P. (2014). Fast Feature Pyramids for Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1532–1545.

Dollár, P., Wojek, C., Schiele, B., and Perona, P. (2009). Pedestrian detection: A benchmark. *2009 IEEE Conference on Computer Vision and Pattern Recognition*.

Donahue, J., Hoffman, J., Rodner, E., Saenko, K., and Darrell, T. (2013). Semi-supervised Domain Adaptation with Instance Constraints. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 668–675.

Durrant-Whyte, H. and Henderson, T. C. (2008). Multisensor Data Fusion. In *Springer Handbook of Robotics*, pages 585–610. Springer Berlin Heidelberg, Berlin, Heidelberg.

Farabet, C., Couprie, C., Najman, L., and LeCun, Y. (2013). Learning Hierarchical Features for Scene Labeling. *Pattern Analysis and Machine Intelligence*, 35(8):1915–1929.

Fei-Fei, L., Fergus, R., and Perona, P. (2007). Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–45.

Felzenszwalb, P. F. and Huttenlocher, D. P. (2004). Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, 59(2):167–181.

Girshick, R. B., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR)*.

Hawke, J., Gurau, C., Tong, C. H., and Posner, I. (2015). Wrong Today , Right Tomorrow : Experience-Based Classification for Robot Perception. In *Field and Service Robotics*.

Hosang, J., Benenson, R., and Schiele, B. (2014). How good are detection proposals, really? In *British Machine Vision Conference (BMVC)*.

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. In *International Conference on Multimedia*, pages 675–678.

Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages (Vol. 1, No. 2, p. 4).

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*, 1:541–551.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision (ECCV)*, volume 8693, pages 740–755. Springer International Publishing.

Marshall, J. A. and Barfoot, T. D. (2008). Design and field testing of an autonomous underground tramming system. *Springer Tracts in Advanced Robotics*, 42:521–530.

McMahon, S., Sünderhauf, N., Upcroft, B., and Milford, M. (2015). How Good Are EdgeBoxes, Really. In *In Workshop on Scene Understanding (SUNw), Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Mosberger, R. and Andreasson, H. (2012). Estimating the 3d position of humans wearing a reflective vest using a single camera system. In *International Conference on Field and Service Robotics (FSR)*.

Mosberger, R., Andreasson, H., and Lilienthal, A. J. (2013). Multi-human Tracking using High-visibility Clothing for Industrial Safety. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 638–644.

Oquab, M., Bottou, L., Laptev, I., and Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR)*.

Phillips, T., Hahn, M., and McAree, R. (2013). An evaluation of ranging sensor performance for mining automation applications. In *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics: Mechatronics for Human Wellbeing, AIM 2013*, pages 1284–1289.

Razavian, A. S., Azizpour, H., Sullivan, J., and Carlsson, S. (2014). CNN Features off-the-shelf: an Astounding Baseline for Recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Roberts, J. M. and Corke, P. I. (2000). Obstacle detection for a mining vehicle using a 2D laser. In *Proceedings of the Australian Conference on Robotics and Automation*, pages 185–190.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and LeCun, Y. (2014). OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *International Conference on Learning Representations (ICLR 2014)*.

Sermanet, P., Kavukcuoglu, K., Chintala, S., and LeCun, Y. (2013). Pedestrian Detection with Unsupervised Multi-stage Feature Learning. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3626–3633. IEEE.

Simonyan, K. and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv technical report*.

Sünderhauf, N., Shirazi, S., Dayoub, F., Upcroft, B., and Milford, M. (2015). On the Performance of ConvNet Features for Place Recognition. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going Deeper with Convolutions. In *Computer Vision and Pattern Recognition (CVPR)*.

Torralba, A., Fergus, R., and Freeman, W. (2008). 80 Millions Tiny Images: a Large Dataset for Non-Parametric Object and Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:1958–1970.

Uijlings, J. R. R., Sande, K. E. A., Gevers, T., and Smeulders, A. W. M. (2013). Selective Search for Object Recognition. *International Journal of Computer Vision*, 104(2):154–171.

Vondrick, C., Patterson, D., and Ramanan, D. (2012). Efficiently Scaling up Crowdsourced Video Annotation. *International Journal of Computer Vision*, 101(1):184–204.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks ? In *Advances in Neural Information Processing Systems (NIPS)*.

Zeiler, M. D. and Fergus, R. (2013). Stochastic Pooling for Regularization of Deep Convolutional Neural Networks. In *International Conference on Learning Representations*.

Zhang, S., Benenson, R., and Schiele, B. (2015). Filtered Channel Features for Pedestrian Detection. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Zitnick, C. and Dollár, P. (2014). Edge Boxes: Locating Object Proposals from Edges. In *European Conference on Computer Vision (ECCV)*.

RESEARCH STUDENTS CENTRE
Examination Enquiries: 07 3138 1839
Email: research.examination@qut.edu.au

Statement of Contribution of Co-Authors for Thesis by Published Paper

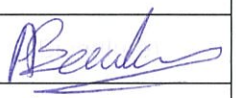
The following is the format for the required declaration provided at the start of any thesis chapter which includes a co-authored publication.

The authors listed below have certified* that:

1. they meet the criteria for authorship in that they have participated in the conception, execution, or interpretation, of at least that part of the publication in their field of expertise;
2. they take public responsibility for their part of the publication, except for the responsible author who accepts overall responsibility for the publication;
3. there are no other authors of the publication according to these criteria;
4. potential conflicts of interest have been disclosed to (a) granting bodies, (b) the editor or publisher of journals or other publications, and (c) the head of the responsible academic unit, and
5. they agree to the use of the publication in the student's thesis and its publication on the QUT ePrints database consistent with any limitations set by publisher requirements.

In the case of this chapter:

Publication title and date of publication or status:
"From ImageNet to Mining: Adapting Visual Object Detection with Minimal Supervision" (published June, 2015)

Contributor	Statement of contribution
Alex Bewley  8th April 2016	Wrote the manuscript, designed and conducted experiments and performed data analysis
Ben Upcroft	Provided input into scope of paper and proofreading/editing

Principal Supervisor Confirmation

I have sighted email or other correspondence from all Co-authors confirming their certifying authorship.

Ben Upcroft		15/4/16
Name	Signature	Date

From ImageNet to Mining: Adapting Visual Object Detection with Minimal Supervision

Alex Bewley and Ben Upcroft

Abstract This paper presents visual detection and classification of light vehicles and personnel on a mine site. We capitalise on the rapid advances of ConvNet based object recognition but highlight that a naive black box approach results in a significant number of false positives. In particular, the lack of domain specific training data and the unique landscape in a mine site causes a high rate of errors. We exploit the abundance of background-only images to train a k-means classifier to complement the ConvNet. Furthermore, localisation of objects of interest and a reduction in computation is enabled through region proposals. Our system is tested on over 10km of real mine site data and we were able to detect both light vehicles and personnel. We show that the introduction of our background model can reduce the false positive rate by an order of magnitude.

1 Introduction

While the mining industry pushes for greater autonomy, there still remains a need for human presence on many existing mine sites. This places significant importance on the safe interaction between human occupied and remotely operated or autonomous vehicles. In this work, we investigate a vision based technique for detecting other vehicles and personnel in the workspace of heavy vehicles such as haul trucks.

Traditionally, methods for detecting light vehicles and personnel from heavy mining equipment have relied on radio transponder based technologies. Despite

Alex Bewley
School of Electrical Engineering and Computer Science, Queensland University of Technology, Brisbane, Australia, e-mail: aj.bewley@qut.edu.au

Ben Upcroft
ARC Centre of Excellence for Robotic Vision, School of Electrical Engineering and Computer Science, Queensland University of Technology, Brisbane, Australia. http://www.roboticvision.org/, e-mail: ben.upcroft@qut.edu.au

transponder based sensors being mature and reliable for ideal conditions, in practise their reliability is circumvented by practical issues around their two way active nature, portable power requirements, limited spatial resolution and human error. Using computer vision offers a unique alternative that is passive and readily available on existing remotely operated vehicles.

Vision based object recognition has made tremendous progress as measured by standard benchmarks [4, 16]. The major advancements in this area can be attributed to both the availability of huge annotated datasets [7, 26, 4, 16] and developments in data driven models such as deep convolutional networks (ConvNets) [13, 24]. In this work we utilise the ConvNet of [13] which has shown astonishing performance on the ImageNet recognition benchmark [4] and extend it to data collected from mine sites with minimal training.

Using ConvNets in different domains requires a large training set relevant to the target task [29]. When the amount of training data is small, data driven approaches tend to over-fit the training samples and not generalise to unseen images. In this work we utilise a pre-trained ConvNet using millions of images from ImageNet and address how to map the original ImageNet classes to mining classes with minimal training effort.

Another consideration regarding this application is that cameras are rigidly coupled to the vehicles orientation and configured with a fixed focal length. This distinguishes it from the ImageNet recognition problem where typical images collected were implicitly pointed at regions of interest and appropriately zoomed. Additionally, due to the wide field of view the majority of the images are background with zero to potentially multiple objects of interest visible in any given frame. To locate the objects, we follow a similar strategy to [10] and apply an initial step for finding likely object locations through a region proposal process before performing object recognition with the ConvNet.

Given that the majority of the images collected in a mine site dataset have zero objects of interest in them, we can provide a standard classifier with a huge amount of labelled background data. Using this newly trained classifier in conjunction with the ConvNet ensures robustness and drastically reduces spurious detections. This classifier is based on k-means clustering offering a convenient way to partition the background data into different categories. This approach accurately captures the characteristics of the background, enabling the discovery of novel non-background objects.

The contributions of this paper are:

- adapting ConvNets to new scenes in a mining context,
- complementing the powerful classification provided by ConvNets with a simple classifier trained on background mine data for increased robustness,
- a novelty detector using ConvNet feature clustering.

This paper is organised with a short review of related literature before describing the proposed method in greater detail. We then analyse the performance of the proposed method on a challenging set of mining videos and conclude with a discussion of the learnt outcomes and avenues for future improvement.

2 Related Work

Here we briefly review object detection methods that are not reliant on two way communication before covering some related work using ConvNets for generic object detection. Early work has focused on range based techniques such as LiDAR [22, 17] commonly used for mapping fixed obstacles such as buildings or underground tunnel walls. Applying these sensors to detecting personnel and vehicles fitted with retro-reflectors, is found to be sensitive to the dynamics of the sensor platform [20]. In this work we focus specifically on detecting potentially dynamic obstacles including vehicles and particularly people from vision based data. To this end, the more relevant prior work is that of [18] which exploits the standardised requirement for personnel on mine sites to wear high-visibility clothing equipped with retro-reflector strips. This enables a single IR camera with active flash to highlight personnel in view which can then be used for tracking [19].

Recent popularity of big data and deep learning have dominated the object recognition problem. Among these data driven approaches, deep convolutional neural networks (ConvNets) with recognition performance quickly approaching human levels [13, 5, 21, 23] are selected for use in this work. ConvNets themselves have been used for over 20 years [14] for tasks such as character recognition. Over recent years ConvNets have made an astonishing impact on the computer vision community [13, 6, 21, 10, 5] thanks to the availability of huge labelled image sets such as ImageNet [3].

Recognising what objects are in an image is only half of the object detection problem. The other half is locating the objects within the image. Sermanet et al. [23] sample over multiple scales and exploit the inherently spatially dense nature of the convolutions within ConvNets to identify regions with high responses. Similarly, [6] also perform convolutions over multiple scales and combine the responses over superpixel segmentation [9]. Another popular approach and the one that we base this work off is the region convolutional neural network (RCNN) of [10]. The RCNN framework efficiently combines the ConvNet of [13] with an object proposal method: selective search [27]. Generic object proposal methods aim to efficiently scan the entire image at different scales and aspect ratios to reduce potentially millions of search windows down to hundreds [11] of the most likely candidates. In this work we use edge box object proposals [30] as the accuracy is higher while also running at an order of magnitude faster [11].

3 Methodology

In this section we outline our detection pipeline and how it differs from [10]. Our method consists of three key phases: 1) Region proposals with non-maximum suppression (NMS), 2) ConvNet recognition and finally, 3) Detections are validated by checking for novelty against the background model. See Fig. 1 for a high-level overview of this pipeline. We bypass the problem of over-fitting on a small dataset

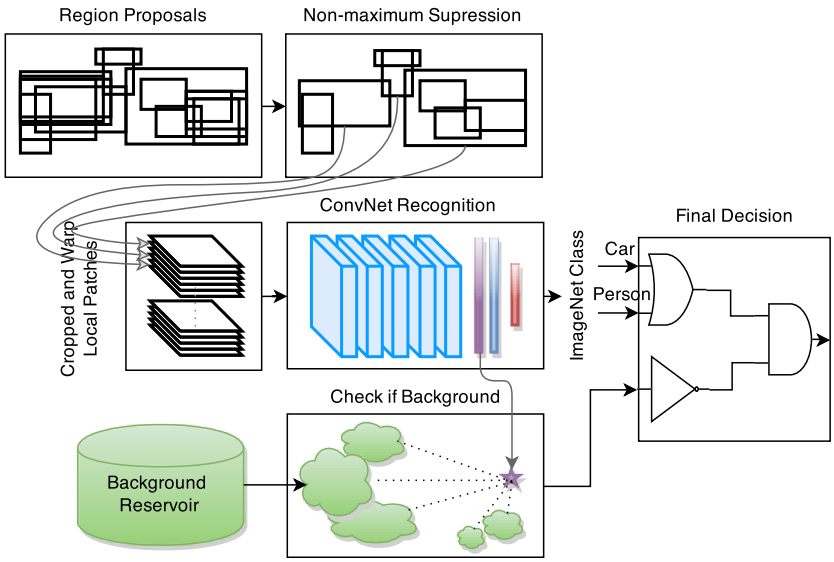


Fig. 1 An illustration of the detection pipeline used in this work. The system parameters are highlighted in blue and green which are learnt offline from an off-the-shelf network and background only images respectively. Note the red output layer of the ConvNet outputs are ImageNet classes (200 different). Any car or person is suppressed if it also matches the background model to minimise the number of false positives.

by using a pre-training ConvNet and map its output to mining relevant classes. This method is then extended with our proposed background modelling technique to significantly reduce the number of false positives generated by the system.

3.1 Region Proposals

The aim of region proposals is to efficiently scan the image to eliminate millions of potential windows, keeping only the regions that are likely to contain an object of interest. We use the `EdgeBoxes` region proposal method [30] over the `selective search` [27] used in the original RCNN work as this method is orders of magnitude faster with comparable accuracy. For a detailed comparison of region proposal methods we refer the reader to [11].

The default parameters for `EdgeBoxes` were adjusted to return a fixed 1000 proposals. These region proposals are then further reduced to approximately 100 regions through a process of non-maximum suppression (NMS). The NMS process considers the score produced by the `EdgeBoxes` method and the overlap with other bounding boxes. As the name suggests it then greedily suppresses all but the

maximum scoring proposal for all adjacent regions overlapping by 30% or more. In contrast to applying NMS after the ConvNet [10], this way we can speed up the detection pipeline by reducing the number of proposals going into the ConvNet while maintaining comparable coverage over the image.

3.2 Region Classification

Having selected regions of the image that have the general characteristics of an object, we now perform object recognition to distinguish the object category. For this we apply the ConvNet from RCNN [10] which is based on the winning architecture [13] for the ImageNet Large Scale Recognition challenge in 2012. For this work, we used the RCNN implementation provided with the Convolutional Architecture for Fast Feature Embedded (`caffe`) [12] framework out-of-the-box.

The original detection task for RCNN was to predict one of 200 classes that represent common objects found in images taken from the internet. For this application we are only interested in distinguishing between three high level categories, namely: `background`, `person` and `light vehicles(LV)`. Using this model in a mining context raises several issues that need addressing:

1. Most of the 200 classes are irrelevant, e.g. jellyfish, miniskirt, unicycle etc.
2. How to associate mining classes with ImageNet classes?
3. Semantically the `background` is significantly different from many of the existing object specific classes.

To gain some insight, we use a small validation set of 200 images to investigate the output of the ConvNet out-of-the-box. This set is made up of cropped mine-site images containing the classes `person` and `LV` along with 90 interesting region proposals extracted from background only images. We also included a few `heavy vehicles (HV)` images in this set but keep them as a separate class to identify any correlations. In Fig. 2 we show the results of naively applying the pre-trained RCNN model to this image set. To better visualise the output we applied a soft-max transform to approximate the output class prediction as a probabilistic estimate. [1]

Not surprisingly, the `person` and `LV` classes are well represented and can be directly mapped from the `person` and `car` ImageNet classes used to train the original ConvNet. On the other hand, the `background` closely resembles uniform random sampling of classes as there are no relevant classes in the existing model such as trees, buildings, or road signs etc. Similarly, the `HV` class prediction also mostly resembles a uniformly random distribution with a slight bias towards the ImageNet classes `snowplow`, `cart` and `bus`. As for this application, we are only

[1] It is important to note that this is for visualisation purposes only and that the y-axis does not represent the true probability since the final SVM layer of RCNN was not calibrated for probabilistic outputs.

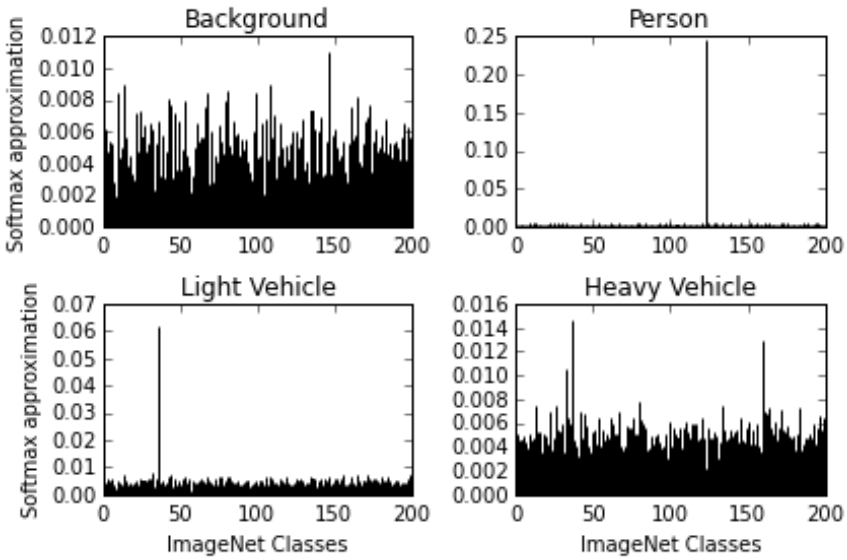


Fig. 2 The average class estimate for a set of mining related images. Notice that person (class 123) and light vehicle/car (class 36) are existing classes for the pre-trained network and can be used directly. The background and the heavy vehicle classes are novel and show a wider spread as they are not modelled with the pre-train ConvNet.

concerned with distinguishing `person` and `LV` from the background, we simply assign all 198 non `person` or `car` outputs as background.

With this simple class mapping approach and assuming that falsely picking one of the positive classes is in fact uniformly random, we expect to eliminate 99% of all the proposed background regions. However, when processing around 100 proposals per frame, the expected false positive rate is once per frame. Next we propose a simple background model that reuses the ConvNet computation to provide a background likelihood estimate for reducing this false positive rate.

3.3 Background Modelling

While on a mine-site the landscape is constantly changing from a geometric perspective, the bleak visual appearance of the background is generally constant. For this, we model the background regions as belonging to one of an arbitrary set of categories, such as the semantic categories of rock, sky, tree etc. If a sample differs significantly from any of these background classes then we can assume it is an object of interest.

Rather than using supervised techniques that require a set of manually annotated images, we instead partition the background data without explicit semantic labels.

Fig. 3 An illustration showing six of the most common types of background region proposals. The rows represent different clusters while the columns show a random background region which is a member of the associated cluster. Each cluster gathers samples with similar visual appearance such as centred on a tree (top row) or centred on sky with an adjacent vertical structure (second row).

To do this, we exploit the assumption that intra-category samples generally appear visually similar to each other, yet may be distinctively different to other background categories. Put another way, the background regions form natural clusters enabling us to employ unsupervised techniques to model their visual appearance. See Fig. 3 for an illustration of the natural background clusters found by applying this method to a mining dataset.

To describe the visual appearance of each region, the intermediate layers of the ConvNet provide a free and compact representation suitable for this task. Addition-

ally, these features have been shown to be robust against lighting and viewpoint changes without any re-training [25]. We refer the interested reader to [13] for an illustration of the ConvNet's inner workings. In general, the first layer of a ConvNet extracts simple colour and texture features in the first layer, and through subsequent layers, these features eventually transition to the learnt specific task [29] such as classifying the 200 ImageNet classes. Along the way irrelevant visual information for the original task (e.g. features describing sky) are lost once it reaches the final layer. With this intuition we reuse the transformed data from one of the ConvNet's intermediate layers as an input to our background model.

To learn this cluster based model, a reservoir of negative samples is required. Gathering background data is a relatively simple task since only inspection for the presence of target objects is necessary. Specifically any image sequence not containing any of the target objects can be used to build an extremely large reservoir by extracting proposals from each frame. Furthermore, we only focus on difficult regions by perform hard-negative-mining [8] of background samples by running the ConvNet detection pipeline over these sequences. By lowering the confidence threshold, near false positive background regions can also be added to build a sufficiently large reservoir.

After extracting an intermediate layer of the ConvNet for each background patch, we then cluster these samples using k-means clustering. At test time, each `person` or `LV` predicted patch is verified by measuring the Euclidean distance between its intermediate feature and each cluster centre. If the nearest background cluster is close in this feature space, i.e. is visually similar, then we suppress the detection and regard it as background.

In building this background model the following choices are to be made: Which layer from the ConvNet? How many clusters? At what distance should a sample be considered background? In the following section we address these design choices through experimental validation.

4 Experiments

4.1 Mining Dataset

The dataset we use for evaluating this work was collected from a light vehicle mounted camera operating in an active mine-site, see Fig. 4. While the motivation is to put vision based sensing on a heavy vehicle, a light vehicle is more practical for gathering a diverse set of visual sequences. The dataset contains both static and dynamic instances of a `person`, `LV` or `HV`.

Continuous video was gathered with and without the camera in motion and on various haul roads and a few light vehicle only zones to capture variation in the environment. This video data was captured at 10 fps and partitioned into various sequences. In this work we use 5 sequences where no people or vehicles are visible

Fig. 4 The experimental dataset gathering vehicle with cameras mounted to the bullbar. Note: all images used in this paper were captured from the camera on the left hand side of the vehicle.

to build our background model. Collectively these background sequences make up 8952 frames in total (approximately 14km).

To evaluate the performance we use another 5 sequences with several instances of `person`, `LV` or `HV`, that we personally annotated using the tool developed by Vondrick et al [28]. These annotated sequences contain 9405 frames in total (approximately 10km). In addition to these sequences we made a small validation set of 200 using other images collected on a mine site from various sources including a few captured at night. This set was used to generate Fig. 2.

4.2 Background Model Validation

Here we describe the experiments performed to design our background modelling system explained in the previous section. From the 5 background sequences, we applied the region proposal and ConvNet detection framework to find challenging region proposals from every tenth frame. While some of the false objects may be observed in multiple frames, the time difference is sufficient to capture a variety of view points for these distracting objects. We lowered the detection threshold to collect region proposals if the ConvNet predicted either a `person` or `car` in the top 5 out of 200 class responses. With this configuration we collect around 8000

hard negatives for our background reservoir. We held out 90 of the most interesting background regions and added them to the validation set.

To address the design decisions for this model, we perform an empirical study using the reservoir containing only negatives and the validation set with both negative and positives. We jointly test different combinations of ConvNet layer features and number of clusters by evaluating their performance on the validation set. For the distance threshold we set this to the distance corresponding to a 95% recall on the positive set. With the recall fixed, the overall performance of the background model is measured by the precision at which it can identify a true negative.

Fig. 5 shows the relative performance of sweeping the number of clusters for different ConvNet layers. While `fc6` layer with 2048 clusters achieved the highest precision of 90% we instead opted to use only 128 clusters with a precision of 89% which is significantly faster to compute. A detailed view of the distances between the validation samples and the cluster centres can be seen in Fig. 6.

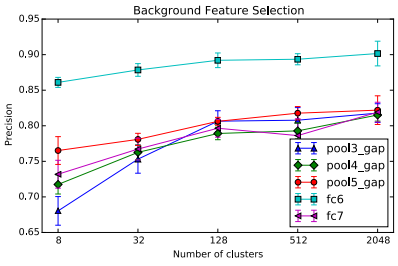


Fig. 5 Cross-validation precision at 95% recall for different ConvNet layers and the number of clusters used to represent the background. Each point shows average of 5 trials.

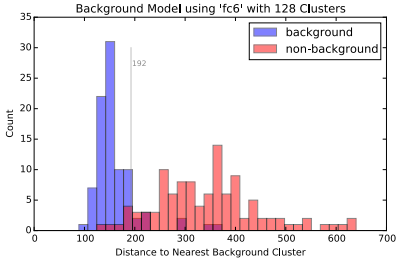


Fig. 6 Detailed view of the distribution of the validation images distance to their nearest background cluster centre. The grey line marks the 95% recall distance threshold.

Implementation Detail

The first 5 ConvNet layers produce dense tensor representations which gradually reduce in size. Then there are two `fully connected` layers `fc6` and `fc7` before the final prediction layer. Again we refer the interested reader to [13] for details of the ConvNet structure. Due to the density of data and the computational complexity of computing distances in such high dimensional feature spaces we only evaluate the ConvNet layers 3-7 and compress convolutional layers 3-5 by pooling all filter responses across the feature map for each tensor in [15] this is referred to global average pooling. In Fig. 5 these are marked as pool{3-5}_gap.

Fig. 7 Validation samples where the background model failed. Images are shown in their warped form, representing the ConvNet input. The four right false negatives were collected at night.

The false negatives and some of the false positives are also shown in Fig. 7. The false negatives are mostly night images which can be put down to the fact that similar images are rare if not non-existent in the ImageNet samples used to train the ConvNet. For the false positives, these are mostly signs which make up a minority of the scene. From these samples we can describe our background model as a form of novelty detection where interesting parts of the scene such as signs are distinguished from the general background. This finding along with the unsupervised clustering shown in Fig. 3 are a testament to the ConvNet's expressive capabilities in representing visual similarity.

4.3 Detection Evaluation

We now evaluate the system on the set of 5 sequences with `person` or `LV` where the task is to locate objects of interest. In this evaluation we consider a true detection if at least 50% of the detection region is covered by a single ground truth object. This differs from the intersection-over-union (IOU) definition of overlap, as we accept detecting a `person`'s head and shoulders without their whole body while IOU would count this as both a miss detection and a false positive. It should be also noted that any detection or miss detection of a `person` or `LV` labelled as partially occluded in the ground truth is ignored in this evaluation. While the system is not designed to detect `HV` we consider any detections which overlap with `HV` objects as neither true or false and are excluded from the evaluation. Additionally, if multiple detections overlap a single ground truth instance, we count this as a single true positive and neither of the overlapping detections are false. An example would be if a person's head is covered by a single detection and their body another.

Table 1 shows the performance of the system before and after applying background suppression. From these results we can see that while there is a slight drop in recall our method for suppressing background regions reduces the false positive rate by an order of magnitude.

Table 1 System Comparison before and after Background Suppression (BGS) on Mining Sequences

Sequence (frames)	F1 Score[a] baseline	(Precision, Recall) with BGS[3]	Mostly Hit[b] -	BGS	Mostly Missed[b] -	BGS	False Positives -	BGS
1 (1462)	0.38 (0.57,0.29)	**0.40** (0.77,0.27)	2	2	16	16	242	**87**
2 (2950)	**0.94** (0.96,0.91)	0.93 (0.97,0.89)	3	3	6	6	73	**47**
3 (599)	0.02 (0.01,0.09)	**0.06** (1.00,0.03)	0	0	2	2	349	**0**
4 (2826)	0.64 (0.56,0.74)	**0.80** (0.95,0.69)	**2**	1	**4**	5	186	**9**
5 (1568)	**0.68** (0.78,0.61)	0.43 (0.92,0.28)	**4**	1	**3**	6	177	**24**
Total			11	7	31	35	1027	**167**

[a] F1, Precision and Recall is computed treating each frame as independent.
[b] Mostly indicates where a single object instance was detected or missed 50% of the time.
[c] The proposed background suppression (BGS) is applied to the baseline `EdgeBox` and ConvNet detector.

5 Conclusions and Future Work

In this paper we presented a vision only system that takes advantage of recent developments in computer vision and machine learning to detect both personnel and light vehicles. We circumvented the problem of ConvNet over-fitting on small datasets by reusing a pretrained model directly and mapping its output to mining classes. We further presented a method for exploiting the abundance of background only images to learn a background cluster model leading to a significant reduction in false positives. This sensing approach was evaluated in an active open-pit mine site environment. The experiments show that the in-pit environment is suitable for object proposals along with background modelling techniques such as the one presented here.

While this work is only concerned with single camera based sensor data we see many opportunities to combine techniques incorporating stereo [2] or range-based sensors [20] for improved robustness. As an initial investigation of vision as a possible sensor on a mine we see many opportunities to further improve on the results. As

more labelled mining image data becomes available we expect to be able to design and fine-tune a ConvNet that performs better in this domain than the existing network. We also plan to extend this work to fuse information from multiple frames by combining the ConvNet appearance model with recent motion segmentation techniques [1].

Acknowledgements This research was funded by the Australian Coal Association Research Program (ACARP). The authors would also like to acknowledge AngloAmerican for allowing data collection at the Dawson operation. Acknowledgement also goes to the high performance computing group at Queensland University of Technology for both support and use of their services when conducting the experiments in this paper.

References

1. Alex Bewley, Vitor Guizilini, Fabio Ramos, and Ben Upcroft. Online Self-Supervised Multi-Instance Segmentation of Dynamic Objects. In *International Conference on Robotics and Automation*, Hong Kong, China, 2014. IEEE.
2. Alex Bewley and Ben Upcroft. Advantages of Exploiting Projection Structure for Segmenting Dense 3D Point Clouds. In *Australian Conference on Robotics and Automation*, 2013.
3. Huawu Deng and David a. Clausi. Unsupervised image segmentation using a simple MRF model with a new implementation scheme. *Pattern Recognition*, 37(12):2323–2335, December 2004.
4. Jai Deng, Wei Dong, Richard Socher, Li-Jai Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009.
5. Jeff Donahue, Judy Hoffman, Erik Rodner, Kate Saenko, and Trevor Darrell. Semi-supervised Domain Adaptation with Instance Constraints. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 668–675, June 2013.
6. Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning Hierarchical Features for Scene Labeling. *Pattern Analysis and Machine Intelligence*, 35(8):1915–1929, 2013.
7. Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, April 2007.
8. Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–45, September 2010.
9. Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, 59(2):167–181, September 2004.
10. Ross B. Girshick, Jeff Donahue, Trevor Darrell, and J Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2014.
11. Jan Hosang, Rodrigo Benenson, and B Schiele. How good are detection proposals, really? In *British Machine Vision Conference (BMVC)*, 2014.
12. Yangqing Jia. {Caffe}: An Open Source Convolutional Architecture for Fast Feature Embedding. \url{http://caffe.berkeleyvision.org/}, 2013.
13. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages (Vol. 1, No. 2, p. 4), 2012.

14. Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1:541–551, 1989.
15. Min Lin, Qiang Chen, and Shuicheng Yan. Network In Network. *arXiv preprint*, 2013.
16. Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision (ECCV)*, volume 8693, pages 740–755. Springer International Publishing, 2014.
17. Joshua A. Marshall and Timothy D. Barfoot. Design and field testing of an autonomous underground tramming system. *Springer Tracts in Advanced Robotics*, 42:521–530, 2008.
18. Rafael Mosberger and Henrik Andreasson. Estimating the 3d position of humans wearing a reflective vest using a single camera system. In *International Conference on Field and Service Robotics (FSR)*, 2012.
19. Rafael Mosberger, Henrik Andreasson, and Achim J. Lilienthal. Multi-human Tracking using High-visibility Clothing for Industrial Safety. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 638–644, 2013.
20. Tyson Phillips, Martin Hahn, and Ross McAree. An evaluation of ranging sensor performance for mining automation applications. In *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics: Mechatronics for Human Wellbeing, AIM 2013*, pages 1284–1289, 2013.
21. Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. CNN Features off-the-shelf: an Astounding Baseline for Recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2014.
22. Jonathan M. Roberts and Peter I. Corke. Obstacle detection for a mining vehicle using a 2D laser. In *Proceedings of the Australian Conference on Robotics and Automation*, pages 185–190, 2000.
23. Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus, and Yann LeCun. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *International Conference on Learning Representations (ICLR 2014)*, December 2014.
24. Pierre Sermanet, Koray Kavukcuoglu, Soumith Chintala, and Yann LeCun. Pedestrian Detection with Unsupervised Multi-stage Feature Learning. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3626–3633. IEEE, June 2013.
25. Niko Sunderhauf, Feras Dayoub, Shirazi Sareh, Upcroft Ben, and Milford Michael. On the Performance of ConvNet Features for Place Recognition. In *arXiv*, 2015.
26. A Torralba, R Fergus, and W Freeman. 80 Millions Tiny Images: a Large Dataset for Non-Parametric Object and Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:1958–1970, 2008.
27. J. R. R. Uijlings, K. E. a. Sande, T. Gevers, and a. W. M. Smeulders. Selective Search for Object Recognition. *International Journal of Computer Vision*, 104(2):154–171, April 2013.
28. Carl Vondrick, Donald Patterson, and Deva Ramanan. Efficiently Scaling up Crowdsourced Video Annotation. *International Journal of Computer Vision*, 101(1):184–204, 2012.
29. Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks ? In *Advances in Neural Information Processing Systems (NIPS)*, 2014.
30. CL Zitnick and P Dollár. Edge Boxes: Locating Object Proposals from Edges. In *European Conference on Computer Vision (ECCV)*, 2014.

Further Analysis

Given the lack of domain specific training data in the context for learning an object detector for mine site operations, this chapter proposed a background model which can be trained with minimal supervision. Comparisons with a DCN pretrained on ImageNet data (previously denoted as DCN Remapped) and fine-tuned with limited amounts of mining images (previously denoted as DCN Retrained), showed that fusing predictions from the retrained method with the proposed background model greatly improved the detection performance. If more mining related training data was available, we expect the DCN detector to further increase in performance, thus mitigating the need for a weakly supervised background model. The remainder of this chapter analyses the value of the proposed weakly supervised background model compared to providing significantly more labelled data for training the DCN detector.

To extend the amount of labelled training data, the positives from all six test sequences from Table 1 in [Bewley and Upcroft, 2016] are used to train six models in a leave-one-sequence-out form of cross validation. The evaluation criteria matches what was described earlier in [Bewley and Upcroft, 2016]. For a fair comparison the same features from each box proposal is provided to training the network as seen by the background model. In other words, the earlier convolutional layers are fixed while the fully connected layers are optimised for the additional training data. Preliminary experiments without fixing these layers lead to over-fitting as the ImageNet weights were catastrophically forgotten.

Table 4.1: F1-score Detection Performance on Mining Sequences

Sequence	DCN Retrained Minimal	DCN Retrained Full	Fused Minimal
Test 1	0.17	**0.75**	0.27
Test 2	0.80	0.75	**0.90**
Test 3	0.09	**0.89**	0.69
Test 4	0.42	0.64	**0.68**
Test 5	0.72	0.81	**0.83**
Test 6	0.61	**0.68**	0.59
Overall	0.541	**0.727**	0.692

Bold indicates the best performance across models.

Table 4.1 reports the aggregate F1 score as the mean performance metric across all six model

training-validation splits. The DCN Retrained (Minimal) and Fused models from [Bewley and Upcroft, 2016] are shown here for comparison. The DCN Retained Full column shows the performance of training the DCN using both all the positives and hard negatives from the other five mining sequences. This represents over 40x more positive training samples per split compared to the DCN Retrained Minimal (8086 vs. 200). However, it is important to note that there is a high degree of correlation across positive samples due to the natural temporal consistency inherent in sequential data.

The results show an overall 19% gain for the model with significantly more training data which also out-performs the background fused model on half of the sequences. This supports the notion that DCN based detectors can benefit from significant amounts of domain specific training data. However, the background fused model still achieves the better F1 score on sequences 2, 4, and 5 which are mostly in the mining pit indicating that the background model is still more reliable in the mine site environment.

From this further analysis, it is clear that gathering significantly more training data in the target specific domain dramatically improves the performance of DCN detectors. Although, in specific areas of the mine, a background model with minimal supervision is preferable, suggesting that future work should investigate novel hybrid methods conditioned on the place of deployment.

Chapter 5

Improved Tracking through Self-Supervised Learning

In this chapter, the idea of learning an appearance model from the environment in which the system has been deployed is extended to object tracking. While a form of tracking was achieved in Chapter 3, only objects which move in the scene were tracked as opposed to general (potentially stationary) objects of a particular category. Furthermore, the appearance information learnt for the tracked objects was tied to a specific instance requiring the appearance of newly discovered instances to be learnt from scratch. This limitation stems from the classifier design where labels represented the object identities and the simple point features lacked the expressive capability to generalise.

Earlier, Chapter 4 demonstrated that features extracted from an intermediate layer of a DCN generalise for tasks beyond object classification. Specifically, the same features used for detection were also used for finding background clusters with similar visual appearance. Given that these features are useful for both detection and measuring visual similarity, it is expected that they are also useful for tracking.

This chapter employs the tracking-by-detection paradigm to decouple the detection and tracking tasks. In particular, any general object detector can be used to locate object instances in the current frame. These detections are then provided to the tracking component responsible for assigning them to tracked objects in the previous frame. A key challenge of visual tracking is estimating the assignment cost which should incorporate appearance information unique to the detected instances. Thus, addressing the final research question: *How to adapt the assignment*

cost using data collected at deployment?

The core contribution of this chapter is a novel visual affinity model which approximates the probability that a pair of detections represent the same object. This model uses a logistic regression classifier which takes a pair of DCN features representing two detections and outputs the likelihood of both detections belonging to the same object instance. In contrast to prior classification based approaches to tracking [Bewley et al., 2014, Kalal et al., 2012], here a single classifier learns a pairwise assignment cost as opposed to appearance models tied to specific instances. Additionally, by focusing on the assignment cost, optimal association across video frames can be achieved using traditional methods such as the Hungarian algorithm [Kuhn, 1955]. Finally, this framework naturally extends to handling previously unobserved instances as the visual affinity model applies to all pairs of detections.

The remainder of this chapter consists of the paper entitled: "ALExTRAC: Affinity Learning by Exploring Temporal Reinforcement within Association Chains" which was presented in 2016 at the International Conference of Robotics and Automation. This paper presents the pairwise affinity model and also includes a method for self-supervision which improves the affinity model with training pairs gathered during deployment. Furthermore, the temporal history of tracked objects are explored for finding challenging matching pairs for training the affinity model to be robust to high variation in visual appearance. The results show that this affinity model accurately estimates an assignment cost for use in a tracking framework as demonstrated on a standard tracking benchmark.

RESEARCH STUDENTS CENTRE
Examination Enquiries: 07 3138 1839
Email: research.examination@qut.edu.au

Statement of Contribution of Co-Authors for Thesis by Published Paper

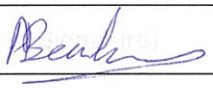
The following is the format for the required declaration provided at the start of any thesis chapter which includes a co-authored publication.

The authors listed below have certified* that:

1. they meet the criteria for authorship in that they have participated in the conception, execution, or interpretation, of at least that part of the publication in their field of expertise;
2. they take public responsibility for their part of the publication, except for the responsible author who accepts overall responsibility for the publication;
3. there are no other authors of the publication according to these criteria;
4. potential conflicts of interest have been disclosed to (a) granting bodies, (b) the editor or publisher of journals or other publications, and (c) the head of the responsible academic unit, and
5. they agree to the use of the publication in the student's thesis and its publication on the QUT ePrints database consistent with any limitations set by publisher requirements.

In the case of this chapter:

Publication title and date of publication or status:
"ALExTRAC: Affinity Learning by Exploring Temporal Reinforcement within Association Chains" (to appear May, 2016)

Contributor	Statement of contribution
Alex Bewley  8th April 2016	Wrote the manuscript, designed and conducted experiments and performed data analysis
Lionel Ott	Provided input into experimental design, scope of paper and proofreading/editing
Fabio Ramos	Provided input into experimental design, scope of paper and proofreading/editing
Ben Upcroft	Provided input into experimental design, scope of paper and proofreading/editing

Principal Supervisor Confirmation

I have sighted email or other correspondence from all Co-authors confirming their certifying authorship.

Ben Upcroft		15/4/16
Name	Signature	Date

ALExTRAC: Affinity Learning by Exploring Temporal Reinforcement within Association Chains

Alex Bewley[1], Lionel Ott[2], Fabio Ramos[2] and Ben Upcroft[1]

***Abstract*— This paper presents a self-supervised approach for learning to associate object detections in a video sequence as often required in tracking-by-detection systems. In this paper we focus on learning an affinity model to estimate the data association cost, which can adapt to different situations by exploiting the sequential nature of video data. We also propose a framework for gathering additional training samples at test time with high variation in visual appearance, naturally inherent in large temporal windows. Reinforcing the model with these difficult samples greatly improves the affinity model compared to standard similarity measures such as cosine similarity. We experimentally demonstrate the efficacy of the resulting affinity model on several multiple object tracking (MOT) benchmark sequences. Using the affinity model alone places this approach in the top 25 state-of-the-art trackers with an average rank of 21.3 across 11 test sequences and an overall multiple object tracking accuracy (MOTA) of 17%. This is considerable as our simple approach only uses the appearance of the detected regions in contrast to other techniques with global optimisation or complex motion models.**

I. Introduction

This paper presents the design and implementation of a self-supervised framework to solve the data association component for tracking-by-detection. In tracking-by-detection, a low level object detector typically operates independently of the high level data association. This independence offers several benefits including: robustness to drift as it does not rely on state information, accommodates a changing number of objects in the scene, and implicit recovery from detection failure. To perform the data association most approaches rely on position information and incorporate motion models [1], [2] where hand crafted appearance features, such as colour histograms are only used to resolve ambiguous situations [3]–[5].

While the tracking-by-detection problem has been investigated with various formulations, little attention has been invested into improving the appearance similarity measure, commonly used for quantifying the data association cost. Our approach addresses this gap by actively modelling the pairwise similarity between detections with a probabilistic classifier, referred to as the *affinity model*. The output of this model can be inserted into any tracking-by-detection framework to improve data association in any arbitrary environment.

We demonstrate that the temporal structure of a video sequence can be explored to gather training samples to reinforce the proposed affinity model. For example, Fig. 1 illustrates an *association chain* where the frame-to-frame affinities are appropriately high and can be used to identify a difficult matching pair with a large temporal separation. To gather reliable matching and non-matching samples, we rely on two basic constraints for exploiting the structure of video data. Firstly, an object should have higher visual affinity to itself than other objects over short temporal durations, and secondly, co-existing and non overlapping detections cannot represent the same object, known as mutual exclusion. These constraints provide a valuable utility in governing the self-supervision, particularly in preventing drift, to facilitate life long learning.

Fig. 1. An illustration of an association chain showing that while the frame-to-frame affinities are strong, the direct affinity over longer temporal windows are affected by viewpoint change and background clutter. Warmer colours denote stronger affinity.

Our design is simple and practical as it can learn at test time without labels, automatically adapting to new visual appearances. Additionally, fewer parameters are required, in contrast to conventional motion model approaches. Finally, this purely appearance based affinity model is complimentary to other approaches relying on object dynamics such as [6] or approaches that incorporate an appearance based cost in a global optimisation [7]. The key contributions of this paper are as follows:

- modelling visual affinity with a probabilistic classifier,
- gathering matching examples with high variance by exploring association chains,
- use of co-existing detections as a source of non-matching examples,
- use of a purely appearance based data association cost,
- affinity model reinforcement without explicit labelling.

[1]A. Bewley and B. Upcroft are with the Queensland University of Technology (QUT) alex.bewley@hdr.qut.edu.au

[2]L. Ott and F. Ramos are with the School of Information Technologies, The University of Sydney, Australia

This paper is organised as follows: In the next section, we position the proposed approach among existing works. An overview of the proposed method is given in section III. In section IV, we demonstrate the performance of the proposed method before a conclusion and an outlook to future work is given in section V.

II. Related Work

Here, we review techniques incorporating visual appearance for multiple object tracking (MOT) with a strong focus on techniques that include a learning component or rely on affinity measures. In a general sense, many techniques associate detections between frames by utilising features extracted from each frame to measure and compare the strength of all potential matches [1], [2], [8], [9]. Selecting among the candidate matches can either be performed in a greedy (winner-take-all) approach [8], [9] or by formulating a bipartite graph [1], [2] which can be solved optimally [10]. These approaches all require a measure of affinity between two candidate detections but have mostly only considered fixed affinity measures such as the intersection kernel [7], [11] or Kullback-Leibler distance [5] between two colour histograms or normalised cross correlation between patches [2].

The focus of this work is to learn an affinity measure which is optimised for the task of visual tracking, and which can complement the above matching and graph optimisation approaches. This approach is also related to the structural SVM approach, originally used for supervised clustering [12]. Kim et al. [13] use implicit spatial-temporal structure of video based tracking by employing a structural SVM to predict the optimal matching as formulated in a bipartite graph structure.

Another approach is to cast data association into a ranking problem where similar objects should be ranked higher over dissimilar objects [14]. Soleraet et al. [5] propose a method based on the Latent Structual SVM to partition a scene and learn which input features are most important in a divide and conquer fashion. While they use colour histograms in their experiments, their learning approach can adapt to any set of provided features. However, the divide step of their approach is based on spatial distance limiting their state-of-the-art performance to only static cameras.

More recently, Choi et al. [15] learnt an affinity model to measure if two detections correspond where the input is a set of low level feature trajectories, requiring that optical flow must first be computed. Probably the most similar work to ours is that of Bae and Yoon [16] which sequentially grows tracklets computing confidence from multiple sources. They learn to associate detections to tracklets using incremental discriminative learning analysis on features composed of both appearance and motion cues. Our method differs from theirs as we do not use motion information and our affinity model estimates a universal pairwise affinity measure while theirs treats each tracklet as a separate class.

In collecting training samples from test data we make use of the mutual exclusion constraint as a hard assumption in self-supervision. The *exclusion principle* was first introduced by [17] for the purpose of disambiguating two intersecting tracks. Similarly, [18] enforced the exclusion property at both the detection and trajectory levels using penalty terms within a conditional random field framework, however their work only considered the position of the detections. Position based exclusion methods are generally combined with other cues including motion and appearance [7] to become competitive. This work extends the notion of exclusion from these earlier works to reinforce the appearance affinity model to coherently distinguish multiple detections when collecting samples for training.

III. Proposed Method

A. Overview

The input to our system is a set of detections for each frame in a sequence. Although our approach is agnostic to the type of detector, we require that a description of the visual appearance is supplied for each detection. Convolutional network features are used as descriptors since they have been found to successfully capture visual similarity in related fields such as image retrieval [19] and clustering [20]. Additionally, when a convolutional network is used for the prior detection step (e.g. [21]), computing such descriptors comes for free as the computation is performed in the detection step. While any sufficient descriptor could be used to describe appearance, a detailed comparison with different descriptor types is beyond the scope of this paper.

Following the lines of [1], [2], [13], we formulate the frame-to-frame matching process as a bipartite graph. This strictly enforces a one-to-one matching among the detections between adjacent frames. The Hungarian algorithm [10] (also known as the Kuhn-Munkres method) is then used to find the optimal assignment which maximises the total affinity across all frame-to-frame matches. The overall performance of this assignment boils down to the quality of the affinity measure employed.

We concentrate on modelling the visual affinity and propose to formulate it as a binary classification problem. This affinity model takes a pair of visual descriptors to produce an output indicating whether the two samples match. The affinity model's output is then used as the matching score to drive the Hungarian based matching to associate detections to tracklets. In the remainder of this section we go into more detail of our approach and describe how we train this model in a self-supervised manner.

B. Pairwise Features

To discriminate a pair of candidate patches as either matching or non-matching, a bidirectional feature is required. For pragmatic reasons, we simply compute the absolute difference between each element of the two patch descriptors as follows:

$$\mathbf{d}_{i,j} = |\mathbf{d}_i - \mathbf{d}_j| \quad (1),$$

where $\mathbf{d}_{i,j}$ is a vector representing the pairwise descriptor computed from the descriptors $\mathbf{d}_i$ and $\mathbf{d}_j$ extracted from

detection patches i and j respectively. A comparison of various descriptor extraction techniques and methods for constructing pairwise feature vectors is beyond the scope of this paper, so unless stated otherwise this is the pairwise feature descriptor used throughout this work.

C. Affinity Metric

The affinity is modelled with a linear logistic regression classifier [22] and trained on a set of candidate patch pairs. Given a descriptor describing the appearance of each patch, the goal is to learn which features within the descriptors are most informative for discriminating between a pair of matching and non-matching patches. The affinity model approximates the probability that two patches match denoted by:

$$p_a(m|\mathbf{d}_{i,j}) = \frac{1}{1+\exp^{-(\theta^T \mathbf{d}_{i,j}+b)}} \qquad (2),$$

where θ denotes the weights corresponding to each feature and b is a bias term. Both θ and b are efficiently optimised through Stochastic Gradient Descent (SGD) on a set of matching and non-matching pairs.

D. Collecting Initial Pairs

As with any classification problem, a set of training samples is required in order to train the affinity model. It is desirable to have a means of collecting training samples in any new environment for which this system is deployed for life-long learning. In the contexts of visual tracking, we can rely on the mutual exclusion constraint to gather non-matching pairs from each frame. Similarly, an initial set of positive pairs can be collected using optimal bipartite matching across frames and conservatively keeping only high confident matches. To achieve this however, an initial measure of similarity is required.

The cosine similarity metric is used as an initial measure of affinity since it possesses several desirable properties, including:

- When features are constrained to positive spaces, the cosine similarity is bounded to interval $[\mathbf{0}, \mathbf{1}]$. This reflects the probabilistic output range for our desired affinity model.
- Computation of cosine distance does not require expensive modelling of the data distribution, i.e. Mahalanobis distance.
- Considered to be more stable in higher dimensions than other measures based on Minkowski distance.

Fig. 2 shows that the cosine similarity metric is a reasonable measure as its maximal match is generally the same object. However, as the cosine similarity rapidly decays with increasing temporal windows we restrict this initial set of positive matching to adjacent frames. This initial set of samples is used to seed the affinity model with a single round of SGD. Next, we describe how these frame-to-frame associations are used to learn an affinity measure for estimating the matching likelihood over multiple frames.

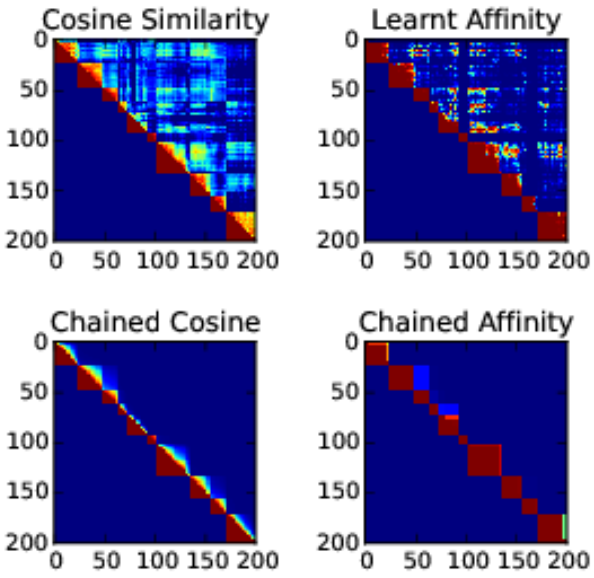


Fig. 2. A comparison between cosine similarity and the learnt affinity model after three iterations on a set of 200 tracklet patches of pedestrians taken from the KITTI benchmark [23]. The lower triangle represents the ground truth matching where the detections have been grouped and reordered such that each red block represents an instance of an object. The upper triangles show the affinity estimate (top row) and chained affinity (bottom row). This comparison demonstrates that having a near binary estimate is important in long association chains. Best viewed in colour.

E. Association Chains

Using the initial affinity model as a similarity measure for frame-to-frame associations, we seek to extend its capability to match more challenging examples with higher variation as observed over larger temporal windows. To achieve this, we explore the frame-to-frame associations to incrementally grow a chain and select pairs along this chain to reinforce the affinity model. This process is provided in Algorithm 1 with key points described in this section.

Using the probability chain rule, the probability that two non-temporally-adjacent detections represent the same object can be modelled as:

$$p_c(m|\mathbf{d}_{i,j}) = \prod_{t=i}^{j-1} p_a(m|\mathbf{d}_{t,t+1}). \qquad (3)$$

This *chained affinity* essentially captures the joint probability that all associated detections in the tracklet segment $[i, j]$ belong to the same object. As the temporal duration between the two detections i and j increases, there is a monotonic decrease in the chained affinity, accurately capturing the growing uncertainty. Put another way, the chained affinity has an upper bound which is less than the minimum link affinity along the chain. This is important in identifying the temporal boundary of an objects existence as shown in Fig. 2.

By exploring the tracklet history, errors in the direct affinity measure can be compared to the chained affinity and used to collect additional positive training samples over

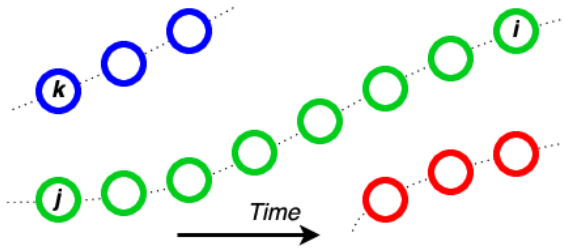


Fig. 3. Three tracklets where colour denotes object identity. By exploring the tracklet, detection i in the current frame can be linked to detection j several frames before by using the chained affinity to create a new positive sample pair (i, j). Similarly, since j and k share the same frame we can also gather a non-matching pair (i, k).

larger temporal windows. To prevent learning from potentially incorrect matches, only confident positive pairs are collected as measured by comparing the chained affinity to a threshold δ (line 6 of Algorithm 1). These samples are used to reinforce the affinity model through incremental learning – causing the frame-to-frame direct affinities to approach 1 – resulting in a reduction in the decay of the chained affinity – and ultimately increasing the chain for gathering additional samples.

Consistently adding new examples of positive matches from the temporal history can lead to an imbalance of the negative to positive ratio. The common remedy of increasing the weighting ratio between the negative and positive classes can be limiting as it is often difficult to estimate the ratio ahead of time. To alleviate this dissension, we build off the association chain affinity introduced in the last section to also find additional non-matching examples over longer temporal windows. The association chains are also used to collect additional negative samples to supplement the non-matching examples of the current frame collected by using the mutual exclusion constraint.

Given that we have formed a tracklet $[i, j]$ with high confidence, other non-overlapping detections in the same distant frame can also be used as negative match examples. The intuition of this is aligned with the notion of self-similarity, i.e. as the visual appearance of a single object changes over time so does the relative appearance of that object with respect to other objects. For example, Fig. 3 shows a matched tracklet chain $[i, j]$ where the older detection j shares the same frame as detection k. Given that i matched j with a confidence of $p_c(m|d_{i,j})$, and that $j \neq k$ due to mutual exclusion, we expect that the direct affinity between i and k should be low (see lines 14 and 15). This places an upper limit on the affinity between i and another non-overlapping detection k from the same frame as j. Errors from the affinity model which violate this constraint are used as non-matching samples during the incremental updating of the model.

F. Classifier Balance

Self-supervised frameworks are prone to drift, particularly where the model learnt is the same model used to gather examples. The intuition behind the mutual exclusion and

Algorithm 1 Exploring tracklets for difficult samples

Input: $\mathbf{d}_i$ ▷ descriptor of current detection i
Input: $\mathbf{t}_{i-1}$ ▷ tracklet list matched to detection i
Input: $f_\theta()$ ▷ Eqn. 2
Output: $X, \mathbf{m}, \mathbf{w}$ ▷ Training inputs, outputs and weights

1: **function** $(\mathbf{d}_i, \mathbf{t}_{i-1}, f_\alpha)$
2: $p_c = f_\alpha(\mathbf{d}_{i,t_i})$
3: **for** $j = reverse\_iterate(\mathbf{t}_{i-1})$ **do**
4: $p_a = f_\alpha(\mathbf{d}_{i,j})$
5: $p_c = p_c * f_\theta(\mathbf{d}_{j-1,j})$ ▷ Eqn. 3
6: **if** $p_c(i, j) < \delta$ **then**
7: **break**
8: **if** $p_c > p_a$ **then**
9: $X \leftarrow \mathbf{d}_{i,j}$
10: $\mathbf{m} \leftarrow 1$
11: $\mathbf{w} \leftarrow (p_c - p_a)$
12: **for each** $k \in frame(j)$ **do**
13: **if** $overlap(j, k) \equiv False$ **then**
14: $p_n = 1 - f_\theta(\mathbf{d}_{i,k})$
15: **if** $p_c > p_n$ **then**
16: $X \leftarrow \mathbf{d}_{i,k}$
17: $\mathbf{m} \leftarrow 0$
18: $\mathbf{w} \leftarrow (p_c - p_n)$
return $X, \mathbf{m}, \mathbf{w}$

bipartite matching constraints offer some robustness to drift by selectively rejecting potential pairs which violate these constraints. However, the bipartite property assumes a one-to-one matching which does not factor in situations where objects enter or leave the scene and/or the detector misses or introduces false positives. These situations are actively handled by further rejecting matches if their affinity is less than 50%. If the affinity model is incorrect with this prediction, i.e. the pair are truly matched then this small failure can lead to severe consequences. This situation occurs if the model becomes negatively biased, which generates a ripple effect where the tracklet is broken, thus limiting the positive feedback effect of the association chain reinforcement causing the model to become further negatively biased.

Not all errors are created equally!

The key to combating this bias problem is in the placement of importance for the samples collected. Lines 11 and 18 of Algorithm 1 weight the positive and negative samples respectively. Both of these were vetted by first checking if the affinity model created an error by comparison with the chained affinity. This weighting of samples by the amount of error creates a balanced interplay between the positive and negative collection of samples, keeping the model neutral.

Finally, the samples collected over the test sequence are then used to retrain the logistic regression classifier. This improves the affinity model and ultimately enhances the performance of the Hungarian algorithm in assigning the frame-to-frame associations during the tracklet building process.

IV. Experiments

The effect of this learnt affinity method is evaluated in the context of tracking-by-detection on various MOT benchmark sequences described in [24]. This MOT benchmark provides a unified framework for evaluating different multiple object trackers over a variety of sequences filmed from different viewpoints, with different lighting conditions, and different levels of target density.

In this experiment, we used the supplied detections provided with the dataset which where generated using the aggregated channel features (ACF) detector [25]. From each detection bounding box, we use the deep 16-layer convolutional network of [26] as a descriptor extractor. Each detection patch is resized to 224×224 and propagated forward through 13 convolutional layers and one fully-connected layer to produce a 4096 dimensional feature to represent the visual descriptor.

Both the Hungarian matching and logistic regression model implementation were from the scikit-learn toolkit [27]. Our unoptimised, single threaded python implementation of this tracker runs at 3.7 frames per second on a 2.5 GHz intel^{TM} i7.

Table I shows the performance of the proposed method using the widely accepted CLEAR MOT metrics [28] evaluated with a 0.5 intersection over union threshold. The Multiple Object Tracking Accuracy (MOTA) combines all false positives, false negatives, and identity switches into a single number, and Multiple Object Tracking Precision (MOTP) measures the average distance between the ground truth and the tracker output. For both the MOTA and MOTP a higher value represents better performance. The ID switches indicate the total number of times a true object has switched identities according to the tracker output following the strict definition in [14]. Other metrics listed are the False Alarms per Frame (FAF), Mostly Tracked (MT) and Mostly Lost (ML) metrics along with the number of False Positives (FP) and False Negatives (FN). The number of Ground Truth object (GT) is also shown for each sequence.

The results are split into training and test sequences. It is important to note that the ground truth labels were not used within the self-supervised procedure. Decisions on selecting hyper-parameters such as the minimum confidence threshold δ and logistic regression regularisation were selected to increase the MOTA on these training sequences. The minimum confidence threshold $\delta = 0.9$ was found to give best results on the training sequences.

We also include the results of other tracking methods discussed in Section II. Namely a general tracking-by-detection method that uses the Hungarian algorithm TBD [2] and two affinity learning based methods: TC_ODAL [16], LDCT [5]. When considering the key MOTA score our approach performs quite favourably to the related method affinity methods indicating that our self-supervised approach is able to correctly adapt to the data observed. However, it is important to note that several of these sequences involve a dynamic camera. This heavily impacts the overall MOTA score for LDCT [5] which focuses primarily on applications with a static camera. Our method also achieves comparable results to a purely motion based technique [6], indicating that appearance is adequate in object tracking and could be treated as an independent low level tracker if combined with a motion based tracker.

Qualitative results are shown in Fig. 5. The vertical lines indicate an ID switch. With closer inspection (see Fig. 4), many of these are caused by a duplicate detection or false positive in a frame adjacent to a object entering of leaving the scene. Since this technique only matches detections in adjacent frames using visual appearance, ID switches and fragmentation errors are to be expected. Nevertheless, given the large number of tracklets and frames, the affinity model produces accuracy data association in the majority of frames. Furthermore, the white circles in Fig. 5, show detections which were not matched to either the previous or next adjacent frames. In the PETS sequence, it is obvious that these detections correspond to a stationary object which are actually false positives generated by the low level detector. This indicates that temporal inconsistency could be used as a cue for false positive detections which are implicitly handled within this technique as they are not matched to any object tracklet. A clear downside however is the amount of track fragmentation. This is caused by the frames where an object is miss-detected or when two detections cover an object creating a new tracklet. These limitations indicate that this method could be further improved if combined with motion based models.

Fig. 4. Frames 74, 77, and 80 of the KITTI-17 sequence showing a false positive in the top frame getting reassigned to an actual pedestrian emerging from behind a foreground pedestrian. Best viewed in colour.

TABLE I

PERFORMANCE OF THE PROPOSED APPROACH ON MOT BENCHMARK SEQUENCES.

		MOTA↑	MOTP↑	FAF↓	GT	MT↑	ML↓	FP↓	FN↓	ID sw↓	Frag↓
Train Sequences	TUD-Stadtmitte	53.8	65.6	0.90%	10	50.0%	0.0%	146	348	40	33
	TUD-Campus	35.7	70.6	0.34%	8	12.5%	12.5%	51	157	23	25
	PETS09-S2L1	67.7	71.4	0.87%	19	78.9%	0.0%	648	603	193	152
	ETH-Bahnhof	29.3	73.4	1.52%	171	23.4%	37.4%	1544	2121	165	208
	ETH-Sunnyday	30.9	76.2	0.29%	30	10.0%	50.0%	118	1104	61	73
	ETH-Pedcross2	7.9	71.3	0.17%	133	0.0%	85.0%	173	5543	54	78
	ADL-Rundle-6	25.4	71.8	2.09%	24	4.2%	8.3%	974	2590	174	160
	ADL-Rundle-8	10.9	72.4	2.51%	28	10.7%	39.3%	1653	4213	178	187
	KITTI-13	5.8	70.9	0.47%	42	4.8%	38.1%	327	355	36	29
	KITTI-17	48.0	70.8	0.14%	9	0.0%	11.1%	44	292	19	21
	Venice-2	13.3	72.6	3.81%	26	11.5%	19.2%	2421	3591	177	164
	Overall	24.5	72.1	1.43%	500	14.6%	45.6%	8099	20917	1120	1130
Test Sequences	TUD-Crossing	51.2	73.0	0.2%	13	15.4%	15.4%	39	459	40	50
	PETS09-S2L2	27.5	70.6	1.0%	42	0.0%	26.2%	449	6153	385	359
	ETH-Jelmoli	32.9	73.2	0.6%	45	13.3%	28.9%	283	1334	86	98
	ETH-Linthescher	15.4	74.1	0.1%	197	1.5%	76.1%	94	7369	90	103
	ETH-Crossing	19.6	74.7	0.1%	26	7.7%	65.4%	13	783	10	10
	AVG-TownCentre	13.3	70.0	1.0%	226	1.3%	61.1%	442	5637	118	152
	ADL-Rundle-1	5.1	71.3	7.0%	32	15.6%	21.9%	3503	5010	321	306
	ADL-Rundle-3	18.1	71.8	3.9%	44	6.8%	22.7%	2420	5523	385	261
	KITTI-16	13.9	72.1	0.5%	17	0.0%	23.5%	105	1279	80	73
	KITTI-19	13.5	66.5	1.1%	62	6.5%	30.6%	1128	3266	227	326
	Venice-1	12.5	71.9	1.7%	17	0.0%	41.2%	757	3120	117	134
*Proposed method	**Overall**	**17.0**	71.2	**1.6%**	721	3.9%	52.4%	9233	39933	1859	1872
◇TC_ODAL [16]	**Overall**	15.1	70.5	2.2%	721	3.2%	55.8%	12970	38538	637	1716
◇LDCT [5]	**Overall**	4.7	**71.7**	2.4%	721	11.4%	32.5%	14066	32156	12348	2918
◇TBD [2]	**Overall**	15.9	70.9	2.6	721	6.4%	47.9%	14943	34777	1939	1963
†SMOT [6]	**Overall**	18.2	71.2	1.5%	721	2.8%	54.8%	8780	40310	1148	2132

* Purely appearance based
◇ Position and appearance based
† Purely motion based

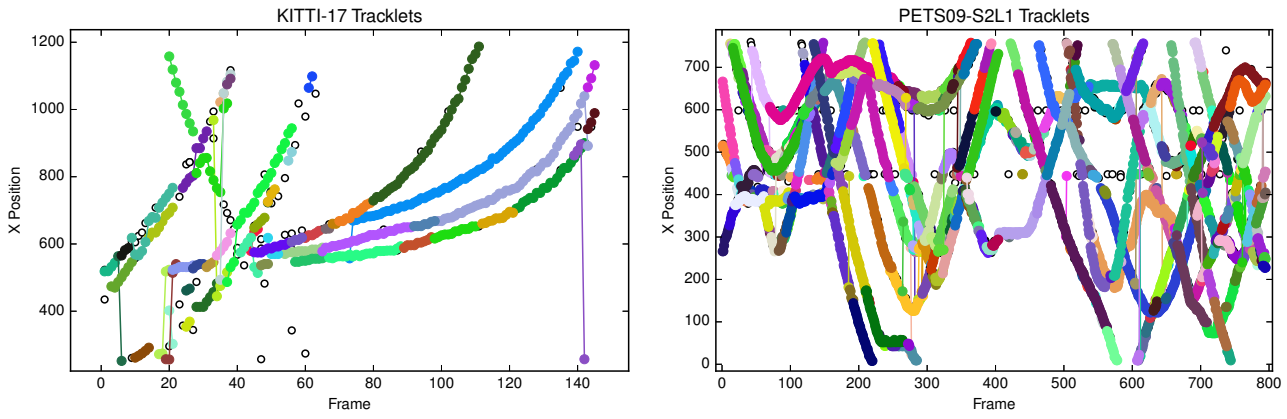


Fig. 5. Spatial-temporal representation of the tracklets discovered in a low target density setting (left) and high density setting (right). It is important to note that frame-to-frame matching was performed using appearance affinity making it robust to complex dynamics as observed in the PETS09-S2L1 sequence. Near vertical lines indicate an ID switch. Best viewed in colour.

V. Conclusion

In this paper we have proposed a method to learn a visual appearance affinity model for the problem of associating object detections across video frames. The proposed method exploits the sequential nature of video sequences to collect additional training data as new frames arrive and explore existing associations for additional training pairs. We show that this affinity model can efficiently be updated from the test data without explicit labelling enabling life-long learning. Experimental evaluation shows that this appearance based affinity model is capable of operating as the primary association score for multiple object tracking. In future work, we intend on extending this affinity model to perform online tracking and learning. Additionally, incorporating motion information and tracklet merging strategies offer potential avenues for reducing track fragmentation.

ACKNOWLEDGMENT

This work is supported by the Australian Coal Association Research Program (ACARP).

References

[1] H. Zhang, A. Geiger, and R. Urtasun, "Understanding High-Level Semantics by Modeling Traffic Patterns," in *International Conference on Computer Vision (ICCV)*, 2013.

[2] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun, "3D Traffic Scene Understanding from Movable Platforms," *Pattern Analysis and Machine Intelligence (PAMI)*, 2014.

[3] A. Torabi and G.-A. Bilodeau, "A Multiple Hypothesis Tracking Method with Fragmentation Handling," in *2009 Canadian Conference on Computer and Robot Vision*. Ieee, May 2009, pp. 8–15.

[4] P. Morton, B. Douillard, and J. Underwood, "Multi-sensor identity tracking with event graphs," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, May 2013, pp. 4742–4748.

[5] F. Solera, S. Calderara, and R. Cucchiara, "Learning to Divide and Conquer for Online Multi-Target Tracking," in *International Conference on Computer Vision (ICCV)*, 2015.

[6] C. Dicle, M. Sznaier, and O. Camps, "The way they move: Tracking multiple targets with similar appearance," in *International Conference on Computer Vision (ICCV)*, 2013.

[7] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking." *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 1, pp. 58–72, Jan. 2014.

[8] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, Jun. 2011, pp. 1201–1208.

[9] A. Bewley, V. Guizilini, F. Ramos, and B. Upcroft, "Online Self-Supervised Multi-Instance Segmentation of Dynamic Objects," in *International Conference on Robotics and Automation*. Hong Kong, China: IEEE, 2014, pp. 1296–1303.

[10] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.

[11] A. R. Zamir, A. Dehghan, and M. Shah, "Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs," in *European Conference on Computer Vision (ECCV)*, 2012.

[12] T. Finley and T. Joachims, "Supervised clustering with support vector machines," *Proceedings of the 22nd International Conference on Machine learning (ICML)*, pp. 217–224, 2005.

[13] S. Kim, S. Kwak, J. Feyereisl, and B. Han, "Online Multi-target Tracking by Large Margin Structured Learning," in *Computer Vision*, 7726th ed. Springer Berlin Heidelberg, 2013, pp. 98–111.

[14] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: Hybrid-Boosted multi-target tracker for crowded scene," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, Jun. 2009, pp. 2953–2960.

[15] W. Choi, "Near-Online Multi-target Tracking with Aggregated Local Flow Descriptor," *arXiv 1504.02340v1*.

[16] S.-H. Bae and K.-J. Yoon, "Robust Online Multi-Object Tracking based on Tracklet Confidence and Online Discriminative Appearance Learning," *Computer Vision and Pattern Recognition (CVPR)*, 2014.

[17] J. MacCormick and A. Blake, "A probabilistic exclusion principle for tracking multiple objects," *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 1, no. 1, pp. 57–71, 1999.

[18] A. Milan, K. Schindler, and S. Roth, "Detection-and Trajectory-Level Exclusion in Multiple Object Tracking," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013.

[19] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the Performance of ConvNet Features for Place Recognition," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015.

[20] A. Bewley and B. Upcroft, "From ImageNet to Mining : Adapting Visual Object Detection with Minimal Supervision," *Proceedings of the 10th International Conference on Field and Service Robotics (FSR)*, 2015.

[21] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Computer Vision and Pattern Recognition (CVPR)*, 2014.

[22] S. H. Walker and D. B. Duncan, "Estimation of the probability of an event as a function of several independent variables," *Biometrika*, vol. 54, no. 1-2, pp. 167–179, 1967.

[23] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets Robotics: The KITTI Dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[24] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking," *arXiv preprint*, 2015.

[25] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast Feature Pyramids for Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014.

[26] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," 2014.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in {P}ython," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[28] K. Bernardin and R. Stiefelhagen, "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics," *EURASIP Journal on Image and Video Processing*, 2008.

Chapter 6

Discussion

Through a series of published works, this thesis has presented both detection and tracking techniques which are capable of identifying novel objects, background distractors and learn to better associate detections over time. This chapter summarises its contributions, discusses the research outcomes and proposes avenues for future work.

6.1 Thesis Summary

This thesis has presented a series of methodologies for the detection and tracking of dynamic objects that are capable of adapting to the target domain. Object detection has advanced significantly due to recent developments in data driven approaches, particularly DCN based approaches. However, much of this success relies on having extensive training examples specific to the deployed environment. As it is often infeasible to collate such specific training sets ahead of time, this thesis considers methods which are able to learn and adapt their models during deployment. Two research questions around this issue were proposed where the first focuses on objects: *How to discover objects which may have appearance characteristics unknown by the detector?* While the second focuses on the background: *How to adapt an appearance based detector for deployment in new and novel environments with different background characteristics to the training data?*

The notion of learning during deployment was further extended to visual object tracking. While some prior tracking methods adapted their models online, they were restricted to single object tracking. Other tracking methods based on the assignment problem were identified as

better suited for the online tracking of multiple targets. However, these assignment based methods employed fixed similarity comparisons and are not able to adapt to the characteristics of the detected objects. This raised the final research question of this thesis: *How to adapt the assignment cost using data collected at deployment?*

Methodologies addressing each of the research questions were presented through a set of published papers and their contributions are summarised as follows.

6.1.1 Unsupervised Discovery of Novel Objects

Chapter 3 of this thesis is composed of a paper presented at the 2014 IEEE International Conference on Robotic Automation (ICRA) in Hong Kong. This paper presented a motion based detector which focuses on distinguishing between the static environment and multiple moving objects with unknown visual appearance. Building upon the prior work of Guizilini and Ramos [2013], this thesis explores the use of geometric constraints to identify the static background while introducing motion clustering to filter and partition dynamic points corresponding to the objects moving in the scene. By focusing on motion, the proposed framework makes no prior assumption on object appearance enabling it to identify any type of moving object compared to appearance based techniques that require manually annotated training samples.

The outcomes of motion based detection are two fold: first as a means of object detection without prior knowledge of appearance and second as a method for gathering positive training examples online. Both of these outcomes were demonstrated by using the discovered motion clusters to train an appearance based classifier without any pretraining. In this framework, a multi-class classifier was trained where the input is represented as the pixel position and colour while the output indicates either the object identity or background. For each frame, the output of the classifier was compared with the detected motion clusters where the classifier maintained appearance knowledge of known object identities and the motion detector would discover and create new object identities. Additionally, since each object is represented as a cluster of points, classifier errors on individual points were also used to gradually update the appearance model to better segment the moving objects from the background over time.

In Chapter 3, objects were represented as a collection of points for both the motion clustering and the appearance based classifier. This allows the classifier to rapidly adapt to new objects as multiple training points are attained from each object, even from the first frame the object

was discovered. Additionally, as most objects contain a mixture of colours, this complexity is naturally handled by the cluster based sampling. However, these benefits came at the cost of managing a large and varying amount of points in the k-NN classifier, prompting the use of whole object based representations and a DCN classifier in Chapters 4 and 5.

6.1.2 Background Models for Adapting Detectors

In Chapter 4, a state-of-the-art DCN based detector, pretrained on ImageNet data was applied in a surface mining environment. This scenario highlighted the need for adapting to the deployed environment as the pretrained detector performed poorly. While motion was shown to identify detector deficiencies during deployment, only miss detections are discovered while false detections persist. To address this issue, Chapter 4 introduced background modelling as an approach for adapting a pretrained generic object detector to novel and unstructured environments.

The background modelling techniques proposed in Chapter 4 were designed to operate alongside a pretrained DCN detector to combine both specific and generic appearance models. In contrast to directly retraining the detector, this approach only requires background data which can be collected with minimal manual annotation. Furthermore, a method was presented for gathering vast amounts of background patches which are novel to the detector for learning the background model. This allowed the background model to focus precisely on appearances that are likely to confuse the pretrained detector and ultimately improve the system's precision.

Chapter 4 contains two articles describing two different variants of the background model. The first variant was presented at the International Conference on Field and Service Robotics (FSR), 2015 in Toronto. This model clustered image patches to represent different types of potentially distracting background objects. In this earlier work, the background model was used to validate and suppress false detections made by the DCN if detection was similar to a background cluster. This was shown to significantly improve detection precision with only a minor drop in recall.

The second variant proposed in Chapter 4 replaces the suppression stage with a probabilistic fusion approach. This led to both better precision and recall as it capitalises on the information provided by both the detector and background model. Furthermore, the background model itself was modified to estimate the probability a detection belongs to background by considering the similarity to all background clusters jointly as opposed to only the nearest cluster. A paper

describing these extensions is included in Chapter 4 which is also published in the Journal of Field Robotics (JFR).

Another facet of the background models is the reuse of an intermediate DCN layer response to describe the visual appearance of an image region. Moreover, they are also practical since they are precomputed for the base detector. Compared to the simple point colour features used in Chapter 3 the DCN features better characterise the whole object appearance for the detector while also providing adequate clusters for the background model.

Given the small amount of labelled mining images available, the introduction of a background model was instrumental in improving the detector performance. In particular, when discriminating people and mine vehicles from the novel background, the performance of the joint cosine similarity variant exceeded both the pretrained DCN and a retrained DCN detector. Finally, these results were further improved by fusing the probabilistic estimates of the high precision joint cosine background model with the high recall retrained DCN. This fused method not only improved the F1 detection performance by 14% but also allowed novel mining specific vehicles to be detected despite being severely under represented in the training data.

Overall the fused detection system demonstrated superior performance over the pretrained DCN detector as it was able to exploit site specific appearances stored in the background model. The only exception to this was in a single test where the system was deployed in part of a mine-site containing store yard items and a processing plant. In these environments the background models underperformed the pretrained DCN since only day-time images from inside the pit and haul roads were used in the deployment specific training. This implies that the image reservoir used to form the background model must be extended with similar images should the detector be required to operate in such environments.

6.1.3 Learning an Improved Appearance Model for Object Tracking

The final outcome of this thesis is an appearance model suitable for tracking multiple objects. Chapter 5 revisited the idea of adapting an appearance based classifier for tracking but instead of learning a classifier to predict individual identities (as performed in Chapter 3), the focus is given to associating detections across frames in a tracking-by-detection framework. This decoupled approach offered multiple benefits as the detector capitalises on the holistic object representations (used in Chapter 4) while the tracker learns to better distinguish between object

instances.

The main contribution of the proposed tracking approach presented is the introduction of a probabilistic classifier based affinity model to estimate if a pair of detections represent the same object. A key benefit of using a classifier is the ability to update its parameters online. Compared to fixed similarity measures, this approach is able to improve the assignments made during deployment, ultimately leading to better tracking performance.

Another contribution of Chapter 5 is a self-supervised framework which exploits the sequential nature of video sequences to continuously collect training pairs as new frames arrive. Furthermore, this framework included a form of introspection where existing frame-to-frame associations are explored over larger temporal windows to find difficult pairs indicated by a disagreement between the affinity estimate and the chained probability. Updating the model with these difficult to associate pairs strengthened the systems robustness to common variations in an object's appearance.

The corresponding paper [Bewley et al., 2016b] describing this learning based object tracker was presented at the International Conference on Robotics and Automation (ICRA) 2016 in Stockholm. This paper included an extensive evaluation using several benchmark MOT sequences showing that the proposed appearance based tracking framework is comparable to other state-of-the-art based trackers which incorporate a myriad of position, motion and appearance cues. These experiments showed that the proposed appearance based affinity model is capable of operating as the primary association score for MOT. Moreover, the self-supervised nature of this approach allows it to improve its association over time by learning the scene specific object assignment cost without explicit supervision. Thus addressing the third and final research question: *How to adapt the assignment cost using data collected at deployment?*

6.2 Future Work

Each of the presented works in this thesis suggested potential avenues for continued development of each system. The following lists a number of possible future directions taking into account all contributions of this thesis.

6.2.1 Motion Clustering in 3D

This thesis has demonstrated that motion clustering is able to detect multiple moving objects without any prior knowledge of their appearance, image location or shape from a monocular sequence. While this technique is capable of handling a dynamic camera, ambiguities stemming from the monocular geometry causes the system not to detect objects that move in a parallel direction to the camera. Specifically, without knowledge of the depth to the object or the object's size, the apparent motion in the image cannot be disentangled from the camera motion.

Adding a second camera in a stereo configuration allows for the object depth to be estimated, thus facilitating the identification of objects moving parallel to the camera. However, as the accuracy of stereo depth estimation degrades rapidly with distance to the object, direct motion clustering on noisy displaced 3D points remain challenging. This could be alleviated by performing the point feature matching by alternating viewpoints each time step. If performed simultaneously with both cameras, i.e. creating virtual camera motion in both directions, the parallel motion ambiguity could be eliminated.

6.2.2 Better Features

In this work, motion clustering was presented as a means to identify previously unknown moving objects and to update a classifier to re-identify the objects moving in the next frame. However, simple colour and position features used in the classifier limited this solution to only learning what colours belonged where, without generalising to long term visual cues such as the shape of the objects found. While simple colour features worked well in urban datasets, when applied to a mining environment where vehicles frequently become dirty, such features lacked the information to segment the vehicles from the background. The DCN features used in Chapters 4 and 5 are better suited for this environment as they implicitly encode higher level information such as shape. Given the tremendous and rapid improvement in DCN architectures and training techniques, it is anticipated that even better features could be learnt for the specific environment along with robustness to various environment conditions. This could be naively achieved through employment of a state-of-the-art DCN architecture with large amounts of training data specific to the target environment or via a more data-efficient form of domain adaptation. Particularly, when there is insufficient training data in the target domain this can be framed as an unsupervised domain adaptation problem.

While unsupervised domain adaptation is an unresolved problem, many recently developed techniques for training neural networks to act as adversaries to one another [Goodfellow et al., 2014] offer a mechanism to transfer a learn model to new domains without extensive labels. One way this could be implemented is by simultaneously training a DCN for the primary detection task while also training it to fool a discriminator. This discriminator is trained with the objective of selecting if the input image came from the labelled source domain (e.g. ImageNet) or the unlabelled target domain (e.g. mine-site data) rendering it inherently independent of class labels in the target domain. Through the classification objective in the primary task the network maintains its ability to distinguish classes (e.g. vehicles from trees) but also learns to remove domain specific noise in the internal DCN representation. Furthermore, this approach could be extended to different environmental and lighting conditions to improve robustness by switching the domains to condition types (e.g. rain, night or dusty).

6.2.3 Adaptive Background Model

A method for learning a novel background model was demonstrated in a mine site scenario where the background is considerably different in visual appearance from the typical images used in benchmark datasets such as ImageNet. This presented model required minimal supervision in the form of verifying that no objects of interest exist in video segments. Mitigating this requirement would further aid the deployment of this system in different environments. Additionally, if the background model could be continuously updated online, this would likely lead to improved detection across different environmental conditions.

Taking a system engineering approach, the background model could be efficiently learnt by exploiting the property that not one but a fleet of haul trucks repeatedly traverse the same route, experiencing the same distractors. These distractors could be identified through reconciliation with other on-board sensors such as Global Positioning System (GPS) and Radio Frequency Identification (RFID). Additionally, if just one vehicle is equipped with additional sensors such as LiDAR or RaDAR, the geometry and location of these distractors could be verified and communicated to other vehicles traversing the same route. Alternatively, image only based SfM techniques can also make use of the geometric shape and appearance of the distractors gathered by frequent revisits to the same location, as recently shown by Hawke et al. [2015, 2017].

6.2.4 Validated Learning after Tracking

The tracking component presented in Chapter 5 focused heavily on the data association step for the purpose of linking detections across adjacent frames. Ideally objects should be re-identified after a period of miss detection or temporary occlusion. Despite using example associations spanning multiple frames to train the presented affinity model, it was unable to associate detections with a dramatic change in appearance which commonly coincides with miss detections and occlusions. This often resulted in fragmented tracks as the non-associated detections would then be used to form new tracklets.

This limitation could be surmised by the self-supervised training procedure and the fixed features extracted from a pretrained network. The self-supervision required a continuous stream of positive matching detections where mining examples with significant appearance change is difficult as the chaining procedure is dependant on the frame-to-frame associations. Additionally, the feature extraction network was trained for classification and fixed for the task of tracking. It is envisioned that by training the network with additional supervision for the task of re-identification, it would capture more instance specific detail (opposed to semantic details for classification) facilitating better associations in these difficult associations and thus also improving the self-supervision process. This additional supervision could come from supplementing the training dataset with additional data from multiple cameras or by exhaustively checking past tracks for potential matches and requesting validation through a form of active learning.

Furthermore, once distant tracklets have been associated, frames separating the fragmented tracklets could be searched for miss detections. This would require that the object trajectory is interpolated to predict the missing object location, then the learnt affinity model could be used to validate if the object was visible. Steps towards this extension have been undertaken in other related work [Bewley et al., 2016a] by considering motion estimation techniques to predict object location. Future work in this direction should also consider methods for combining evidence from tracklets on either side of the missing frame to explain the cause for the miss detection before using the example to update the detector model.

6.2.5 Extending with New Network Architectures and More Data

The work in the Chapter 4 and Chapter 5 made use of recent developments of deep learning as a generic tool for feature extraction and object recognition. Given the rapid pace of research in that field, it is expected that further advancements in deep learning and new network architectures would also improve the methods presented in this dissertation. In parallel to the work completed in this thesis, the RCNN detector used has undergone two revised network architectures Girshick [2015], Ren et al. [2015] each achieving both faster and better performance on standard single frame detection benchmarks. Beyond better single frame recognition performance it is also interesting to consider the potential of incorporating recurrent networks (similar to [Ondruska and Posner, 2016]) for predicting the future states of objects.

Finally, if extensive volumes of deployment specific training data was available, we could expect further improvements for vision-based sensing in such environments. The analysis of additional supervision in Chapter 5 highlighted the value of labelled data showing that modest performance gains are achievable through additional labels and the magnitude of these gains could potentially be extended further through further investment in labelled data. While the methods presented in this thesis learn from mostly unlabelled data observed during deployment, extensive trials under various environmental conditions and over multiple seasons would be required in order to speculate how these techniques perform in the long term.

6.2.6 Environmental Conditions

Beyond naively adding more labelled training data, it is important to capture the diversity of appearances expected during deployment. A significant concern with applying vision in outdoor and industrial environments such as a mine site is that it needs to handle environmental conditions including night, rain, dust, fog and snow. The effect of these adverse conditions on a computer vision systems level of perception varies from: a change in the appearance (at best), to a reduction in light where the objects are no longer visible (at worst). These effects also apply to human vision which is arguably the main sensing modality for existing industrial operations. As a result many industrial environments are controlled to enhance the visibility of objects under such conditions by adding high-visibility colours, lights, and reflectors.

While administrative controls lessen the likelihood of losing visibility, the challenge remains

with how to deal with the change in appearance. For example, the night-time sequence from the mine site trial in Chapter 4 showed a slight reduction in detection performance since personnel appear as horizontal white strips (see Figure 12 of Chapter 4). This scenario is another example of the training set being insufficient to cover the variation observed during deployment requiring additional specific training data to detect these cases. To ease the burden of labelling data across all seasons and conditions, the techniques suggested for extracting better features (Section 6.2.2) would also address the changes in environmental conditions. Namely, adversarial domain adaptation could be used towards training a condition invariant feature encoder where labels are only required for a subset of all expected conditions.

Finally, this work has only looked at enhancing detection and tracking using standard colour cameras and there are obvious benefits to using a range of sensing modalities. Particularly, in the extreme case where an object may not be visible to the camera but still visible to RADAR. The fusion of multiple modalities along with extensive trials in adverse conditions would be a suitable avenue for future work in further applying computer vision on mine sites and other industrial environments.

6.3 Conclusion

This thesis presented a vision-based detection and tracking system for environments where there is a lack of scene structure and prior knowledge of object appearances is limited. To achieve this, the focus of this thesis revolves around the ability of the sensing system to adapt to different or unseen objects and novel background by learning at the time of deployment. This problem was addressed on multiple fronts: detecting objects with unknown appearance, robustness to deployment specific background distractors, and learning during deployment to improve tracking performance. In working towards this, contributions were presented in three corresponding areas, namely, unsupervised object discovery through motion clustering, adapting an appearance based DCN object detector with a background model, and continuous affinity self-supervised learning for object tracking. These contributions aid towards enabling the rapid deployment of vision based detection and tracking in environments without requiring a large set of training examples. It is expected that the work presented in this thesis will also aid in facilitating future progression of vision based technologies by lessening the effort required to adapt data driven detection to different environments. Through the research presented in this thesis, it is hoped to inspire other researchers to consider applying computer vision beyond the standard benchmarks where learning from the environment is essential.

Appendix A

Problem Formulation

The problem of detection and tracking of multiple objects can be formulated within a DBN framework as follows: given a sequence of observations $\mathbf{z}_{0:t}$ the states $\mathbf{x}_t$ of m observable objects can be inferred. This is illustrated using the graphical plate model notation in Figure A.1. In this model the observations are the images captured which evolve over time based on both the camera motion $\mathbf{x}^s$ and that of independently moving targets within view of the camera. The states $\mathbf{x}^s$ and $\mathbf{x}^o$ may encode the position, motion, shape or visual appearance of the scene and objects respectively.

The detection component is responsible for detecting the number of objects and their locations, while tracking aims to associate each detection to the same object across frames. However, the number of target objects m is usually unknown and is constantly changing over time as objects enter and leave the Field Of View (FOV). When combined with unknown object dynamics and potential errors in the detection, estimating the state of each object becomes challenging. Since this thesis focuses on vision based detection and tracking, features such as shape and appearance can be informative to make robust data associations that are independent of motion.

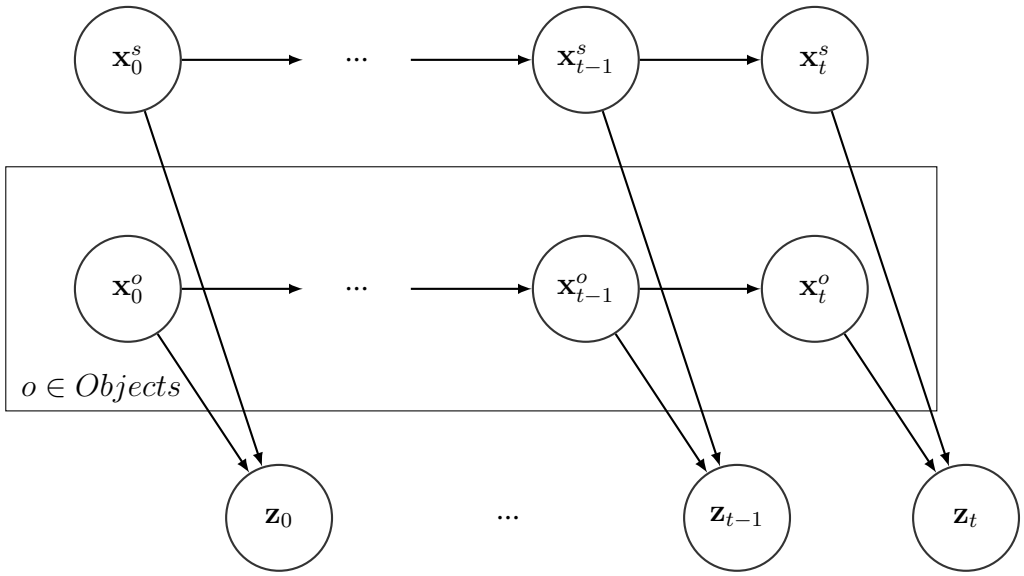


Figure A.1: The detection and tracking problem represented in the form of a graphical plate model. Note the observations $\mathbf{z}$ (camera images) are jointly influenced by both the sensor state $\mathbf{x}^s$ and the state of an arbitrary number of individual moving objects $\mathbf{x}^o$. The transition of states for each moving object (including the sensor) is modelled as a dynamic Bayesian network with first order Markov chain assumption.

Appendix B

Related Authored Publications

The following papers are included in this appendix as they are related to the works carried out in the thesis but do not form part of the main contribution.

The first paper [Bewley and Upcroft, 2013] presents an approach to nearest neighbour search with a constant time complexity when provided with 3D point cloud data as generated by a stereo-vision algorithm. This is demonstrated to speed up a classical clustering approach that relies heavily on nearest neighbour searches. Despite the use of unsupervised clustering in this paper, the main contribution is in the nearest neighbour search technique which requires 3D (stereo) data which was not used in the later parts of this thesis.

The second paper [Bewley et al., 2016a] included in this appendix focuses on speed and simplicity for multiple object tracking. It details a simple algorithm that employs classical tracking techniques applied to visual tracking where objects are represented as a bounding box in the image coordinate space. Since this paper does not employ any learning based techniques nor does it address any of the research questions, it was omitted from the main part of this thesis. However, it does provide useful insights to visual object tracking showing that advances made in the appearance models used for object detection implicitly improves object tracking.

Proceedings of Australasian Conference on Robotics and Automation, 2-4 Dec 2013, University of New South Wales, Sydney Australia

Advantages of Exploiting Projection Structure for Segmenting Dense 3D Point Clouds

Alex Bewley and Ben Upcroft
Queensland University of Technology, Brisbane, Australia
{aj.bewley,ben.upcroft}@qut.edu.au

Abstract

Timely and comprehensive scene segmentation is often a critical step for many high level mobile robotic tasks. This paper examines a projected area based neighbourhood lookup approach with the motivation towards faster unsupervised segmentation of dense 3D point clouds. The proposed algorithm exploits the projection geometry of a depth camera to find nearest neighbours which is time independent of the input data size. Points near depth discontinuations are also detected to reinforce object boundaries in the clustering process. The search method presented is evaluated using both indoor and outdoor dense depth images and demonstrates significant improvements in speed and precision compared to the commonly used Fast library for approximate nearest neighbour (FLANN) [Muja and Lowe, 2009].

1 Introduction

Modern sensors such as time-of-flight cameras, Microsoft Kinect, calibrated stereo cameras and high definition 3D LIDAR provide rich depth information with rates of over a million points per second. Timely and comprehensive scene understanding through interpretation of this data is critical to effective decision making in mobile robotic applications. A common approach to scene understanding is to subdivide sensor data into smaller meaningful portions which can later be classified if the target application requires. While segmentation is ubiquitous in the computer vision and robotics communities, many approaches assume specific domain knowledge in the form of model fitting or supervised learning. Unsupervised methods such as spatial clustering utilise nearest neighbour techniques to efficiently partition similar data into groups which differ from other groups.

One of the biggest drawbacks of using spatial clustering in robotics has been the excessive computational load primarily due to nearest neighbour searching. While considerable attention has been given to improving the computational efficiency of nearest neighbour problems [Elseberg *et al.*, 2012], the overhead of preprocessing the data into an organised structure often prohibits these methods for online robotic perception problems. The primary focus of efficient nearest neighbour searching in these domains is to minimise both the query time and storage complexity of high dimensional databases at the cost of increased build time. As data is constantly acquired in robotic applications, maintaining complex data structures online induces a significant computational overhead.

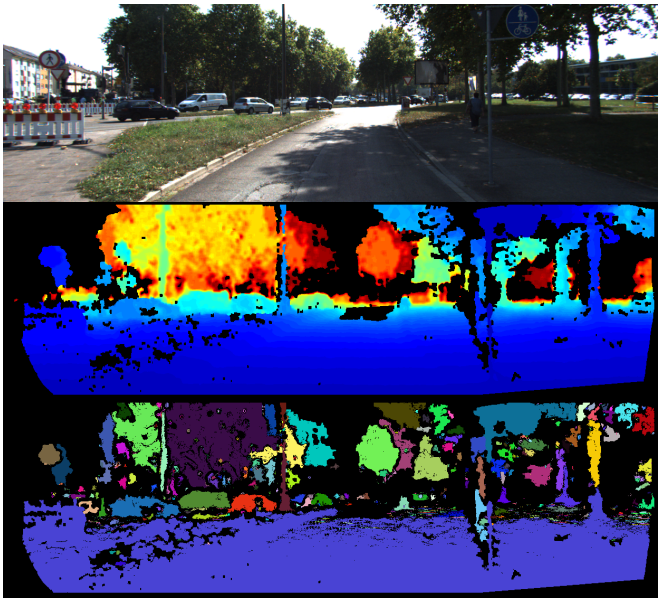


Figure 1: Top: visual frame depicting the scene for the reader's convenience. Middle: dense depth map representation of scene as input to clustering algorithm (computed using [Geiger *et al.*, 2011]). Blue represents close points and red is the maximum depth fixed at 150m. Bottom: Output of spatial clustering algorithm where each colour denotes a different cluster.

This paper takes an alternate approach for nearest neighbour searching for robotic applications where the domain is restricted to the Euclidean distance as a dissimilarity metric in $\mathbb{R}^3$ space. We exploit the 2.5D nature of dense depth maps by utilising the projection properties of modern sensors to selectively search neighbouring pixels. The key advantages of such an approach is through the constant time access to potential neighbours and we further show that the expected number of pixels examined for each kNN query is independent of the total image size. While the proposed approach can be adapted for various 3D sensors (Kinect-like, 3D LiDAR etc.) with simple projection models, we demonstrate the approach using stereo generated depth maps. Furthermore we show the k-nearest neighbour graph (kNNG) constructed using this approach is suitable for spatial clustering as shown in Fig. 1.

A key motivation for this work is in attempting to bring various sophisticated segmentation algorithms from the data mining community to robotics by addressing efficiency issues to meet real-time requirements. The work presented here was developed without knowledge of a similar approach described as $OrganizedNeighbor$ search in the point cloud library (PCL) [Rusu and Cousins, 2011]. To the best of the author's knowledge a detailed analysis of the $OrganizedNeighbor$ method isn't presented in any scientific publication. In this paper we use our independently developed method PANSearch which is based on the same projection principles as $OrganizedNeighbor$ with the aim of highlighting scenarios where such an approach is advantageous over a common tree based efficient search.

The paper is organised as follows: The next section positions the proposed approach among existing works. Depth maps and kNNG are described in section 3, while section 4 details efficient kNNG construction. Section 5 describes how the kNNG is used for unsupervised segmentation. Section 6 shows the performance of the proposed method before a conclusion is given in section 7.

2 RELATED WORK

Unsupervised spatial clustering methods are used across many disciplines, including computer vision, pattern recognition, data mining and more recently robotics [Klasing *et al.*, 2008; Moosmann and Fraichard, 2010; Bewley *et al.*, 2011]. These methods are favourable as they do not require prior knowledge of the number of clusters in the scene and make no assumptions on the shape or convexity of the clusters. One of the most common spatial clustering algorithms is DBSCAN [Ester *et al.*, 1996] which selects unvisited points at random to begin growing a cluster by expanding along neighbouring points according to a minimum local density criteria. Defining a fixed minimum point density to construct clusters from 3D sensor data such as in [Klasing *et al.*, 2008] can lead to over segmentation for distant objects as point sparsity increases with increased distance from the sensor. Cluster methods based on kNN graphs [Jarvis and Patrick, 1973; Ertoz *et al.*, 2003; Pauling *et al.*, 2009] resolves this problem of finding clusters with variable density by considering the mutual neighbourhood between points. Typically, the most computationally expensive step in these clustering methods is constructing the kNNG of the data. When the input is large and dense, these approaches prohibit the real-time performance required in robotic sensing.

The neighbourhood search problem has been extensively studied with a large focus on algorithms that can accommodate higher dimensional data and numerous similarity metrics. Associating neighbouring pixels with neighbouring 3D points has been used in surface normal estimation [Strom *et al.*, 2010; Douillard *et al.*, 2011] based on the 4 and 8 connected neighbourhood structures borrowed from the computer vision community[Felzenszwalb and Huttenlocher, 2004] for segmenting image data. These methods do not account for scenarios where the nearest neighbours may not be directly adjacent to the query point as common in 3D data with noise. A more relevant subset of the literature is concerned with closest point searching commonly associated to point cloud registration problems [Jost and Hugli, 2003; Elseberg *et al.*, 2012]. The majority of literature regarding efficient nearest neighbour searching can be found in the pattern recognition and data mining domains. The algorithms closely related to this work can be categorised into: space partitioning [Warnekar and Krishna, 1979; Nievergelt *et al.*, 1984; Meagher, 1982], data partitioning [Friedman *et al.*, 1977; Kolahdouzan and Shahabi, 2004; Freund *et al.*, 2007], or dimensionality reduction [Friedman *et al.*, 1975; Andoni *et al.*, 2006; Connor and Kumar, 2009; Min *et al.*, 2010].

Space partitioning methods can be as simple as dividing the search space into a grid structure [Warnekar and Krishna, 1979; Nievergelt *et al.*, 1984] to enable efficient constant time access to points binned into a specific location bucket. These methods typically require an excessive amount of memory when high resolution grids are used to minimise the number of points in each bucket. Several memory efficient space partitioning methods based on octrees [Meagher, 1982] have been proposed which have variable resolution based on local point densities. These methods require prior knowledge of the optimal partitioning resolution for efficient kNN searching in non-uniform data density distributions.

The second category divides the search space along partitions defined by points from the dataset. Binary tree based methods such as the kd-tree are commonly

used to recursively split the data along axial aligned partitions with the median (for kd-trees [Friedman *et al.*, 1977] or bounding box (for R-trees [Guttman, 1984]) of each subset. This data structure allows for efficient spatial querying by cutting down on the number of candidate points when traversing the tree. Many variants of the kd-tree have been proposed to improve search times to find approximate nearest neighbours (ANN) using randomised trees [Arya *et al.*, 1998] and random projections [Freund *et al.*, 2007]. Other data defined partitioning methods construct a Voronoi representation of the entire dataset [Kolahdouzan and Shahabi, 2004] or recursive subsets in the form of a k-means tree [Nister and Stewenius, 2006; Wang, 2011]. Muja and Lowe have developed the fast library for approximate nearest neighbours (FLANN) [Muja and Lowe, 2009] which selects between a hierarchical k-means tree and randomised k-d tree algorithm with automatic parameter optimisation, dependent on the dataset. The authors reported an order of magnitude speed improvement over other state-of-the-art ANN search implementations and as a result it is now part of several robotics related software packages[1,2]. In our experiments we use this implementation as a basis of comparison in speed and precision.

The final category is dimensionality reduction based search methods. The various tree based partitioning methods listed above share the undesirable characteristic that the number of partitions checked for each NN query increases with the dataset size. Locality preserving functions can be used to evaluate a relative index into the data structure which is derived from the query position, providing constant access time to each partition. For example Connor and Kumar [Connor and Kumar, 2009] use the Morton Z-order which is synonymous to a depth first search of an octree.

Instead of using tree-like data structures an increasingly common form of dimensionality reduction is in the use of hashing functions that transform the $\mathbb{R}^d$ feature vectors into a single binary value used as an index. Singh and Singh [Singh and Singh, 2012] compute multiple two-dimensional projections of higher dimensional data with each projection relative to a different point before partitioning into a two dimensional grid of buckets. Local sensitivity hashing (LSH) methods [Lv *et al.*, 2007; Andoni *et al.*, 2006] map similar points to the same bucket with high probability. These methods typically require a substantial amount of memory and are better suited to range queries.

These methods all require a form of preprocessing of the data before performing nearest neighbour queries on the dataset. In this paper we eliminate this step by exploiting the implicit structure in depth maps produced by modern 3D sensors and propose an efficient projection based kNN search algorithm which does not scale in search complexity with the dataset size.

[1]http://www.ros.org/wiki/flann

[2]http://rock-robotics.org

3 Dense 3D Map and Graph Representations

Recent approaches to real-time 3D segmentation utilising 2D lasers [Klasing *et al.*, 2009] have inspired our work in extending this concept to dense 2.5D depth maps. Storing the depth data in a structured order corresponding to the scan angle, enables efficient spatial access to 3D data, constrained by the perspective geometry of the sensors. Fig. 2 shows the projection of a sphere (representing the upper limit of potential neighbours) onto the image plane using a projective camera model applied to the depth map.

A depth map is defined in this paper as a two dimensional grid of pixels such that each pixel represents the spatial distance along the z-axis to a 3D point. Using a projective camera model the x and y coordinates can be easily computed using the pixel's depth and grid location. By replacing the single depth value with the corresponding 3D coordinate at each pixel location while keeping the grid structure, we allow for efficient spatial indexing as opposed to storing the point cloud as a simple list and then computing a k-d tree.

For the remainder of this paper we use the notation of a single lower case character to represent a pixel index or offset in the discrete grid space such as $p \in \mathbb{Z}^2$ while upper case characters signify the corresponding 3D point $P \in \mathbb{R}^3$. The grid storing 3D points can be thought of as a $\mathbb{Z}^2 \mapsto \mathbb{R}^3$ mapping function such that $P = \mathcal{I}_{xyz}(p)$. Using this alternative representation has several desirable attributes for nearest neighbour searching, including: the data is already organised into a grid as provided by the sensor, the grid format enables constant time random access to each cell, each grid cell maps to only a single 3D point and the projective nature of the data preserves 3D neighbourhoods in the 2D grid.

The kNNG construction problem can be defined as follows: given a set of n points where $P_i \in \mathbb{R}^3 \{i = 1 \dots n\}$, construct a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where each vertex corresponds to a pixel in the depth map and $\mathcal{E}$ is the set all pairs $\mathcal{E} : (P_i, P_j), i \neq j$ such that P_j is one of the k-nearest neighbours of P_i. A naive approach to finding the k-nearest neighbours to a given point P_i, is to search over every other point P_j keeping only the k points with the shortest distance. Constructing a kNNG using this approach has a time complexity of $\mathcal{O}(n^2)$. Now that these concepts have been introduced, we proceed to describe the PAN-Search algorithm.

4 PAN-Search: Projected Area Neighbourhood Searching

When dealing with dense depth maps as an input for kNNG based clustering, the dense 3D map representation enables efficient neighbourhood searching regardless of the input image size. In this section we present a simple search algorithm for finding kNN in 3D metric space for any given query point Q positioned at q in the dense 3D map $\mathcal{I}_{xyz}$.

Algorithm 1 shows the key steps for finding the kNN from a given location q in the dense 3D map $\mathcal{I}_{xyz}$. The critical concept behind this algorithm is exploiting the characteristic of projection models that the kNN points to Q are projected near to q in grid space. By initially selecting k candidate neighbours the 3D search space is bound to the kth closest point P_k shown in Fig. 2. The grid cells covered by the projection of this boundary are searched in an order designed to rapidly meet the terminating criteria described below.

Algorithm 1 Projected Area Neighbourhood Searching

Input: $\mathcal{I}_{xyz}$ ▷ each pixel is a 3D point
Input: q
Input: k
Input: f ▷ focal length in pixels
Input: Search Strategy *search_list*
Output: kNN

```
1:  function PAN-Search(I_xyz, q, k, f, search_list)
2:      Q ← I_xyz(q)
3:      for i = 1 to ||search_list|| do
4:          r_o ← ||o_i||          ▷ where o_i is the pixel offset
5:          P ← I_xyz(q + o_i)
6:          d ← ||P − Q||
7:          kNN.insert(o_ij, d, k)
8:          if kNN.size() > k then
9:              kNN.sort()                    ▷ sort by d
10:             kNN.pop()                     ▷ remove largest d
11:             r_max ← updateBoundary(kNN_k, I_z(q), f)
12:             if (r_o > r_max) then return kNN
13:             end if
14:         end if
15:     end for
16: end function
```

4.1 Image Search Pattern

The key to rapidly narrowing the search space is in the order in which neighbouring pixels are visited to seek the k closest points. The projection characteristic that neighbouring points on an object's surface are projected to a neighbouring location in the grid space we can efficiently traverse radially outward from the given query point q. The efficiency of this approach is degraded as the query point approaches the boundary of an object. As the ratio of boundary points to non-boundary points is small with real world depth maps the overall effect of boundary points is minimal.

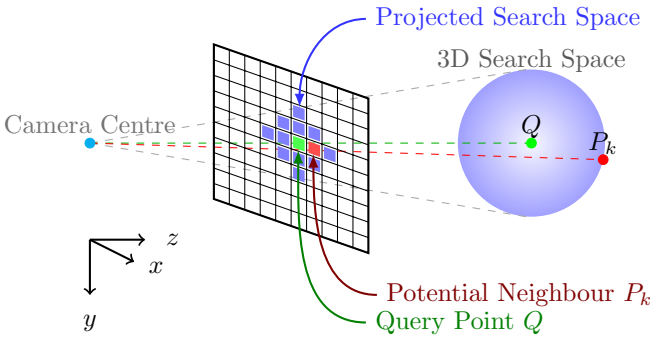


Figure 2: Illustration of the projected image search space from a sphere centred at the query point Q and radius of $\|QP_k\|$ where P_k is the kth closest potential neighbouring point found so far for Q.

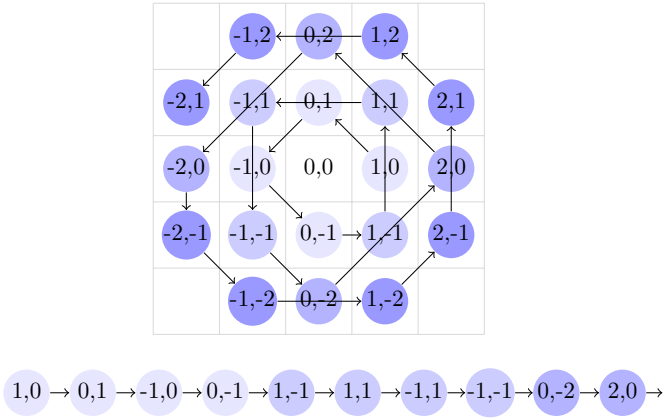


Figure 3: The index offsets for the first 20 candidate neighbours the in the fixed search pattern. The colour intensity of each node indicates the radial distance to the query position q.

The order in which neighbouring pixels are prioritised is determined by their proximity to the query pixel q. Fig 3 shows a potential sequence of index offsets to be applied to q in a radially growing search pattern. A simple yet efficient search strategy is to sort the pixel index offsets by their radius to the query pixel with an arbitrary order selected for offsets with equivalent radius in the image. This essentially forces points projected closer in the grid space to be searched completely before expanding the search radius. The sorting of these offsets can be computed offline as this pattern is kept constant and data independent for efficient online performance.

4.2 Terminating Criteria

Once a minimum of k valid pixels are considered, any point closer than P_k would also need to lie in the projected search boundary. As closer points are discovered using the radially expanding search pattern, the distance from the query point Q to the kth closest point decreases, contracting the projected search area towards Q. The search for kNN terminates when all pixels in the projected boundary with radius defined by the kth closest point has been checked.

While the true perspective projection of a sphere into the image plane casts an ellipse, we argue that a simple approximation of a circle around q as the image boundary saves on complex computation while maintaining reasonable precision in finding the true nearest neighbours. The maximum radius of the projected search boundary $\mathcal{B}_q$ in grid space for a camera like sensor is given by:

$$r_{max} = \frac{\|QP_k\|.f}{|Q_z|} \tag{1}$$

where $\|QP_k\|$ is the radius of the bounding sphere, f is the focal length in pixels and Q_z is the depth at q. The terminating condition to finish the search is when all pixels $p \in \mathcal{B}_q$ have been considered such that $\|pq\| \leq r_{max}$ (see line 11 and 12 of Algorithm 1).

Additionally a bailout condition can be set by limiting the size of the search pattern to terminate the search after a maximum number of pixel offsets are considered. This bailout can be used to simultaneously identify outliers while capping the maximum time spent searching for neighbours of outlier points.

Fig. 4 shows the relative number of pixels considered at each point in the depth map (shown in Fig. 1) with white pixels near object boundaries representing points where the bailout condition was met before the projected area could contract to the furthest offset in the search pattern. As expected, pixels around object boundaries take longer to compute compared to smooth continuous surfaces as the initial candidate neighbourhood set starts with points from both the foreground and background. Another interesting observation is that surfaces with high gradients relative to the view point carry high search cost while surfaces more parallel to the image plane have low search cost.

4.3 Search Complexity

The performance of this algorithm in terms of time is dependent on the data stored in the dense 3D map, with a worst-case complexity of $\mathcal{O}(l)$, where l is the maximum length of our ordered search pattern. However with the exception of occluded boundaries and discontinuities it is expected that the physical proximity of a given point is also proximal in the image grid space. Using a radially expanding search pattern, the expected run time is $\mathcal{O}(k)$ for points on smooth surfaces as opposed to k-d tree searching which is $\mathcal{O}(klog(n))$ where $1 \leq k \ll n$.

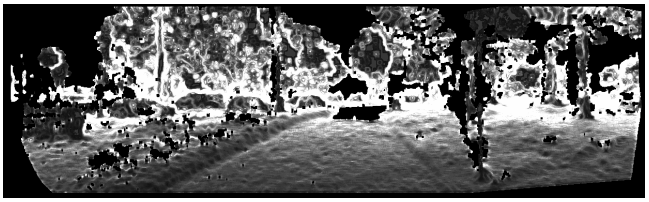


Figure 4: Heat map of where pixels with high run-times are located. The white (hottest) pixels coincide with the bailout condition.

5 Spatial Clustering using a kNN Graph

The kNNG can now be used as a basis for unsupervised segmentation in a region growing framework that spatially clusters the dense 3D input data. It is important to note that the kNNG in this raw form is directional as the kNN relationship is asymmetric, making the resulting clusters dependent on the initial seeding. To overcome this, we only expand a cluster along edges where both vertices are mutual neighbours, effectively cutting asymmetric edges. The resulting undirected graph enables deterministic clustering results with randomly selected seeds. For more details on this clustering method, the reader is encouraged to refer to [Jarvis and Patrick, 1973; Ertoz *et al.*, 2003].

This form of clustering is aided with knowledge of object boundaries discovered using the early bailout condition. As the resulting kNN points found up to the bailout are not guaranteed, we consider the corresponding vertex to be invalid and remove it from the graph. This not only prevents clusters growing across invalid neighbourhood edges, but also aids the clustering, as these points typically lie on an object's boundary.

6 Experiments and Results

The proposed approach is evaluated by comparing the speed and correctness against an efficient k-d tree implementation. The k-d tree software used in this evaluation is the OpenCV (version 2.4.3) implementation of FLANN (version 1.6.11), which can also be found PCL [Rusu and Cousins, 2011]. The k-d tree construction and search parameters for FLANN are left as default in all tests unless stated otherwise.

6.1 Datasets

We evaluate the performance of our system using 20 depth images derived from the ground truth disparity maps of the 2006 Middlebury dataset [Scharstein and

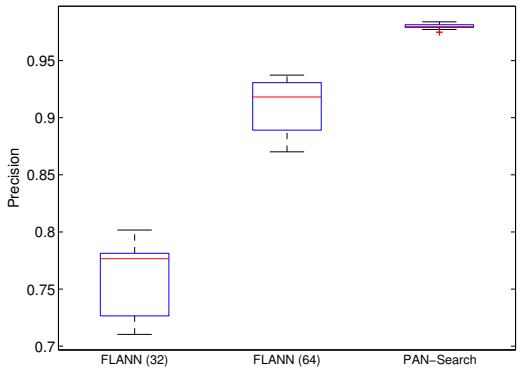


Figure 5: Average precision of 10 NN over 20 depth images using ELAS on the KITTI Dataset.

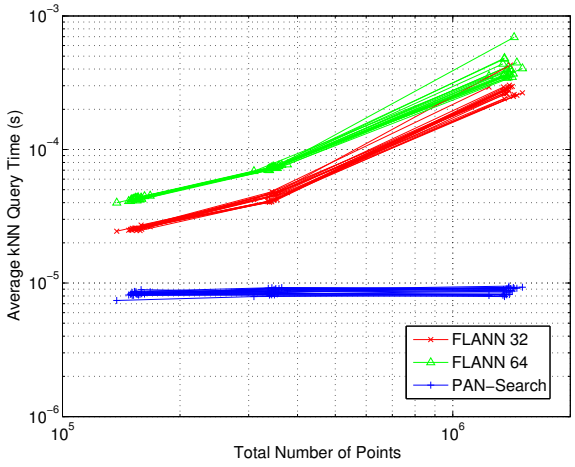


Figure 6: Average kNN query time (k=10) for 20 images from the Middlebury 2006 dataset using three different resolutions. Best viewed in colour.

Pal, 2007]. This dataset was selected as it contains various dense disparities at three difference resolution settings. While this provides a convenient benchmark to measure the speed performance for different input sizes (shown in Fig. 6) it lacks spatial variety as all stereo images are of indoor scenes or close objects. We address this by additionally comparing the performance of the proposed method using depth images generated from a stereo vision system of an outdoor traffic scene. For this we selected 20 frames of the KITTI dataset [Geiger *et al.*, 2012] from various sequences which contain several objects of potential interest such as cars and pedestrians. The stereo image pairs were converted to dense depth images using the Efficient LArge-scale Stereo (ELAS) matching algorithm published in [Geiger *et al.*, 2011] along with the stereo parameters from [Geiger *et al.*, 2012]. An example of the depth image generated using ELAS is shown in Fig. 1.

6.2 Correctness Evaluation

The precision of the proposed method is compared to that of FLANN with default search parameters of 32 maximum checks per query and 64 maximum checks. The results of applying these algorithms to the 20 depth images from the KITTI dataset are shown in Fig. 5. The exact nearest neighbours for each valid pixel were found using a brute force linear search considering potential neighbours across the entire image. Precision here is calculated as the number of points $P \in \mathcal{Q}_{knn}$ with $\|PQ\| \leq \|P'_kQ\|$ where P'_k is the true kth nearest neighbour from the brute force search. This experiment supports the projected circle approximation of the spherical boundary to limit the search space in PAN-Search.

6.3 Timing

A prototype of the PAN-Search algorithm was implemented in C++ and was deployed on a single core of a 2.2GHz Intel i7 processor with 8GB of RAM. OpenCV 2.4[3] was used for storing the neighbourhood graph as a matrix while the *priority_queue* from the standard template library is used for maintaining an ordered list of nearest neighbours for kNN searching. The performance of the proposed kNN graph construction algorithm was compared to the OpenCV implementation of FLANN on the same test computer.

Fig. 6 shows that PAN-Search's kNN time is independent of the total size of the input cloud as its search space is defined by the local neighbourhood, which is accessible in constant time due to the grid structure of the dense 3D image. The total time (in seconds) to construct the kNN graph for 20 frames from the KITTI datasets is shown in Fig. 7 where the value of k ranges from 1 to 20. Our $\mathcal{O}(k)$ method outperforms FLANN with 32 and 64 checks on this dataset up to $k = 14$ and 20 respectively. It is important to note that the recorded times is only for kNN graph construction and doesn't include the preprocessing time to construct the internal k-d trees used by FLANN. The k-d tree preprocessing took an average of 0.5 and 2.2 seconds on the KITTI and full resolution Middlebury datasets respectively.

7 Conclusion

In this paper we have shown a way of exploiting the 2.5D nature of dense depth maps by utilising the projection properties of modern sensor data to selectively

[3] http://opencv.org/

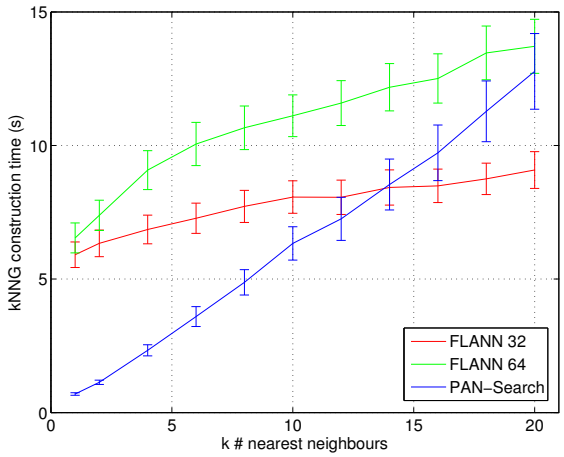


Figure 7: Run time vs. k showing mean and standard deviation for 20 frames from the KITTI traffic dataset. Note the y-axis represent the total time to construct the kNNG over an 0.5 Mega-pixel depth image.

search neighbouring pixels until no potential candidates remain. This provides an alternative solution to efficiently finding the k nearest neighbours to a set of given query points from a dense depth map with known camera projection matrix. This approach demonstrates a significant speed-up over common tree based search methods as it maintains a constant search time per pixel regardless of image size. Furthermore we experimentally evaluate range of k values where this method outperforms the ubiquitous k-d tree method. Additionally each kNN query can be computed in parallel providing a further speed-up on what is presented here. We believe that this is an important step towards constructing kNN graphs in real-time and ultimately real-time unsupervised segmentation of dense 3D data. In future work we intend on using other sources of information in addition to the 3D spatial location for constructing clusters based on the kNN graph connectivity.

Acknowledgments

This research has been supported by the Australian Coal Association Research Program (ACARP). The authors also would like to thank Kylie He and Peter Corke for valuable discussions and useful feedback regarding this work.

References

[Andoni *et al.*, 2006] A Andoni, M Datar, N Immorlica, P Indyk, and V Mirrokni. Locality-sensitive hashing using stable distributions. In *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice*, chapter 4, pages 55–67. MIT Press, 2006.

[Arya *et al.*, 1998] Sunil Arya, David M. Mount, Nathan S. Netanyahu, Ruth Silverman, and Angela Y. Wu. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *Journal of the ACM*, 45(6):891–923, November 1998.

[Bewley *et al.*, 2011] Alex Bewley, Rajiv Shekhar, Sam Leonard, Ben Upcroft, and Paul Lever. Real-time volume estimation of a dragline payload. In *2011 IEEE International Conference on Robotics and Automation*, pages 1571–1576, Shanghai, China, May 2011. IEEE.

[Connor and Kumar, 2009] Michael Connor and Piyush Kumar. Fast construction of k-nearest neighbor graphs for point clouds. *IEEE transactions on visualization and computer graphics*, 16(4):599–608, 2009.

[Douillard *et al.*, 2011] B. Douillard, J. Underwood, N. Kuntz, V. Vlaskine, A. Quadros, P. Morton, and A. Frenkel. On the segmentation of 3D LIDAR point clouds. In *2011 IEEE International Conference on Robotics and Automation*, pages 2798–2805. IEEE, May 2011.

[Elseberg *et al.*, 2012] Jan Elseberg, Stéphane Magnenat, Roland Siegwart, and Nüchter Andreas. Comparison of nearest-neighbor-search strategies and implementations for efficient shape registration. *Journal of Software Engineering for Robotics (JOSER)*, 3(1):2–12, 2012.

[Ertoz *et al.*, 2003] Levent Ertoz, Michael Steinbach, and Vipin Kumar. Finding Clusters of Different Sizes, Shapes, and Densities in Noisy, High Dimensional Data. In *Second SIAM International Conference on Data Mining*, 2003.

[Ester *et al.*, 1996] Martin Ester, Hans-peter Kriegel, Jörg Sander, and Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *2nd International Conference on Knowledge Discovery and Data Mining*, 1996.

[Felzenszwalb and Huttenlocher, 2004] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, 59(2):167–181, September 2004.

[Freund *et al.*, 2007] Yoav Freund, Sanjoy Dasgupta, Mayank Kabra, and Nakul Verma. Learning the structure of manifolds using random projections. *Neural Information Processing Systems*, 20:537, 2007.

[Friedman *et al.*, 1975] J.H. Friedman, F. Baskett, and L.J. Shustek. An Algorithm for Finding Nearest Neighbors. *Computers, IEEE Transactions on*, C-24(10):1000–1006, 1975.

[Friedman *et al.*, 1977] Jerome H Friedman, Jon Louis Bentley, and Raphael Ari Finkel. An Algorithm for Finding Best Matches in Logarithmic Expected Time. *ACM Trans. Math. Softw.*, 3(3):209–226, 1977.

[Geiger *et al.*, 2011] Andreas Geiger, Martin Roser, and Raquel Urtasun. Efficient Large-Scale Stereo Matching. In *Proceedings of the 10th Asian conference on*

Computer vision - Volume Part I, pages 25–38, Queenstown, New Zealand, 2011. Springer-Verlag.

[Geiger *et al.*, 2012] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. Ieee, June 2012.

[Guttman, 1984] Antonin Guttman. R-Trees: A Dynamic Index Structure for Spatial Searching. In *Proceedings of the 1984 ACM SIGMOD international conference on Management of data - SIGMOD '84*, page 47, New York, New York, USA, 1984. ACM Press.

[Jarvis and Patrick, 1973] R.a. Jarvis and E.a. Patrick. Clustering Using a Similarity Measure Based on Shared Near Neighbors. *IEEE Transactions on Computers*, C-22(11):1025–1034, November 1973.

[Jost and Hugli, 2003] Timothée Jost and H. Hugli. A multi-resolution ICP with heuristic closest point search for fast and robust 3D registration of range images. In *Fourth International Conference on 3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings.*, pages 427–433. IEEE, 2003.

[Klasing *et al.*, 2008] Klaas Klasing, Dirk Wollherr, and Martin Buss. A clustering method for efficient segmentation of 3D laser data. In *IEEE International Conference on Robotics and Automation*, pages 4043–4048. Ieee, May 2008.

[Klasing *et al.*, 2009] K. Klasing, D. Wollherr, and M. Buss. Realtime segmentation of range data using continuous nearest neighbors. In *IEEE International Conference on Robotics and Automation*, pages 2431–2436. Ieee, May 2009.

[Kolahdouzan and Shahabi, 2004] Mohammad Kolahdouzan and Cyrus Shahabi. Voronoi-based K nearest neighbor search for spatial network databases. In *Proceedings of the Thirtieth international conference on Very large data bases - Volume 30*, VLDB '04, pages 840–851. VLDB Endowment, 2004.

[Lv *et al.*, 2007] Qin Lv, William Josephson, Zhe Wang, Moses Charikar, and Kai Li. Multi-probe LSH: efficient indexing for high-dimensional similarity search. In *Proceedings of the 33rd international conference on Very large data bases*, pages 950–961, 2007.

[Meagher, 1982] D Meagher. Geometric modeling using octree encoding. *Computer Graphics and Image Processing*, 19(2):129–147, 1982.

[Min *et al.*, 2010] Kerui Min, Linjun Yang, John Wright, Lei Wu, Xian-Sheng Hua, and Yi Ma. Compact projection: Simple and efficient near neighbor search with practical memory requirements. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3477–3484. Ieee, June 2010.

[Moosmann and Fraichard, 2010] Frank Moosmann and Thierry Fraichard. Motion estimation from range images in dynamic outdoor scenes. In *2010 IEEE International Conference on Robotics and Automation*, pages 142–147. IEEE, May 2010.

[Muja and Lowe, 2009] Marius Muja and David G Lowe. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. In *International Conference on Computer Vision Theory and Application VISSAPP'09)*, pages 331–340. INSTICC Press, 2009.

[Nievergelt *et al.*, 1984] J. Nievergelt, Hans Hinterberger, and Kenneth C. Sevcik. The Grid File: An Adaptable, Symmetric Multikey File Structure. *ACM Transactions on Database Systems*, 9(1):38–71, January 1984.

[Nister and Stewenius, 2006] D Nister and H Stewenius. Scalable Recognition with a Vocabulary Tree. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06)*, volume 2, pages 2161–2168. IEEE, 2006.

[Pauling *et al.*, 2009] Frederick Pauling, Michael Bosse, and Robert Zlot. Automatic Segmentation of 3D Laser Point Clouds by Ellipsoidal Region Growing. In *Australasian Conference on Robotics and Automation (ACRA)*, 2009.

[Rusu and Cousins, 2011] Radu Bogdan Rusu and Steve Cousins. 3D is here: Point Cloud Library (PCL). In *2011 IEEE International Conference on Robotics and Automation*, pages 1–4. IEEE, May 2011.

[Scharstein and Pal, 2007] Daniel Scharstein and Chris Pal. Learning Conditional Random Fields for Stereo. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, June 2007.

[Singh and Singh, 2012] Vishwakarma Singh and Ambuj K. Singh. SIMP. In *Proceedings of the 15th International Conference on Extending Database Technology - EDBT '12*, page 492, New York, New York, USA, 2012. ACM Press.

[Strom *et al.*, 2010] J Strom, A Richardson, and E Olson. Graph-based segmentation for colored 3D laser point clouds. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2131–2136. IEEE, October 2010.

[Wang, 2011] Xueyi Wang. A fast exact k-nearest neighbors algorithm for high dimensional search using k-means clustering and triangle inequality. In *The 2011 International Joint Conference on Neural Networks*, volume 43, pages 1293–1299. IEEE, July 2011.

[Warnekar and Krishna, 1979] C.S. Warnekar and G. Krishna. A heuristic clustering algorithm using union of overlapping pattern-cells. *Pattern Recognition*, 11(2):85–93, January 1979.

SIMPLE ONLINE AND REALTIME TRACKING

Alex Bewley[†], *Zongyuan Ge*[†], *Lionel Ott*[◇], *Fabio Ramos*[◇], *Ben Upcroft*[†]

Queensland University of Technology[†], University of Sydney[◇]

ABSTRACT

This paper explores a pragmatic approach to multiple object tracking where the main focus is to associate objects efficiently for online and realtime applications. To this end, detection quality is identified as a key factor influencing tracking performance, where changing the detector can improve tracking by up to 18.9%. Despite only using a rudimentary combination of familiar techniques such as the Kalman Filter and Hungarian algorithm for the tracking components, this approach achieves an accuracy comparable to state-of-the-art online trackers. Furthermore, due to the simplicity of our tracking method, the tracker updates at a rate of 260 Hz which is over 20x faster than other state-of-the-art trackers.

Index Terms— Computer Vision, Multiple Object Tracking, Detection, Data Association

1. INTRODUCTION

This paper presents a lean implementation of a tracking-by-detection framework for the problem of multiple object tracking (MOT) where objects are detected each frame and represented as bounding boxes. In contrast to many batch based tracking approaches [1, 2, 3], this work is primarily targeted towards online tracking where only detections from the previous and the current frame are presented to the tracker. Additionally, a strong emphasis is placed on efficiency for facilitating realtime tracking and to promote greater uptake in applications such as pedestrian tracking for autonomous vehicles.

The MOT problem can be viewed as a data association problem where the aim is to associate detections across frames in a video sequence. To aid the data association process, trackers use various methods for modelling the motion [1, 4] and appearance [5, 3] of objects in the scene. The methods employed by this paper were motivated through observations made on a recently established visual MOT benchmark [6]. Firstly, there is a resurgence of mature data association techniques including Multiple Hypothesis Tracking (MHT) [7, 3] and Joint Probabilistic Data Association (JPDA) [2] which occupy many of the top positions of the MOT benchmark. Secondly, the only tracker that does not use the Aggregate Channel Filter (ACF) [8] detector is also the top ranked tracker, suggesting that detection quality could be holding back the other trackers. Furthermore, the trade-off between accuracy and speed appears quite pronounced, since the speed of most accurate trackers is considered too slow for realtime applications (see Fig. 1). With the prominence of traditional data association techniques among the top online and batch trackers along with the use of different detections used by the top tracker, this work explores how simple MOT can be and how well it can perform.

Keeping in line with Occam's Razor, appearance features beyond the detection component are ignored in tracking and only the bounding box position and size are used for both motion estimation and data association. Furthermore, issues regarding short-term and long-term occlusion are also ignored, as they occur very rarely and their explicit treatment intro-

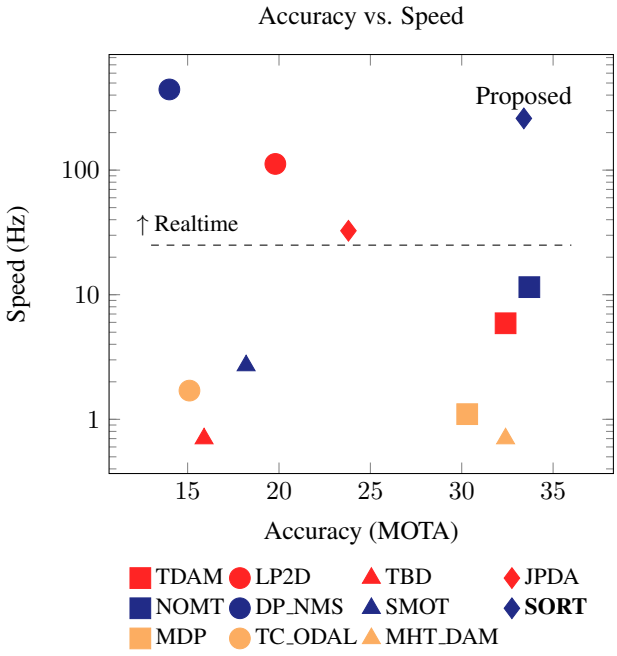


Fig. 1. Benchmark performance of the proposed method (**SORT**) in relation to several baseline trackers [6]. Each marker indicates a trackers accuracy and speed measured in frames per second (FPS) [Hz], i.e. higher and more right is better.

Thanks to ACARP for funding.

duces undesirable complexity into the tracking framework. We argue that incorporating complexity in the form of object re-identification adds significant overhead into the tracking framework – potentially limiting its use in realtime applications.

This design philosophy is in contrast to many proposed visual trackers that incorporate a myriad of components to handle various edge cases and detection errors [9, 10, 11, 12]. This work instead focuses on efficient and reliable handling of the common frame-to-frame associations. Rather than aiming to be robust to detection errors, we instead exploit recent advances in visual object detection to solve the detection problem directly. This is demonstrated by comparing the common ACF pedestrian detector [8] with a recent convolutional neural network (CNN) based detector [13]. Additionally, two classical yet extremely efficient methods, Kalman filter [14] and Hungarian method [15], are employed to handle the motion prediction and data association components of the tracking problem respectively. This minimalistic formulation of tracking facilitates both efficiency and reliability for online tracking, see Fig. 1. In this paper, this approach is only applied to tracking pedestrians in various environments, however due to the flexibility of CNN based detectors [13], it naturally can be generalized to other objects classes.

The main contributions of this paper are:

- We leverage the power of CNN based detection in the context of MOT.
- A pragmatic tracking approach based on the Kalman filter and the Hungarian algorithm is presented and evaluated on a recent MOT benchmark.
- Code will be open sourced to help establish a baseline method for research experimentation and uptake in collision avoidance applications.

This paper is organised as follows: Section 2 provides a short review of related literature in the area of multiple object tracking. Section 3 describes the proposed lean tracking framework before the effectiveness of the proposed framework on standard benchmark sequences is demonstrated in Section 4. Finally, Section 5 provides a summary of the learnt outcomes and discusses future improvements.

2. LITERATURE REVIEW

Traditionally MOT has been solved using Multiple Hypothesis Tracking (MHT) [7] or the Joint Probabilistic Data Association (JPDA) filters [16, 2], which delay making difficult decisions while there is high uncertainty over the object assignments. The combinatorial complexity of these approaches is exponential in the number of tracked objects making them impractical for realtime applications in highly dynamic environments. Recently, Rezatofighi et al. [2], revisited the JPDA formulation [16] in visual MOT with the goal to address the combinatorial complexity issue with an efficient approximation of the JPDA by exploiting recent developments in solving integer programs. Similarly, Kim et al. [3] used an appearance model for each target to prune the MHT graph to achieve state-of-the-art performance. However, these methods still delay the decision making which makes them unsuitable for online tracking.

Many online tracking methods aim to build appearance models of either the individual objects themselves [17, 18, 12] or a global model [19, 11, 4, 5] through online learning. In addition to appearance models, motion is often incorporated to assist associating detections to tracklets [1, 19, 4, 11]. When considering only one-to-one correspondences modelled as bipartite graph matching, globally optimal solutions such as the Hungarian algorithm [15] can be used [10, 20].

The method by Geiger et al. [20] uses the Hungarian algorithm [15] in a two stage process. First, tracklets are formed by associating detections across adjacent frames where both geometry and appearance cues are combined to form the affinity matrix. Then, the tracklets are associated to each other to bridge broken trajectories caused by occlusion, again using both geometry and appearance cues. This two step association method restricts this approach to batch computation. Our approach is inspired by the tracking component of [20], however we simplify the association to a single stage with basic cues as described in the next section.

3. METHODOLOGY

The proposed method is described by the key components of detection, propagating object states into future frames, associating current detections with existing objects, and managing the lifespan of tracked objects.

3.1. Detection

To capitalise on the rapid advancement of CNN based detection, we utilise the Faster Region CNN (**FrRCNN**) detection framework [13]. **FrRCNN** is an end-to-end framework that consists of two stages. The first stage extracts features and proposes regions for the second stage which then classifies the object in the proposed region. The advantage of this framework is that parameters are shared between the two stages creating an efficient framework for detection. Additionally, the network architecture itself can be swapped to any design which enables rapid experimentation of different architectures to improve the detection performance.

Here we compare two network architectures provided with **FrRCNN**, namely the architecture of Zeiler and Fergus (**FrRCNN(ZF)**) [21] and the deeper architecture of Simonyan and Zisserman (**FrRCNN(VGG16)**) [22]. Throughout this work, we apply the **FrRCNN** with default parameters learnt for the PASCAL VOC challenge. As we are only interested in pedestrians we ignore all other classes and only pass person detection results with output probabilities greater than 50% to the tracking framework.

Table 1. Comparison of tracking performance by switching the detector component. Evaluated on Validation sequences as listed in [12].

Tracker	Detector	Detection		Tracking	
		Recall	Precision	ID Sw	MOTA
MDP [12]	ACF	36.6	75.8	222	24.0
	FrRCNN(ZF)	46.2	67.2	245	22.6
	FrRCNN(VGG16)	**50.1**	76.0	**178**	33.5
Proposed	ACF	33.6	65.7	224	15.1
	FrRCNN(ZF)	41.3	72.4	347	24.0
	FrRCNN(VGG16)	49.5	**77.5**	274	**34.0**

In our experiments, we found that the detection quality has a significant impact on tracking performance when comparing the **FrRCNN** detections to **ACF** detections. This is demonstrated using a validation set of sequences applied to both an existing online tracker **MDP** [12] and the tracker proposed here. Table 1 shows that the best detector (**FrRCNN(VGG16)**) leads to the best tracking accuracy for both **MDP** and the proposed method.

3.2. Estimation Model

Here we describe the object model, i.e. the representation and the motion model used to propagate a target's identity into the next frame. We approximate the inter-frame displacements of each object with a linear constant velocity model which is independent of other objects and camera motion. The state of each target is modelled as:

$$\mathbf{x} = [u, v, s, r, \dot{u}, \dot{v}, \dot{s}]^T,$$

where u and v represent the horizontal and vertical pixel location of the centre of the target, while the scale s and r represent the scale (area) and the aspect ratio of the target's bounding box respectively. Note that the aspect ratio is considered to be constant. When a detection is associated to a target, the detected bounding box is used to update the target state where the velocity components are solved optimally via a Kalman filter framework [14]. If no detection is associated to the target, its state is simply predicted without correction using the linear velocity model.

3.3. Data Association

In assigning detections to existing targets, each target's bounding box geometry is estimated by predicting its new location in the current frame. The assignment cost matrix is then computed as the intersection-over-union (IOU) distance between each detection and all predicted bounding boxes from the existing targets. The assignment is solved optimally using the Hungarian algorithm. Additionally, a minimum IOU is imposed to reject assignments where the detection to target overlap is less than IOU_{min}.

We found that the IOU distance of the bounding boxes implicitly handles short term occlusion caused by passing targets. Specifically, when a target is covered by an occluding object, only the occluder is detected, since the IOU distance appropriately favours detections with similar scale. This allows both the occluder target to be corrected with the detection while the covered target is unaffected as no assignment is made.

3.4. Creation and Deletion of Track Identities

When objects enter and leave the image, unique identities need to be created or destroyed accordingly. For creating trackers, we consider any detection with an overlap less than IOU_{min} to signify the existence of an untracked object. The tracker is initialised using the geometry of the bounding box with the velocity set to zero. Since the velocity is unobserved at this point the covariance of the velocity component is initialised with large values, reflecting this uncertainty. Additionally, the new tracker then undergoes a probationary period where the target needs to be associated with detections to accumulate enough evidence in order to prevent tracking of false positives.

Tracks are terminated if they are not detected for T_{Lost} frames. This prevents an unbounded growth in the number of trackers and localisation errors caused by predictions over long durations without corrections from the detector. In all experiments T_{Lost} is set to 1 for two reasons. Firstly, the constant velocity model is a poor predictor of the true dynamics and secondly we are primarily concerned with frame-to-frame tracking where object re-identification is beyond the scope of this work. Additionally, early deletion of lost targets aids efficiency. Should an object reappear, tracking will implicitly resume under a new identity.

4. EXPERIMENTS

We evaluate the performance of our tracking implementation on a diverse set of testing sequences as set by the MOT benchmark database [6] which contains both moving and static camera sequences. For tuning the initial Kalman filter covariances, IOU_{min}, and T_{Lost} parameters, we use the same training/validation split as reported in [12]. The detection architecture used is the **FrRCNN(VGG16)** [22].

4.1. Metrics

Since it is difficult to use one single score to evaluate multi-target tracking performance, we utilise the evaluation metrics defined in [24], along with the standard MOT metrics [25]:

- MOTA(↑): Multi-object tracking accuracy [25].
- MOTP(↑): Multi-object tracking precision [25].
- FAF(↓): number of false alarms per frame.

Table 2. Performance of the proposed approach on MOT benchmark sequences [6].

Test Sequences		MOTA↑	MOTP↑	FAF↓	MT↑	ML↓	FP↓	FN↓	ID sw↓	Frag↓
TBD [20]	Batch	15.9	70.9	2.6%	6.4%	47.9%	14943	34777	1939	1963
ALExTRAC [5]	Batch	17.0	71.2	1.6%	3.9%	52.4%	9233	39933	1859	1872
DP_NMS [23]	Batch	14.5	70.8	2.3%	6.0%	40.8%	13171	34814	4537	3090
SMOT [1]	Batch	18.2	71.2	1.5%	2.8%	54.8%	8780	40310	1148	2132
NOMT [11]	Batch	**33.7**	71.9	**1.3%**	12.2%	44.0%	7762	32547	**442**	**823**
RMOT [4]	**Online**	18.6	69.6	2.2%	5.3%	53.3%	12473	36835	684	1282
TC_ODAL [17]	**Online**	15.1	70.5	2.2%	3.2%	55.8%	12970	38538	637	1716
TDAM [18]	**Online**	33.0	**72.8**	1.7%	**13.3%**	39.1%	10064	**30617**	464	1506
MDP [12]	**Online**	30.3	71.3	1.7%	13.0%	38.4%	9717	32422	680	1500
SORT (Proposed)	**Online**	**33.4**	72.1	**1.3%**	11.7%	**30.9%**	**7318**	32615	1001	1764

- MT(↑): number of mostly tracked trajectories. I.e. target has the same label for at least 80% of its life span.
- ML(↓): number of mostly lost trajectories. i.e. target is not tracked for at least 20% of its life span.
- FP(↓): number of false detections.
- FN(↓): number of missed detections.
- ID sw(↓): number of times an ID switches to a different previously tracked object [24].
- Frag(↓): number of fragmentations where a track is interrupted by miss detection.

Evaluation measures with (↑), higher scores denote better performance; while for evaluation measures with (↓), lower scores denote better performance. True positives are considered to have at least 50% overlap with the corresponding ground truth bounding box. Evaluation codes were downloaded from [6].

4.2. Performance Evaluation

Tracking performance is evaluated using the MOT benchmark [6] test server where the ground truth for 11 sequences is withheld. Table 2 compares the proposed method **SORT** with several other baseline trackers. For brevity, only the most relevant trackers, which are state-of-the-art online trackers in terms of accuracy, such as (**TDAM** [18], **MDP** [12]), the fastest batch based tracker (**DP_NMS** [23]), and all round *near* online method (**NOMT** [11]) are listed. Additionally, methods which inspired this approach (**TBD** [20], **ALExTRAC** [5], and **SMOT** [1]) are also listed. Compared to these other methods, **SORT** achieves the highest MOTA score for the online trackers and is comparable to the state-of-the-art method **NOMT** which is significantly more complex and uses frames in the near future. Additionally, as **SORT** aims to focus on frame-to-frame associations the number of lost targets (**ML**) is minimal despite having similar false negatives to other trackers. Furthermore, since **SORT** focuses on frame-to-frame associations to grow tracklets, it has the lowest number of lost targets by a considerable margin in comparison to the other methods.

4.3. Runtime

Most MOT solutions aim to push performance towards greater accuracy, often, at the cost of runtime performance. While slow runtime may be tolerated in offline processing tasks, for robotics and autonomous vehicles, realtime performance is essential. Fig. 1 shows a number of trackers on the MOT benchmark [6] in relation to both their speed and accuracy. This shows that methods which achieve the best accuracy also tend to be the slowest (bottom right in Figure 1). On the opposite end of the spectrum the fastest methods tend to have lower accuracy (top left corner in Figure 1). **SORT** combines the two desirable properties, speed and accuracy, without the typical drawbacks (top right in Figure 1). The tracking component runs at 260 Hz on single core of an Intel i7 2.5GHz machine with 16 GB memory.

5. CONCLUSION

In this paper, a simple online tracking framework is presented that focuses on frame-to-frame prediction and association. We showed that the tracking quality is very dependent on detection performance and by capitalising on recent developments in detection, state-of-the-art tracking quality can be achieved with only classical tracking methods. The presented framework achieves best in class performance with respect to both speed and accuracy, while other methods typically sacrifice one for the other. The simplicity of the presented framework makes it well suited as a baseline, allowing for new methods to focus on object re-identification to handle long term occlusion. As our experimentation has highlighted the importance of detection quality, future work will investigate a tightly coupled tracking and detection framework.

6. REFERENCES

[1] C. Dicle, M. Sznaier, and O. Camps, "The way they move: Tracking multiple targets with similar appear-

ance," in *International Conference on Computer Vision*, 2013.

[2] S. H. Rezatofighi, A. Milan, Z. Zhang, A. Dick, Q. Shi, and I. Reid, "Joint Probabilistic Data Association Revisited," in *International Conference on Computer Vision*, 2015.

[3] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple Hypothesis Tracking Revisited," in *International Conference on Computer Vision*, 2015.

[4] J. H. Yoon, M. H. Yang, J. Lim, and K. J. Yoon, "Bayesian Multi-Object Tracking Using Motion Context from Multiple Objects," in *Winter Conference on Applications of Computer Vision*, 2015.

[5] A. Bewley, L. Ott, F. Ramos, and B. Upcroft, "ALExTRAC: Affinity Learning by Exploring Temporal Reinforcement within Association Chains," in *International Conference on Robotics and Automation*. 2016, IEEE.

[6] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking," *arXiv preprint*, 2015.

[7] D. Reid, "An Algorithm for Tracking Multiple Targets," *Automatic Control*, vol. 24, pp. 843–854, 1979.

[8] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast Feature Pyramids for Object Detection," *Pattern Analysis and Machine Intelligence*, vol. 36, 2014.

[9] S. Oh, S. Russell, and S. Sastry, "Markov Chain Monte Carlo Data Association for General Multiple-Target Tracking Problems," in *Decision and Control*. 2004, pp. 735–742, IEEE.

[10] A. Perera, C. Srinivas, A. Hoogs, and G. Brooksby, "Multi-Object Tracking Through Simultaneous Long Occlusions and Split-Merge Conditions," in *Computer Vision and Pattern Recognition*. 2006, IEEE.

[11] W. Choi, "Near-Online Multi-target Tracking with Aggregated Local Flow Descriptor," in *International Conference on Computer Vision*, 2015.

[12] Y. Xiang, A. Alahi, and S. Savarese, "Learning to Track : Online Multi-Object Tracking by Decision Making," in *International Conference on Computer Vision*, 2015.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Advances in Neural Information Processing Systems*, 2015.

[14] R. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[15] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.

[16] Y. Bar-Shalom, *Tracking and data association*, Academic Press Professional, Inc., 1987.

[17] S. H. Bae and K. J. Yoon, "Robust Online Multi-Object Tracking based on Tracklet Confidence and Online Discriminative Appearance Learning," *Computer Vision and Pattern Recognition*, 2014.

[18] Y. Min and J. Yunde, "Temporal Dynamic Appearance Modeling for Online Multi-Person Tracking," oct 2015.

[19] A. Bewley, V. Guizilini, F. Ramos, and B. Upcroft, "Online Self-Supervised Multi-Instance Segmentation of Dynamic Objects," in *International Conference on Robotics and Automation*. 2014, IEEE.

[20] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun, "3D Traffic Scene Understanding from Movable Platforms," *Pattern Analysis and Machine Intelligence*, 2014.

[21] M. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in *European Conference on Computer Vision*, 2014.

[22] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *International Conference on Learning Representations*, 2015.

[23] H. Pirsiavash, D. Ramanan, and C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *Computer Vision and Pattern Recognition*. 2011, IEEE.

[24] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: HybridBoosted multi-target tracker for crowded scene," in *Computer Vision and Pattern Recognition*. 2009, IEEE.

[25] K. Bernardin and R. Stiefelhagen, "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics," *Image and Video Processing*, , no. May, 2008.

Appendix C

Postface

Shortly after submitting this dissertation for examination, a number of collaborative works were published following along the directions indicated in the future work (Section 6.2). These include: training a feature encoder to be invariant to appearance changes in the operating environment [Wulfmeier et al., 2017], switching detector models to suit the background of the operating environment [Hawke et al., 2017], training a feature encoder for the task of association in tracking [Wojke et al., 2017], and learning a tracker model which jointly incorporates appearance and motion while also suppressing background distractors [Kosiorek et al., 2017]. The remainder of this section briefly discusses the relation to avenues for future work (Section 6.2) while the reader should refer to the corresponding reference for self-contained detail of the proposed method.

Firstly, the work presented in [Wulfmeier et al., 2017] describes a technique for training a better feature extractor from limited training data through the use of adversarial learning [Goodfellow et al., 2014]. Here vast amounts of unlabelled data with various conditions causing changes in appearance can be modelled and aligned to the distribution of the labelled data. This facilitated the use of a trained classifier across all domains, demonstrating improved performance in free space segmentation across day, overcast and night conditions.

In [Hawke et al., 2017], a place specific object detector was demonstrated to out perform a recent DCN approach by exploiting local background appearance. This follows a similar direction as suggested with the adaptive background model where different models could be employed in different regions of the operational environment to overcome and adapt to known background distractors. This approach is very applicable to autonomous vehicles frequently

traversing the same route experiencing common distractors such as a truck travelling between a shovel and dump site.

On the tracking front, the simple position based tracker presented in Appendix B was extended to use appearance information to recover from miss detections and reduce fragmentation caused by drastic changes in appearance [Wojke et al., 2017]. This was achieved by going beyond affinity learning (restricted to working with fixed features) by optimising a whole network feature extractor for re-identification. Here the network output was constrained to model cosine similarity corresponding to the initial affinity metric used in Chapter 5 but the lower layers are fine-tuned to capture instance specific detail resulting in improved tracking.

Finally, Kosiorek et al. [2017] implicitly combines both appearance and position information into a unified tracking framework inspired by the attention mechanisms in human perception. This approach proposes a novel network architecture making use of recurrent layers, different forms of attention and the recently proposed dynamic filters [De Brabandere et al., 2016]. Recurrent layers within this architecture are able to caption specific detail of a target object and learn to adapt its filters to suppress distracting stimuli in the next frame.

Overall these works focus on the problem of learning better features for their associated task with invariance to appearance changed caused by environmental conditions and sudden changes. Additionally, the notion of adapting to changes in appearance of the background or the object itself were also explored. Together these works demonstrate the continued efforts towards effective and reliable use of computer vision in dynamic environments with minimal supervision, thus further extending the frontier of research pursued in this thesis.

References

Supreeth Achar, Bharath Sankaran, Stephen Nuske, Sebastian Scherer, and Sanjiv Singh. Self-supervised segmentation of river scenes. *Proceedings - IEEE International Conference on Robotics and Automation*, (May):6227–6232, 2011.

Chad Aeschliman, Johnny Park, and Avinash C. Kak. A probabilistic framework for joint segmentation and tracking. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1371–1378, jun 2010.

S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26 (11):1475–1490, nov 2004.

Motilal Agrawal, Kurt Konolige, and MR Blas. Censure: Center surround extremas for realtime feature detection and matching. *European Conference on Computer Vision (ECCV)*, pages 102–115, 2008.

A. Alahi, R. Ortiz, and P. Vandergheynst. FREAK: Fast Retina Keypoint. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 510–517, jun 2012.

Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2189–202, nov 2012.

Susana Alvarez and Maria Vanrell. Texton theory revisited: A bag-of-words approach to combine textons. *Pattern Recognition*, 45(12):4312–4325, dec 2012.

Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. People-tracking-by-detection and people-detection-by-tracking. *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, jun 2008.

A. Andriyenko, K. Schindler, and S. Roth. Discrete-continuous optimization for multi-target tracking. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, (June):1926–1933, jun 2012.

Mihael Ankerst, MM Breunig, HP Kriegel, and J Sander. OPTICS: ordering points to identify the clustering structure. In *ACM SIGMOD International Conference on Management of Data*, pages 49–60, 1999.

Hossein Azizpour and Ivan Laptev. Object detection using strongly-supervised deformable part models. In *European Conference on Computer Vision (ECCV)*, 2012.

Hossein Azizpour, Ali Sharif Razavian, Josephine Sullivan, Atsuto Maki, and Stefan Carlsson. From Generic to Specific Deep Representations for Visual Recognition. In *Computer Vision and Pattern Recognition (CVPR) Workshops*, Boston, Massachusetts, 2015. IEEE.

Boris Babenko, Ming-Hsuan Yang, and Serge Belongie. Visual Tracking with Online Multiple Instance Learning. *IEEE transactions on pattern analysis and machine intelligence*, dec 2010.

M. Bajracharya, B. Moghaddam, a. Howard, S. Brennan, and L. H. Matthies. A Fast Stereo-based System for Detecting and Tracking Pedestrians from a Moving Vehicle. *The International Journal of Robotics Research*, 28(11-12):1466–1485, jul 2009.

Bar-Shalom. *Tracking and data association*. Academic Press Professional, Inc., 1987.

Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding (CVIU)*, 110(3):346–359, 2008.

R. Benenson, M. Mathias, T. Tuytelaars, and L. Van Gool. Seeking the strongest rigid detector. In *Computer Vision and Pattern Recognition (CVPR)*, 2013.

Rodrigo Benenson, Markus Mathias, Radu Timofte, and L. Van Gool. Pedestrian detection at 100 frames per second. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2903–2910. IEEE, jun 2012.

Je rome Berclaz, Franc ois Fleuret, Engin Tu retken, and Pascal Fua. Multiple Object Tracking Using K-Shortest Paths Optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1806–1819, 2011.

Keni Bernardin and Rainer Stiefelhagen. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. *Image and Video Processing*, (May), 2008.

Alex Bewley and Ben Upcroft. Advantages of Exploiting Projection Structure for Segmenting Dense 3D Point Clouds. In *Australian Conference on Robotics and Automation*, 2013.

Alex Bewley and Ben Upcroft. From ImageNet to Mining : Adapting Visual Object Detection with Minimal Supervision. *Proceedings of the 10th International Conference on Field and Service Robotics (FSR)*, 2015.

Alex Bewley and Ben Upcroft. Background Appearance Modeling with Applications to Visual Object Detection in an Open Pit Mine. *Journal of Field Robotics (JFR)*, (submitted to), 2016.

Alex Bewley, Vitor Guizilini, Fabio Ramos, and Ben Upcroft. Online Self-Supervised Multi-Instance Segmentation of Dynamic Objects. In *International Conference on Robotics and Automation*, pages 1296–1303. IEEE, 2014.

Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple Online and Realtime Tracking. *arXiv preprint*, feb 2016a.

Alex Bewley, Lionel Ott, Fabio Ramos, and Ben Upcroft. ALExTRAC: Affinity Learning by Exploring Temporal Reinforcement within Association Chains. In *International Conference on Robotics and Automation (ICRA)*, 2016b.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. 2006.

Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory - COLT' 98*, pages 92–100, New York, New York, USA, 1998. ACM Press.

Paulo Vinicius Koerich Borges. Pedestrian Detection Based on Blob Motion Statistics. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(2):224–235, feb 2013.

A. Bosch, A. Zisserman, and X. Muoz. Scene Classification Using a Hybrid Generative/Discriminative Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(4):712–727, apr 2008.

Guillaume Bouchard and Bill Triggs. Hierarchical Part-Based Visual Object Categorization. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 710–715. IEEE, 2005.

Jean-yves Bouguet. Pyramidal implementation of the Lucas Kanade feature tracker. Technical report, Intel Corporation, Microprocessor Research Labs, 2000.

Thierry Bouwmans. Recent advanced statistical background modeling for foreground detection: A systematic survey. *Recent Patents on Computer Science*, 4(3):147–176, 2011.

GR Bradski. Computer Vision Face Tracking For Use in a Perceptual User Interface. 1998.

Leo Breiman. Statistical Modeling: The Two Cultures. *Statistical Science*, 16(3):199–215, 2001a.

Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001b.

Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. *Classification and regression trees*. Brooks/Cole, 1984.

Thomas Brox and Jitendra Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE transactions on pattern analysis and machine intelligence*, 33(3):500–13, mar 2011.

Thomas Brox, M Rousson, Rachid Deriche, and Joachim Weickert. Unsupervised segmentation incorporating colour, texture, and motion. *Computer Analysis of Images and Patterns*, (August 2003):353–360, 2003.

Gertjan J. Burghouts and Jan-Mark Geusebroek. Performance evaluation of local colour invariants. *Computer Vision and Image Understanding*, 113(1):48–62, jan 2009.

Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. BRIEF: Binary Robust Independent Elementary Features. In *European Conference on Computer Vision (ECCV)*, 2010.

J. Carreira and C. Sminchisescu. CPMC : Automatic Object Segmentation Using Constrained Parametric Min-cuts. *TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 34(7):1312–1328, 2012.

Daniel Castro, Urbano Nunes, and António Ruano. Feature Extraction for Moving Objects Tracking System in Indoor Environments. *Transform*, 2004.

Albert Y. C. Chen and Jason J. Corso. Propagating multi-class pixel labels throughout video frames. In *2010 Western New York Image Processing Workshop*, pages 14–17. Ieee, nov 2010.

O Chum and J Matas. Matching with PROSAC-progressive sample consensus. In *Computer Vision and Pattern Recognition (CVPR)*, 2005.

Dorin Comaniciu and Peter Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, may 2002.

Peter Corke, Rohan Paul, Winston Churchill, and Paul Newman. Dealing with shadows: Capturing intrinsic scene appearance for image-based outdoor localisation. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2085–2092, 2013.

Nico Cornelis, Bastian Leibe, Kurt Cornelis, and Luc Van Gool. 3D Urban Scene Modeling Integrating Recognition and Reconstruction. *International Journal of Computer Vision*, 78 (2-3):121–141, jul 2008.

Hingorani J Cox and Sunita L Hingorani. An Efficient Implementation of Reid's Multiple Hypothesis Tracking Algorithm and Its Evaluation for the Purpose of Visual Tracking. *Pattern Analysis and Machine Intelligence*, 18(2):138–150, feb 1996.

N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. Ieee, 2005.

Jesse Davis and Mark Goadrich. The relationship between Precision-Recall and ROC curves. *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pages 233–240, 2006.

Bert De Brabandere, Xu Jia, Tinne Tuytelaars, and Luc Van Gool. Dynamic Filter Networks. In *Neural Information Processing Systems (NIPS)*, Barcelona, Spain, 2016.

A P Dempster, N M Laird, and D B Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39 (1):1–38, 1977.

Jai Deng, Wei Dong, Richard Socher, Li-Jai Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, jun 2009.

S Dhanabal and S Chandramathi. A Review of various k-Nearest Neighbor Query Processing Techniques. *International Journal of Computer Applications*, 31(7):14–22, 2011.

Caglayan Dicle, Mario Sznaier, and Octavia Camps. The way they move: Tracking multiple targets with similar appearance. In *International Conference on Computer Vision (ICCV)*, 2013.

Tao Ding, Mario Sznaier, and Octavia I. Camps. Fast track matching and event detection. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, jun 2008.

Piotr Dollár, Zhuowen Tu, Pietro Perona, and Serge Belongie. Integral Channel Features. *BMVC 2009 London England*, pages 1–11, 2009a.

Piotr Dollár, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A benchmark. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009b.

Piotr Dollár, S Belongie, and Pietro Perona. The fastest pedestrian detector in the west. *British Machine Vision Conference (BMVC)*, 2010.

Piotr Dollar, Ron Appel, Serge Belongie, and Pietro Perona. Fast Feature Pyramids for Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1532–1545, aug 2014.

Pedro Domingos. A few useful things to know about machine learning. *Communications of the ACM*, 55(10):78, 2012.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. oct 2013.

Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative Unsupervised Feature Learning with Convolutional Neural Networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 766—-774, 2014.

Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. Wiley-Interscience, second edition, 2001.

A Elgammal, R. Duraiswami, D. Harwood, and L.S. Davis. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proceedings of the IEEE*, 90(7):1151–1163, jul 2002.

Levent Ertoz, Michael Steinbach, and Vipin Kumar. Finding Clusters of Different Sizes, Shapes, and Densities in Noisy, High Dimensional Data. In *Second SIAM International Conference on Data Mining*, 2003.

Martin Ester, Hans-peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *International Conference on Knowledge Discovery and Data Mining*, 1996.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2):303–338, sep 2009.

Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning Hierarchical Features for Scene Labeling. *Pattern Analysis and Machine Intelligence*, 35(8):1915–1929, 2013.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, apr 2007.

Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, 59(2):167–181, sep 2004.

Pedro F. Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, jun 2008.

Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–45, sep 2010.

Dave Ferguson, Michael Darms, Chris Urmson, and Sascha Kolski. Robotics Institute: Detection, Prediction, and Avoidance of Dynamic Obstacles in Urban Environments. In *Intelligent Vehicles Symposium*, pages 1149–1154. IEEE, 2008.

P. Fieguth and D. Terzopoulos. Color-based tracking of heads and other mobile objects at video frame rates. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 21–27. IEEE Comput. Soc, 1997.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

R. A. Fisher. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 7(2):179–188, sep 1936.

Yoav Freund and Robert E Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139, aug 1997.

Brendan J Frey and Delbert Dueck. Clustering by passing messages between data points. *Science (New York, N.Y.)*, 315(5814):972–6, feb 2007.

K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21(1):32–40, jan 1975.

D. Gabor. Theory of communication. *Journal of the Institute of Electrical Engineers*, 93: 429–457, 1946.

D. M. Gavrila and S. Munder. Multi-cue Pedestrian Detection and Tracking from a Moving Vehicle. *International Journal of Computer Vision*, 73(1):41–59, jul 2006.

Weina Ge and Robert T Collins. Multi-target Data Association by Tracklets with Unsupervised Parameter Estimation. In *British Machine Vision Conference (BMVC)*, 2008.

ZongYuan Ge, Christopher McCool, Conrad Sanderson, Alex Bewley, Zetao Chen, and Peter Corke. Fine-Grained Bird Species Recognition via Hierarchical Subset Learning.

In *International Conference on Image Processing*, volume 2015-Decem, pages 561–565, Quebec, Canada, sep 2015. IEEE.

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361. IEEE, jun 2012.

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research (IJRR)*, 2013.

Andreas Geiger, Martin Lauer, Christian Wojek, Christoph Stiller, and Raquel Urtasun. 3D Traffic Scene Understanding from Movable Platforms. *Pattern Analysis and Machine Intelligence (PAMI)*, 2014.

Ross Girshick. Fast R-CNN. In *International Conference on Computer Vision*, pages 1440–1448, 2015.

Ross B. Girshick, Jeff Donahue, Trevor Darrell, and J Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2014.

Ian J Goodfellow, Jean Pouget-abadie, Mehdi Mirza, Bing Xu, and David Warde-farley. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2672–2680, 2014.

N.J. Gordon, D.J. Salmond, and A. F M Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *Radar and Signal Processing*, 140(2):107–113, 1993.

H. Grimmett, R. Triebel, R. Paul, and I. Posner. Introspective classification for robot perception. *The International Journal of Robotics Research*, pages 1–20, aug 2015.

W. Eric L. Grimson, Chris Stauffer, Raquel Romano, and Lily Lee. Using adaptive tracking to classify and monitor activities in a site. In *Computer Vision and Pattern Recognition (CVPR)*, 1998.

Vitor Guizilini and Fabio Ramos. Online self-supervised segmentation of dynamic objects. In *International Conference on Robotics and Automation (ICRA)*, 2013.

J. Hafner, H.S. Sawhney, W. Equitz, M. Flickner, and W. Niblack. Efficient color histogram indexing for quadratic form distance functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(7):729–736, jul 1995.

Sam Hare, Amir Saffari, and Philip H. S. Torr. Struck: Structured output tracking with kernels. In *International Conference on Computer Vision (ICCV)*, pages 263–270. IEEE, nov 2011.

R.I. Hartley. In defence of the 8-point algorithm. In *Proceedings of IEEE International Conference on Computer Vision*, pages 1064–1070. IEEE Comput. Soc. Press, 1997.

R.˜I. Hartley and A Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

Jeffrey Hawke, Corina Gurau, Chi Hay Tong, and Ingmar Posner. Wrong Today , Right Tomorrow : Experience-Based Classification for Robot Perception. In *Field and Service Robotics*, 2015.

Jeffrey Hawke, Alex Bewley, and Ingmar Posner. What Makes a Place? Building Bespoke Place Dependent Object Detectors for Robotics. In *Intelligent Robots and Systems (IROS)*, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *http://arxiv.org/pdf/1406.4729.pdf*, jun 2014.

Shengfeng He, Qingxiong Yang, Rynson W.H. Lau, Jiang Wang, and Ming-Hsuan Yang. Visual Tracking via Locality Sensitive Histograms. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2427–2434. IEEE, jun 2013.

M. Heikkilä;, M. Pietikäinen, and J. Heikkilä. A Texture-based Method for Detecting Moving Objects. *Procedings of the British Machine Vision Conference 2004*, pages 21.1–21.10, 2004.

Marko Heikkilä and Matti Pietikäinen. A texture-based method for modeling the background and detecting moving objects. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):657–62, apr 2006.

Thomas Hofmann, Bernhard Schölkopf, and Alexander J. Smola. Kernel methods in machine learning. *The Annals of Statistics*, 36(3):1171–1220, jun 2008.

Brian G. Horn and Berthold K. P. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17:185—-203, 1981.

Jan Hosang, Rodrigo Benenson, and B Schiele. How good are detection proposals, really? In *British Machine Vision Conference (BMVC)*, 2014.

A Howard, L H Matthies, A Huertas, M Bajracharya, and A Rankin. Detecting pedestrians with stereo vision: safe operation of autonomous ground vehicles in dynamic environments. In *Int. Symp. of Robotics Research*, 2007.

Chang Huang, Bo Wu, and Ramakant Nevatia. Robust Object Tracking by Hierarchical Association of Detection Responses. In *European Conference on Computer Vision (ECCV)*, pages 788–801, 2008.

I. Guyon, B. Boser, and V. Vapnik. Automatic Capacity Tuning of Very Large VC-Dimension Classifiers. *Advances in Neural Information Processing Systems*, 5:147–155, 1993.

Michael Isard and Andrew Blake. Condensationconditional density propagation for visual tracking. *International journal of computer vision*, 29(1):5–28, 1998.

Anil K Jain and Richard C Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

Anand Singh Jalal. The State-of-the-Art in Visual Object Tracking. *Informatica*, 36:227–248, 2012.

R.a. Jarvis and E.a. Patrick. Clustering Using a Similarity Measure Based on Shared Near Neighbors. *IEEE Transactions on Computers*, C-22(11):1025–1034, nov 1973.

O. Javed, S. Ali, and M Shah. Online Detection and Classification of Moving Objects Using Progressively Improving Detectors. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 696–701. IEEE, 2005.

B. Julesz and J. R. Bergen. Human Factors and Behavioral Science : Textons, The Fundamental Elements in Preattentive Vision and Perception of Textures. *Bell System Technical Journal*, 62(6):1619–1645, jul 1983.

S.J. Julier and J.K. Uhlmann. Unscented Filtering and Nonlinear Estimation. *Proceedings of the IEEE*, 92(3):401–422, mar 2004.

Oswaldo Ludwig Junior, David Delgado, Valter Goncalves, and Urbano Nunes. Trainable classifier-fusion schemes: An application to pedestrian detection. In *2009 12th International IEEE Conference on Intelligent Transportation Systems*, number 1. IEEE, oct 2009.

Zdenek Kalal, Jiri Matas, and Krystian Mikolajczyk. P-N learning: Bootstrapping binary classifiers by structural constraints. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 49–56. Ieee, jun 2010.

Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Tracking-Learning-Detection. *IEEE transactions on pattern analysis and machine intelligence*, 34(7):1409–1422, dec 2012.

R E Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(Series D):35–45, 1960.

Fahad Shahbaz Khan, Joost van de Weijer, Sadiq Ali, and Michael Felsberg. Evaluating the Impact of Color on Texture Recognition. In *International Conference on Computer Analysis of Images and Patterns(CAIP)*, York, UK, 2013.

Zia Khan, Tucker Balch, and Frank Dellaert. MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE transactions on pattern analysis and machine intelligence*, 27(11):1805–19, nov 2005.

Chanho Kim, Fuxin Li, Arridhana Ciptadi, and James M. Rehg. Multiple Hypothesis Tracking Revisited. In *International Conference on Computer Vision (ICCV)*, 2015.

Sijong Kim, Jungwon Kang, and MJ Chung. Monocular vision based independently moving feature detection using image correspondences. In *2012 12th International Conference on Control, Automation and Systems*, pages 2181–2184, 2012.

B Kitt, F Moosmann, and C Stiller. Moving on to dynamic environments: Visual odometry using feature classification. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5551–5556. IEEE, oct 2010.

Esther B. Koller-Meier and Frank Ade. Tracking Multiple Objects Using the Condensation Algorithm. *Robotics and Autonomous Systems*, 34(2-3):93–105, feb 2001.

Adam R Kosiorek, Alex Bewley, and Ingmar Posner. Hierarchical Attentive Recurrent Tracking. In *Neural Information Processing Systems (NIPS)*, 2017.

Matej Kristan, Roman Pflugfelder, Aleš Leonardis, Jiri Matas, Fatih Porikli, Luka Cehovin, Georg Nebehay, Gustavo Fernandez, Toma Vojir, Adam Gatt, Ahmad Khajenezhad, Ahmed Salahledin, Ali Soltani-Farani, Ali Zarezade, Alfredo Petrosino, Anthony Milton, Behzad Bozorgtabar, Bo Li, Chee Seng Chan, Cherkeng Heng, Dale Ward, David Kearney, Dorothy Monekosso, Hakki Can Karaimer, Hamid R. Rabiee, Jianke Zhu, Jin Gao, Jingjing Xiao, Junge Zhang, Junliang Xing, Kaiqi Huang, Karel Lebeda, Lijun Cao, Mario Edoardo Maresca, Mei Kuan Lim, Mohamed El Helw, Michael Felsberg, Paolo Remagnino, Richard Bowden, Roland Goecke, Rustam Stolkin, Samantha Yueying Lim, Sara Maher, Sebastien Poullot, Sebastien Wong, Shin'Ichi Satoh, Weihua Chen, Weiming Hu, Xiaoqin Zhang, Yang Li, and Zhiheng Niu. The Visual Object Tracking VOT2013 Challenge Results. In *2013 IEEE International Conference on Computer Vision Workshops*, pages 98–111. IEEE, dec 2013.

Matej Kristan, Roman Pflugfelder, Ales Leonardis, Jiri Matas, Fatih Porikli, Luka ??ehovin, Georg Nebehay, Gustavo Fernandez, Tomas Vojir, Adam Gatt, Ahmad Khajenezhad, Ahmed Salahledin, Ali Soltani-Farani, Ali Zarezade, Alfredo Petrosino, Anthony Milton, Behzad Bozorgtabar, Bo Li, Chee Seng Chan, Cherkeng Heng, Dale Ward, David Kearney, Dorothy Monekosso, Hakki Can Karaimer, Hamid R. Rabiee, Jianke Zhu, Jin Gao, Jingjing Xiao, Junge Zhang, Junliang Xing, Kaiqi Huang, Karel Lebeda, Lijun Cao, Mario Edoardo Maresca, Mei Kuan Lim, Mohamed ELHelw, Michael Felsberg, Paolo Remagnino, Richard Bowden, Roland Goecke, Rustam Stolkin, Samantha Yue Ying Lim, Sara Maher, Sebastien Poullot, Sebastien Wong, Shin'Ichi Satoh, Weihua Chen, Weiming Hu, Xiaoqin Zhang, Yang Li, and Zhiheng Niu. The visual object tracking VOT2014 challenge results. In *European Conference on Computer Vision (ECCV)*, 2014.

Matej Kristan, Roman Pflugfelder, Aleš Leonardis, Jiri Matas, Luka Čehovin, Georg Nebehay, Tomáš Vojí\vr, Gustavo Fernandez, and Others. The Visual Object Tracking VOT2015 Challenge Results. In *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pages 564–586. IEEE, dec 2015.

Matej Kristan, Roman Pflugfelder, Aleš Leonardis, Jiri Matas, Luka Čehovin, Georg Nebehay, Tomáš Vojí\vr, Gustavo Fernandez, and Others. The visual object tracking {VOT2016} challenge results. *Eccvw*, pages 1–27, 2016.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep

Convolutional Neural Networks. *Advances In Neural Information Processing Systems*, pages 1–9, 2012.

Rolf G Kuehni. *Color space and its divisions: color order from antiquity to the present*. John Wiley & Sons, 2003.

H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955.

Neeraj Kumar, Li Zhang, and Shree Nayar. What is a Good Nearest Neighbors Algorithm for Finding Similar Patches in Images? In *Computer VisionECCV 2008*, pages 364—-378. Springer, 2008.

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1: 541–551, 1989.

Alain Lehmann, Bastian Leibe, and Luc Van Gool. PRISM: PRincipled Implicit Shape Model. In *British Machine Vision Conference (BMVC)*, 2009.

Bastian Leibe, Ales Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an implicit shape model. In *ECCV'04 Workshop on Statistical Learning in Computer Vision*, number May, pages 1–16, 2004.

Bastian Leibe, Aleš Leonardis, and Bernt Schiele. Robust Object Detection with Interleaved Categorization and Segmentation. *International Journal of Computer Vision*, 77(1-3):259–289, nov 2007.

Philip Lenz, Julius Ziegler, Andreas Geiger, and Martin Roser. Sparse scene flow segmentation for moving object detection in urban environments. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 926–932, Baden-Baden, Germany, jun 2011. IEEE.

Thomas Leung and Jitendra Malik. Representing and Recognizing the Visual Appearance of Materials using Three-dimensional Textons. *International Journal of Computer Vision*, 43 (1):29–44, 2001.

Stefan Leutenegger, Margarita Chli, and Roland Y. Siegwart. BRISK: Binary robust invariant scalable keypoints. In *International Conference on Computer Vision (ICCV)*, pages 2548–2555. IEEE, nov 2011.

Xi Li, Weiming Hu, Chunhua Shen, Zhongfei Zhang, Anthony Dick, and Anton Van Den Hengel. A survey of appearance models in visual object tracking. *ACM Transactions on Intelligent Systems and Technology*, 4(4):Article 58, sep 2013.

Yuan Li, Chang Huang, and Ram Nevatia. Learning to associate: HybridBoosted multi-target tracker for crowded scene. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2953–2960. IEEE, jun 2009.

Ce Liu, Jenny Yuen, Antonio Torralba, Josef Sivic, and William Freeman. SIFT flow: dense correspondence across different scenes. In *European Conference on Computer Vision (ECCV)*, volume 1, pages 28–42, 2008.

Ce Liu, Jenny Yuen, and Antonio Torralba. SIFT flow: dense correspondence across scenes and its applications. *IEEE transactions on pattern analysis and machine intelligence*, 33(5): 978–94, may 2011.

A. Lookingbill, J. Rogers, D. Lieb, J. Curry, and S. Thrun. Reverse Optical Flow for Self-Supervised Adaptive Autonomous Robot Navigation. *International Journal of Computer Vision*, 74(3):287–302, jul 2007.

David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, nov 2004.

B Lucas and T Kanade. An iterative image registration technique with an application to stereo vision. In *Image Understanding Workshop*, volume 130, pages 121–130, 1981.

J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. *International Conference on Computer Vision*, 1(1):57–71, 1999.

W James MacLean, Allan D Jepson, and Richard C Frecker. Recovery of Egomotion and Segmentation of Independent Object Motion Using the EM Algorithm. In *In Proceedings of the 5th British Machine Vision Conference*, pages 175–184. BMVA Press, 1994.

E. Maggio and A. Cavallaro. Learning Scene Context for Multiple Object Tracking. *IEEE Transactions on Image Processing*, 18(8):1873–1884, aug 2009.

Santiago Manen, Matthieu Guillaumin, and Luc Van Gool. Prime Object Proposals with Randomized Prim's Algorithm. In *International Conference on Computer Vision (ICCV)*, 2013.

Marta Marron, Miguel Angel Sotelo, Juan Carlos Garcia, David Fernandez, and Ignacio Parra. 3D-Visual Detection of Multiple Objects and Structural Features in Complex and Dynamic Indoor Environments. In *IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics*, pages 3373–3378. IEEE, nov 2006.

J Matas, O Chum, M Urban, and T Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, sep 2004.

Marina Meila and Jianbo Shi. Learning segmentation by random walks. In *In Advances in Neural Information Processing*, pages 470—-477, 2000.

Banislav Micusik and Tomas Pajdla. Multi-label image segmentation via max-sum solver. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1–6. IEEE, jun 2007.

Sebastian Mika, Gunnar Ratsch, Jason Weston, Bernhard Schölkopf, and Klaus-Robert Muller. Fisher discriminant analysis with kernels. *Ieee*, pages 41–48, 1999.

Anton Milan, Konrad Schindler, and Stefan Roth. Detection- and Trajectory-Level Exclusion in Multiple Object Tracking. In *Computer Vision and Pattern Recognition (CVPR)*, number June, pages 3682–3689. IEEE, jun 2013.

Anton Milan, Stefan Roth, and Konrad Schindler. Continuous energy minimization for multitarget tracking. *IEEE transactions on pattern analysis and machine intelligence*, 36 (1):58–72, jan 2014.

Marius Muja and David G Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *International Conference on Computer Vision Theory and Application*, pages 331–340. INSTICC Press, 2009.

Songhwai Oh Songhwai Oh, S. Russell, and S. Sastry. Markov chain Monte Carlo data association for general multiple-target tracking problems. *Conference on Decision and Control (CDC)*, 1:735–742, 2004.

T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, jul 2002.

Peter Ondruska and Ingmar Posner. Deep Tracking: Seeing Beyond Seeing Using Recurrent Neural Networks. *The Thirtieth AAAI Conference on Artificial Intelligence (AAAI)*, 2016.

Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2014.

L. Ott and F. Ramos. Unsupervised online learning for long-term autonomy. *The International Journal of Robotics Research*, 32(14):1724–1741, dec 2013.

F Pedregosa, G Varoquaux, A Gramfort, V Michel, B Thirion, O Grisel, M Blondel, P Prettenhofer, R Weiss, V Dubourg, J Vanderplas, A Passos, D Cournapeau, M Brucher, M Perrot, and E Duchesnay. Scikit-learn: Machine Learning in {P}ython. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

A.G.a. Perera, C. Srinivas, A. Hoogs, and G. Brooksby. Multi-Object Tracking Through Simultaneous Long Occlusions and Split-Merge Conditions. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2006.

Federico Pernici and Alberto Del Bimbo. Object Tracking by Oversampling Local Features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(12):2538–2551, dec 2014.

M. Piccardi. Background subtraction techniques: a review. In *2004 IEEE International Conference on Systems, Man and Cybernetics*, pages 3099–3104. Ieee, 2004.

David Pierce and Claire Cardie. Limitations of Co-training for Natural Language Learning from Large Datasets. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'01)*, pages 1–9, 2001.

Nicolas Pinto, David D. Cox, and James J. DiCarlo. Why is Real-World Visual Object Recognition Hard? *PLoS Computational Biology*, 4(1), 2008.

Hamed Pirsiavash, Deva Ramanan, and Charless C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1201–1208. IEEE, jun 2011.

J.R. Quinlan. Induction of Decision Trees. *Machine Learning*, 1(1):81–106, 1986.

Pekka Rantalankila, Juho Kannala, and Esa Rahtu. Generating Object Segmentation Proposals Using Global and Local Search. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2417–2424. IEEE, jun 2014.

Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. CNN Features off-the-shelf: an Astounding Baseline for Recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2014.

Vikas Reddy, Conrad Sanderson, and Brian C. Lovell. Improved Foreground Detection via Block-Based Classifier Cascade With Probabilistic Decision Integration. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(1):83–93, jan 2013.

Donald B Reid. An Algorithm for Tracking Multiple Targets. *Automatic Control*, 24:843–854, 1979.

Erik Reinhard and Tania Pouli. Colour spaces for colour transfer. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6626 LNCS:1–15, 2011.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Nips*, pages 1–10, 2015.

S. H. Rezatofighi, A. Milan, Z. Zhang, A. Dick, Q. Shi, and I. Reid. Joint Probabilistic Data Association Revisited. In *International Conference on Computer Vision (ICCV)*, 2015.

Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-Supervised Self-Training of Object Detection Models. *Workshops on Application of Computer Vision (WACV)*, 1:29–36, 2005.

David Ross, Jongwoo Lim, Ruei-Sung Lin, and Ming-Hsuan Yang. Incremental Learning for Robust Visual Tracking. *International Journal of Computer Vision*, 77(1-3):125–141, aug 2007.

Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision (ECCV)*, pages 430–443, 2006.

Peter J Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984.

Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: an efficient alternative to SIFT or SURF. In *International Conference on Computer Vision (ICCV)*, pages 2564–2571. Ieee, nov 2011.

Paul E. Rybski, Daniel Huber, Daniel D. Morris, and Regis Hoffman. Visual classification of coarse vehicle orientation using Histogram of Oriented Gradients features. In *2010 IEEE Intelligent Vehicles Symposium*, pages 921–928. Ieee, jun 2010.

Samuele Salti, Andrea Cavallaro, and Luigi Di Stefano. Adaptive appearance modeling for video tracking: survey and evaluation. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 21(10):4334–48, oct 2012.

Torsten Sattler, Bastian Leibe, and Leif Kobbelt. SCRAMSAC: Improving RANSAC's efficiency with a spatial consistency filter. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2090–2097. IEEE, sep 2009.

Dirk Schulz, Wolfram Burgard, Dieter Fox, and Armin B Cremers. People tracking with a mobile robot using sample-based joint probabilistic data association filters. *International Journal of Robotics Research*, 22(2):2003, 2003.

D. Serby, E.K. Meier, and Luc Van Gool. Probabilistic object tracking using multiple features. *International Conference on Pattern Recognition (ICPR)*, pages 184–187 Vol.2, 2004.

Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus, and Yann LeCun. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *International Conference on Learning Representations (ICLR 2014)*, dec 2014.

Yaser Sheikh, Omar Javed, and Takeo Kanade. Background Subtraction for Freely Moving Cameras. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1219–1225. Ieee, sep 2009.

J. Shi and C. Tomasi. Good features to track. In *Computer Vision and Pattern Recognition (CVPR)*, pages 593–600. IEEE Comput. Soc. Press, 1994.

Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

R. Sibson. SLINK: an optimally efficient algorithm for the single-link cluster method. *The Computer Journal*, 16(1):30–34, 1973.

Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations*, 2015.

A. Singh, S. Sawan, M. Hanmandlu, V.K. Madasu, and B.C. Lovell. An Abandoned Object Detection System Based on Dual Background Segmentation. In *International Conference on Advanced Video and Signal Based Surveillance*, pages 352–357. IEEE, sep 2009.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout : A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research (JMLR)*, 15:1929–1958, 2014.

Deqing Sun, Erik B. Sudderth, and Michael J. Black. Layered segmentation and optical flow estimation over time. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1768–1775. Ieee, jun 2012.

N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford. On the Performance of ConvNet Features for Place Recognition. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015.

MJ Swain and DH Ballard. Indexing via color histograms. In *International Conference on Computer Vision (ICCV)*, pages 390–393, 1990.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going Deeper with Convolutions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Carlo Tomasi and Takeo Kanade. Detection and Tracking of Point Features. *International Journal of Computer Vision*, 1991.

PHS Torr and A Zisserman. MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. *Computer Vision and Image Understanding*, 78:138–156, 2000.

A Torralba, R Fergus, and W Freeman. 80 Millions Tiny Images: a Large Dataset for Non-Parametric Object and Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:1958–1970, 2008.

Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1521–1528. IEEE, jun 2011.

C. Town and S. Moran. Robust Fusion of Colour Appearance Models for Object Tracking. In *Procedings of the British Machine Vision Conference 2004*, pages 70.1–70.10. British Machine Vision Association, 2004.

Rudolph Triebel, Hugo Grimmett, Rohan Paul, and Ingmar Posner. Introspective Active Learning for Scalable Semantic Mapping. In *Robotics: Science and Systems*, 2013.

Alina Trifan, António J R Neves, and Bernardo Cunha. Evaluation of color spaces for user-supervised color classification in robotic vision. In *International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV).*, 2013.

Oncel Tuzel, Fatih Porikli, and Peter Meer. Human Detection via Classification on Riemannian Manifolds. In *mputer Vision and Pattern Recognition*, pages 1–8. IEEE, jun 2007.

Christos Tzomakas and Werner von Seelen. Vehicle Detection in Traffic Scenes Using Shadows. Technical report, Institut fur Neuroinformatik, Ruhr-Universitat Bochum, 1998.

J. R. R. Uijlings, K. E. A. Sande, T. Gevers, and A. W. M. Smeulders. Selective Search for Object Recognition. *International Journal of Computer Vision*, 104(2):154–171, apr 2013.

Koen E A van de Sande, Theo Gevers, and Cees G M Snoek. Evaluating color descriptors for object and scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1582–96, sep 2010.

J. van de Weijer, T. Gevers, and A.D. Bagdanov. Boosting color saliency in image feature detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):150–156, jan 2006.

Joost Van De Weijer, Cordelia Schmid, and Jakob Verbeek. Learning color names from real-world images. *Computer Vision and Pattern Recognition (CVPR)*, 2007.

Vlamimir Vapnik. *Statistical learning theory*. 1998.

Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 511–518. IEEE Comput. Soc, 2001.

Paul Viola and Michael J Jones. Detecting pedestrians using patterns of motion and appearance. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 734–741 vol.2. Ieee, 2003.

Paul Viola and Michael J. Jones. Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57(2):137–154, may 2004.

S. H. Walker and D. B. Duncan. Estimation of the probability of an event as a function of several independent variables. *Biometrika*, 54(1-2):167–179, 1967.

Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple Online and Realtime Tracking with a Deep Association Metric. In *International Conference on Image Processing*, 2017.

Bo Wu and Ram Nevatia. Tracking of multiple, partially occluded humans based on static body part detection. In *Computer Vision and Pattern Recognition*, pages 951–958. IEEE, 2006.

Bo Wu and Ram Nevatia. Detection and Tracking of Multiple, Partially Occluded Humans by Bayesian Combination of Edgelet based Part Detectors. *International Journal of Computer Vision*, 75(2):247–266, jan 2007.

Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online Object Tracking: A Benchmark. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2411–2418. IEEE, jun 2013.

Markus Wulfmeier, Alex Bewley, and Ingmar Posner. Addressing Appearance Change in Outdoor Robotics with Adversarial Domain Adaptation. In *Intelligent Robots and Systems (IROS)*, 2017.

Hanxuan Yang, Ling Shao, Feng Zheng, Liang Wang, and Zhan Song. Recent advances and trends in visual tracking: A review. *Neurocomputing*, 74(18):3823–3831, nov 2011.

Shao-Wen Yang and Chieh-Chih Wang. Simultaneous egomotion estimation, segmentation, and moving object detection. *Journal of Field Robotics*, 28(4):565–588, 2011.

Rui Yao, Qinfeng Shi, Chunhua Shen, Yanning Zhang, and Anton van den Hengel. Part-Based Visual Tracking with Online Latent Structural Learning. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2363–2370. IEEE, jun 2013.

Alper Yilmaz, Omar Javed, and Mubarak Shah. Object Tracking: A Suvey. *ACM Computing Surveys*, 38(4):13–es, dec 2006.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks ? In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

Qian Yu, Gérard Medioni, and Isaac Cohen. Multiple target tracking by spatio-temporal Monte Carlo Markov chain data association. In *Computer Vision and Pattern Recognition (CVPR)*, volume 31. IEEE, dec 2007.

Amir Roshan Zamir, Afshin Dehghan, and Mubarak Shah. Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs. In *European Conference on Computer Vision (ECCV)*, 2012.

Jian Zhang, Sakrapee Paisitkriangkrai, and Chunhua Shen. An overview of fast pedestrian detection: Feature selection and cascade framework of boosted features. In *International Conference on Multimedia*, pages 1566–1567. IEEE, jun 2009.

Lu Zhang and Laurens van der Maaten. Structure Preserving Object Tracking. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, (1):1838–1845, jun 2013.

Shengyan Zhou Shengyan Zhou and K. Iagnemma. Self-supervised learning method for unstructured road detection using Fuzzy Support Vector Machines. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 1183–1189, 2010.

Jiajun Zhu, Michael Steven Montemerlo, Christopher Paul Urmson, and Andrew Chatham. Object Detection and Classification for Autonomous Vehicles, 2012.

Long (Leo) Zhu, Yuanhao Chen, Alan Yuille, and William Freeman. Latent hierarchical structural learning for object detection. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1062–1069. Ieee, jun 2010.

Xiaojin Zhu. Semi-Supervised Learning Literature Survey Contents. *Sciences New York*, 10 (1530):10, 2008.

Ming-ming Cheng Ziming, Zhang Wen-yan Lin, and Philip Torr. BING : Binarized Normed Gradients for Objectness Estimation at 300fps. In *Computer Vision and Pattern Recognition (CVPR)*, 2014.

CL Zitnick and P Dollár. Edge Boxes: Locating Object Proposals from Edges. In *European Conference on Computer Vision (ECCV)*, 2014.

Zoran Zivkovic and Ferdinand van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 27(7):773–780, may 2006.