UNIVERSITY OF KWAZULU-NATAL

GAUSSIAN MIXTURE MODEL CLASSIFIERS FOR DETECTION AND TRACKING IN UAV VIDEO STREAMS

Treshan Pillay

Supervised By: Mr Bashan Naidoo

2017

GAUSSIAN MIXTURE MODEL CLASSIFIERS FOR DETECTION AND TRACKING IN UAV VIDEO STREAMS

Treshan Pillay

Supervised By: Mr Bashan Naidoo

Submitted in fulfilment of the degree of Master of Engineering,

School of Engineering, University of KwaZulu-Natal, Durban, South Africa

2017

As the candidate's supervisor I agree to the submission of this dissertation.

Date of Submission: _____

Supervisor: _____

Mr Bashan Naidoo

COLLEGE OF AGRICULTURE, ENGINEERING AND SCIENCE

DECLARATION 1 - PLAGIARISM

I, Treshan Pillay declare that

1. The research reported in this thesis, except where otherwise indicated, is my original research.

2. This thesis has not been submitted for any degree or examination at any other university.

3. This thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

4. This thesis does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

a. Their words have been re-written but the general information attributed to them has been referenced

b. Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.

5. This thesis does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the thesis and in the References sections.

Signed: _____

COLLEGE OF AGRICULTURE, ENGINEERING AND SCIENCE

DECLARATION 2 - PUBLICATIONS

DETAILS OF CONTRIBUTION TO PUBLICATIONS that form part and/or include research presented in this thesis (include publications in preparation, submitted, *in press* and published and give details of the contributions of each author to the experimental work and writing of each publication)

Publication 1:

T. Pillay, B. Naidoo, "Gaussian Mixture Model classifiers for detection in UAV video streams", [Submitted for review to International Journal of Remote Sensing], 2017

Publication 2:

T. Pillay, B. Naidoo, "Gaussian Mixture Model classifiers for tracking in UAV video streams", [Submitted for review to International Journal of Remote Sensing], 2017

Signed: _____

Acknowledgements

I would like to thank my supervisor, Mr Bashan Naidoo, for his academic support, mathematical expertise and contributions towards the fulfilment of my degree of Master of Engineering. He has been accommodating in the challenges that I faced with regards to being based in a different province and conducting research on a part-time basis.

I extend my gratitude towards the CSIR - Defence, Peace, Safety and Security, for their financial support and creating an apt research environment, which includes experts, research based courses, research tools and equipment.

In addition, I would like to thank, Mr Herman Le Roux (CSIR employee) and Mr Asheer Bachoo (former CSIR employee) for their initial academic support which provided a strong foundation within the overall research space.

I thank my family, both immediate and extended, for their on-going support, encouragement and patience during my studies. The values that my parents have instilled in me has enabled me to take on all the challenges that I have faced.

Lastly, I would like to thank my wife, Raveshni Durgiah, for all the advice, motivation, support and strong belief in me. She has been with me every step of way including late nights and early mornings. She has inspired me to pursue my goals and has played a key role in making this possible.

Abstract

Manual visual surveillance systems are subject to a high degree of human-error and operator fatigue. The automation of such systems often employs detectors, trackers and classifiers as fundamental building blocks. Detection, tracking and classification are especially useful and challenging in Unmanned Aerial Vehicle (UAV) based surveillance systems. Previous solutions have addressed challenges via complex classification methods. This dissertation proposes less complex Gaussian Mixture Model (GMM) based classifiers that can simplify the process; where data is represented as a reduced set of model parameters, and classification is performed in the low dimensionality parameter-space. The specification and adoption of GMM based classifiers on the UAV visual tracking feature space formed the principal contribution of the work. This methodology can be generalised to other feature spaces.

This dissertation presents two main contributions in the form of submissions to ISI accredited journals. In the first paper, objectives are demonstrated with a vehicle detector incorporating a two stage GMM classifier, applied to a single feature space, namely Histogram of Oriented Gradients (HoG). While the second paper demonstrates objectives with a vehicle tracker using colour histograms (in RGB and HSV), with Gaussian Mixture Model (GMM) classifiers and a Kalman filter.

The proposed works are comparable to related works with testing performed on benchmark datasets. In the tracking domain for such platforms, tracking alone is insufficient. Adaptive detection and classification can assist in search space reduction, building of knowledge priors and improved target representations. Results show that the proposed approach improves performance and robustness. Findings also indicate potential further enhancements such as a multi-mode tracker with global and local tracking based on a combination of both papers.

Table of Contents

List of Figures

List of Tables

List of Acronyms

DATMO .....Detection and Tracking Moving Objects

DR .....Detection Rate

EM .....Expectation Maximization

FAR .....False Alarm Rate

FAST .....Features from Accelerated Segment Test

FIFO .....First-In-First-Out

FJ .....Figueiredo and Jain

FN .....False Negative

FP .....False Positive

fps .....frames per second

FSM .....Finite State Machine

GLCM .....Grey Level Co-occurrence Matrix

GMM .....Gaussian Mixture Model

HoG .....Histogram of Oriented Gradients

HSV ….....Hue, Saturation, Value

LAB ……...Lightness and a and b

ML .....Maximum Likelihood

PDF ....Probability Density Functions

RGB ....Red, Green, Blue

ROI ....Regions of Interest

SIFT .....Scale Invariant Feature Transform

SVM .....State Vector Machine

TP .....True Positive

TR . .....Tracking Rate

UAV ....Unmanned Aerial Vehicle

Part I

Introduction

1 Introduction

Aerial visual surveillance studies commonly apply, and continue to develop, the fundamental processing steps of detection, classification and tracking. Previous works have shown that detection, classification and tracking of objects are necessary steps in numerous applications [1-5]. These steps have been applied to fixed [6-9] and mobile [1, 10-12] camera platforms for both image and video analysis. More specifically, unmanned aerial camera platforms have the advantage of broader surveillance scope and higher mobility. However, studies have identified various disruptive factors emanating from such data streams, for example; moving background [13], unrestricted pose variation [2], illumination [4], and low contrast between objects and background [12]. Despite these challenges, the growing volumes of data creates a need for automated interpretation tools that reduce human-operator workload and human error. Visual surveillance assists in military and civil applications such as; law enforcement, situational awareness, search and rescue, traffic monitoring and crowd surveillance [2, 4, 12, 13]. Several publications identify and address challenges in the use of unmanned aerial vehicle (UAV) surveillance [14-17], and this work aims to contribute new ideas in addressing those challenges.

The topics of detection and tracking of objects were researched and two journal papers submitted. The principal contribution in both works is the specification and adoption of Gaussian Mixture Model (GMM) based classifiers on commonly used feature spaces. The first paper focuses on detection of ground based objects from UAV video streams using GMM supervised classifiers. While the second paper focuses on the tracking of detected ground based objects from UAV video streams using GMM online classifiers. The GMM has gained recognition due to its ability to represent some classes of real-world data in an efficient and accurate manner [18]. They are capable of representing arbitrary univariate and multivariate distributions in a closed-form representation as a convex combination of Gaussian distributions. Furthermore, they may be applied to any probability distribution over any feature space. The submitted papers employ dimensionality reduction and classification of object probability distributions over various feature spaces, and shows how this forms a sound basis for detection and tracking in UAV video streams.

1.1 Object Detection

An efficient object detector has to accurately determine the location, extent and shape of the objects of interest, despite the challenges faced by aerial platforms [19]. In the past, numerous published works addressed the challenges associated with detection and classification [20-23]. There are various approaches to the problem. However, each approach only addresses a part of the entire problem, therefore a common trend with later works is to combine different approaches. Some of these techniques are discussed next.

In video sequences, information associated with consecutive images can be extracted with motion-based techniques, namely; frame differencing, background subtraction and optical flow. Frame differencing is a simple, fast calculation used to obtain an outline of the moving object by calculating the pixel-wise difference between two consecutive images [8, 24]. However these methods cannot extract all the relevant motion pixels during periodic movements in background, rapid motion, and prompt illumination variations [25]. Therefore it is generally used as a pre-processing step as in [21]. Background subtraction accumulates information about the background scene to produce a background model [26]. The models are compared with the frames to identify moving regions. The methods are categorized as; parametric (frame averaging, single and multiple Gaussian, median filter), non-parametric (kernel density estimation, codebook model) and predictive techniques (Kalman filter background modelling, Eigenbackground) [8, 27]. These are applied to sequences from fixed platforms and require additional advanced methods for moving platforms.

Optical flow methods are better suited for aerial platforms, and they are less susceptible to occlusion, illumination variation and complex or noisy backgrounds [8]. In optical flow, objects are characterised with flow vectors to segment and detect the moving regions overtime. The flow vectors represent the velocity and direction of each pixel or sub-pixel [28]. Mobile platforms have the additional problem of two types of motion in the video, namely, camera motion and object motion. Several works have overcome the problem by estimating the camera's motion. Most commonly used methods are homography and the Lucas–Kanade method [8]. In [1, 16, 21, 22] various features are extracted and homography is applied to track the features between frames, thus a motion estimate is formed. The Lucas–Kanade method is similar in methodology and is part of the proposed solutions described in [1 21 29]. Rodríguez-Canosa [1] further developed these methods to detect and track dynamic moving objects by filtering the difference between artificial and real optical flow using homography and Lucas–Kanade methods respectively. Other works utilise geometric features with optical flow, namely; Hasan [30] uses geometric constraints of the ground plane, Maier [29] uses epipolar geometry

and Cheraghi [11] uses projective geometry. While some geometric methods require metadata such as, the position, altitude and origination of the camera as in [1, 30], which is not always readily available. Motion-based techniques are well suited in segmenting moving objects and eliminating background elements, resulting in fewer false detections. However, most techniques require iterative calculations which increases computational complexity. Furthermore these techniques cannot detect objects if they are stationary [31].

Appearance-based and knowledge-based methods are able to detect moving and stationary objects, and in most cases have less computational complexity than motion-based techniques. The challenge with these detectors is to obtain sufficient information pertaining to the objects of interest whilst minimising the number of false positives detected. To overcome this problem several works employ motion methods as an initial step to exclude background elements. In [30] homography is used to detect moving regions and then identify objects of interest using appearance based pre-trained classifiers. While in [21] the Lucas–Kanade method is used with image registration for background subtraction. Thereafter, binary image classification with blobs is applied for foreground detection. However both works are able to detect only moving objects.

Knowledge based methods are rule-based approaches that encode prior information that describes the object of interest, and can detect non-moving objects. These methods employ a verification step that sufficiently reduces the number of false positive detections by rejecting them. [32] developed a solution for detecting people that builds prior knowledge from the person's motion and appearance. The knowledge is used to automatically select feature sets, training data scales and scales used for detection. These elements are used to construct a classifier with AdaBoost classification [22]. [33] uses knowledge-based priors to describe specific constraints for vehicle detection. Their proposed solution has two main steps, the first step labels the contents of each frame as vehicle, road and background, while the last step filters false positives with knowledge-based spatial reasoning. Knowledge based methods are beneficial for detection applications, however, they require extensive training of classifiers and complex classification leads to higher computational cost.

Appearance based methods use visual information like colour, texture and shape which are obtained through feature extraction. The features are used to acquire models (or templates) from a set of training images. For detection, the models are used by classifiers with statistical analysis and/or machine learning to find the relevant features that belong to the object of interest. Generally the learned features are in the form of distribution models or discriminant functions. Nizar [34] uses appearance based methods to detect vehicles, motorcycles and people. This study utilises Histogram of

Gradients (HoG) features with a State Vector Machine (SVM) classifier. However, the study is applied to fixed camera imagery and is not sufficient for aerial platforms. A similar method was proposed by [15] to detect vehicles for aerial imagery by using additional features with a SVM classifier. The features extracted consisted of; Shape: FAST (Features from Accelerated Segment Test) with corner detectors, HoG, and Colour: HSV colour feature with the Grey Level Co-occurrence Matrix (GLCM) feature. Instead of using both the shape and colour features with a classifier, [36] initially obtains a high density Harris corner feature set. It then clusters heavily overlapping responses and the final detection of vehicles is achieved with a colour-based binary classifier. The use of multiple feature classes is beneficial in aerial platform applications but can have high computational cost. Therefore [19] proposed the use of only Harris corner features at the cost of more complex classification which is achieved with unsupervised clustering and a cascade of boosted classifiers. In another study the search space was reduced by first applying background colour removal which allowed a larger variety of features to be extracted (Harris corner and canny edge detectors) [36]. Since colour features were initially obtained for background removal, it is also used for classification with Dynamic Bayesian Networks. Thereafter, k-means is used to cluster each observation whereas in the training phase, the conditional probability tables of the Bayesian Network model are obtained via the Expectation Maximization (EM) algorithm. An alternative is to use Gaussian Mixture Models (GMM) for classification as these are capable of representing real world data in an efficient and accurate way, thus fewer feature classes are required [38]. Since classification is a challenge with both knowledge and appearance based methods, it is worth investigating the use of GMM for classification.

1.2 Object Tracking

The aim of object tracking is to locate the position of an object over time from a video stream and to associate the position of the object in consecutive video frames. In multiple object tracking [5, 10], the association of each object is crucial, whereas for selected object tracking [39] it is important to differentiate the selected object from other objects in the scene. A key initial step is to first isolate the objects of interest, commonly used approaches are; background subtraction [21], motion detection [1], segmentation [10] and foreground detection [40]. Object detection approaches and their benefits in the context of tracking are discussed in Section 1.1.

Since tracking is related to the motion of objects, several works explore motion-based techniques. [9] uses a Lucas-Kanade tracker to track cyclists. A two-frame differential method for motion estimation via optical flow is used to minimise the estimation error between subsequent frames. This method, however, it is better suited for stationary camera platforms because camera motion can complicate

optical flows. For aerial platforms, [41] uses the SNIFF object tracking algorithm to track vehicles and people. The algorithm was developed at Sarnoff Corporation for real-time application and is based on robust change detection and optical flow based linkage. Whereas [1] developed a real time detection and tracking method for moving objects (DATMO). They calculate the difference between artificial and real optical flow using homography and Lucas–Kanade methods respectively; and thereafter filter and group the dynamic object motion vectors. However, if the object becomes stationary, the motion-based methods then assumes that the object is a background element and stops tracking.

Region, Contour and Feature based techniques are able to continue tracking even after the object becomes stationary. These methods, including some motion based techniques, therefore require static tracking algorithms such as classification to operate. Popular tracking algorithms are Kalman and Particle filters. The Kalman filter is used to estimate the object position in the next frame by using the previously estimated states and current measurements to recursively estimate the next state [8]. While the Particle filter sequentially estimates the latent state variables of a dynamic system based on a sequence of observations using Monte Carlo sampling techniques [8].

A region based tracker was developed by [16] using oriented bounding boxes and a Kalman filter. They assign a region to a specific track if a minimum threshold for the bounding box intersection area of region and Kalman prediction is exceeded. This method was applied to detection, segmentation and tracking of moving objects from UAVs. Whereas, Cao [13] proposed a feature based tracker with a Particle filter to track vehicles from aerial imagery. Cao [13] first estimates the camera's motion for the filter and for each particle, using colour histogram and Hu moments. Another feature based method demonstrated by [15], uses SIFT features and classification instead of tracking algorithms for vehicle tracking. However, a drawback of this method is that a forward-backward tracking algorithm had to be developed to feedback information to the classifier, in order to generate more training samples. Subsequently, it is evident that improvements for classification in this area are required.

1.3 Object Classification

The discussion of object detection and tracking highlights the importance of classification and reveals challenges associated with it. A classifier has to distinguish different classes of object, and this poses a significant challenge when one considers the separability and variability of real data. Nevertheless classifiers are successfully utilised in several different ways. In some instances classification is used as part of the process to eliminate false positive detections [20, 36]. While in other cases objects are

classified into different categories for detection and/or tracking [23, 30, 34, 42], or used for object recognition [43, 44].

Statistical classifiers generally utilise two types of learning methods, namely, generative and discriminative learning. A generative model learns the joint probability distribution p(x,y) whereas a discriminative model learns the conditional probability distribution p(y|x). While the distribution p(x,y) from generative models are used to generate likely pairs, p(y|x) distribution from discriminative models is the natural distribution for classifying a given example x into a class, y. Both classification models are useful for tracking and detection, however discriminative classifiers generally outperform generative models in classification tasks in terms of computation cost and handling missing data [45].

Binary classifiers are the simplest form of discriminative classification, which is restricted to two possible classes. Gleason [35] showed that a binary classifier can be used to detect vehicles from aerial imagery. The classifier uses heavily clustered corner features as input data and colour-based properties to further refine the models. However, binary classification can only describe linear decision boundaries and is overconfident, resulting in additional false positives. Furthermore, binary classification is inefficient as it is prone to over-fitting in high dimensions. To overcome these problems other classifiers have been developed. For non-linearity, State Vector Machine (SVM) is used while for overconfidence, Bayesian classifiers are utilised. Classification trees have been incorporated with boosting to further increase computational speed. Interestingly, Gaussian process classification assists in overcoming both non-linearity and overconfidence problems [46].

Previously, Nizar [34] and Reilly [23] utilised a non-probabilistic approach, SVM, in its original form, by using extracted features for the classifiers. The SVM with a convex objective function guarantees convergence. While Nizar [34] proposes multi-object tracking and detection for transport surveillance with HoG features and Reilly [23] performed shadow detection with blob and wavelet features. Similar to the binary classifier, SVM also executes only on two classes. However another non-probabilistic approach, AdaBoost, finds the best feature sets and constructs a cascade of classifiers to extend to multiple classes. AdaBoost can be used for different views (front, back, left and right view) of detection and tracking, and improves performance [9]. However, Viola and Jones [32] showed increased performance with active learning SVM for image retrieval. The disadvantage of non-probabilistic approaches is that they do not assign certainty to its predictions.

On the other hand, Probabilistic approaches, such as Bayesian, Gaussian and Classification trees assign certainties. Cheng [36] proposed a method for detecting vehicles from aerial imagery using Bayesian classification and multiple feature sets, namely, edge detection, corner detection, colour

transform and colour classification. The classifier successfully incorporated all the feature sets, regardless of the various forms and representations. Bayesian classifiers interpolate real world models with priors, therefore are more accurate. However specifying the priors is a challenge for complex models and can result in high computational cost. Random forest classifiers have tree-structured classification, where the processing for each data example is different and becomes steadily more specific. This is useful for multi-class problems [46]. Sedai [47] and Yu [48] use a combination of shape detectors and descriptors with random forest classifiers. Yu [48] applies to wide area remote sensing where multiple classes exist. This is well suited to the classifier. While Sedai [47] uses the classifier for complex multiscale detectors and descriptors for MRI images. Although tree classifiers significantly reduce computational cost, they tend to overfit data, thus it is not always efficient [46]. Gaussian process classification accurately represents and fits data in an efficient way [46]. GMM classification may meet these requirements.

1.4 Gaussian Mixture Model Classification

The GMM has gained recognition due to its ability to represent some classes of real-world data in an efficient and accurate manner [18]. They are capable of representing arbitrary univariate and multivariate distributions in a closed-form representation as a convex combination of Gaussian distributions. The GMM has been beneficial to numerous applications including other research areas, some examples include; emotion recognition [49], probabilistic trajectory prediction [50], spectral unmixing for multi-spectral data processing [51] and data classification in high energy physics [52]. It is further utilised for image processing of medical data [53-57]. In other image processing areas, including the current research space, GMMs are used to aid the tracking and detection process [24, 58-62]. The use of GMMs extending into multi-disciplines is an indication of its ability to adapt and efficiently represent data.

A common approach with GMMs for detection, tracking and classification is background subtraction due to the ability to handle complex background scenes [63-67]. However, GMMs cannot properly model noisy or nonstationary backgrounds and requires additional methods for optimisation. An alternative approach to GMM background subtraction is to use GMMs for image classification and segmentation instead. Permuter [68, 69] has used this method by applying GMMs on colour and texture features. The classification with GMMs are achieved through Expectation Maximization (EM) and Maximum Likelihood algorithm. However, both works require careful initialisation of GMM parameters and optimal feature sets. To address this challenge, Tao [70] applied Figueiredo and Jain (FJ) algorithm instead of EM, which does not require initialisation of parameters. They developed an

optimized GMM classifier with FJ and SVM, applied to remote sensing images in urban areas. The methods from Permuter [68] and Tao [70] can be merged and adapted for detection from aerial videos. The application of FJ algorithm by Tao [70] can improve classification, while the methodology from Permuter [68] can simplify the classification process for detection and tracking from aerial platforms.

In the context of detection from aerial platforms, a common trend is either to add several different feature sets or to use complex classification methods. Both Chen [15] and Gleason [35] utilise simple classification methods with multiple features sets, such as, colour, FAST (Features from Accelerated Segment Test), Harris corner detectors and HoG. The use of multiple types of features are beneficial in aerial platforms but can have high computational cost. To overcome this problem some works use fewer features with advanced classifiers. An example of this is from [19], who proposed the use of only Harris corner features with unsupervised clustering and a cascade of boosted classifiers. The unsupervised clustering through k-means is required to assist the classifiers by grouping data, at the cost of additional computation. Another method by Cheng [36] also uses clustering by k-means to aid classification with Dynamic Bayesian Networks. Further challenges arise in their training, where the conditional probability tables of the Bayesian Network model are required and obtained via the Expectation Maximization (EM) algorithm. An efficient alternative to clustering data through k-means is to use GMM, which will find data parameters while clustering the data, furthermore the training phase in [36] can be simplified with GMM classification. In addition, simpler low-dimensional feature spaces can be used and the GMM is capable of representing distributions within these spaces as a parametric probabilistic model [38].

In the context of tracking from aerial platforms, there are proposed methods that treat the tracking problem as a classification task [71-75]. Despite the success they have demonstrated, numerous issues remain to be addressed. Firstly, these methods need a set of labelled training instances (samples) to determine the decision boundary for separating the target object. Secondly, in most cases, there is not sufficient instances for unsupervised learning. GMM classification is an efficient unsupervised alternative that is capable of labelling instances and requires fewer instances. Since classification is a challenge for detection and tracking from aerial platforms, it is worth investigating the use of GMM classification.

1.5 Feature Extraction

The feature extraction process is a key factor for the GMM classification, as probability distribution functions (PDF) are generated from the feature sets. These functions represent the attributes that

identify and distinguish objects from the scene. The GMM classifiers create probabilistic models for the distributions thus simplifying the comparison process. Different types of feature sets are required for detection and tracking, namely, shape-based for the detection while colour-based features are used for tracking. Other applicable features in this area of research are texture-based and motion-based features.

1.5.1 Shape Features

Shape features are generally used in two different ways. Firstly, to identify the precise pixels that belong to an object from the current scene. Secondly, to extract information about the identity or other characteristics such as the position of the object [46]. The information is used for various applications, consisting of; representation [46], segmentation [10], recognition [44], detection [48] and tracking [41]. For shape feature extraction, detectors and descriptors can be utilised and can function as a collective. Detectors assist in determining the location of prominent key points which belong to objects of interest which are beneficial in reducing search space [76]. Descriptors are used to determine which key points come from the corresponding locations in different image regions. It is capable of representing a cluster of key points as a single point descriptor which is useful for data representation for classification [76].

The Canny Edge Detector is widely used and highly cited and is commonly used with machine learning methods to identify object boundaries in images. In [36] the detector is used as one of the feature sets for the classification process in aerial video streams. However, since the illumination varies within a scene, different thresholds for the edge detector are required according. To overcome the problem, Cheng [36] applies the moment-preserving thresholding method, which adaptively selects the lower and higher threshold values. In the same works, Harris Corner detector is utilised without thresholding, implying that it is more robust to changes in aerial applications. The corner detector can also be used for motion detection, by tracking the features with Lucas–Kanade optical flow method [12], [77]. Furthermore, it is possible to use Harris as the only feature set for classification but in fixed images [19]. Yu [48] uses histogram-based shape descriptor to represent Harris corner features for wide area remote sensing application. Therefore feature descriptors are useful in representing detectors and can assist in the application for GMM classification.

Histogram of oriented Gradients (HoG) and Scale Invariant Feature Transform (SIFT) are the most commonly used descriptors for aerial video stream applications. SIFT has been used to detect corresponding Harris corners for camera estimation in wide area motion imagery [16]. A combination of both SIFT and HoG is proposed for classification of vehicles and surgical fixed images [77].

However, for aerial platforms these descriptors require additional methods and/or advance classification techniques [1, 9, 15, 78], have demonstrated various combinations of HoG and SIFT with different detectors and provide a comparison of different feature detectors and descriptors for vehicle classification from UAV's.

1.5.2 Colour Features

Colour-based methods are widely applied in image processing. A common use is for prior background subtraction, however colour-based methods can be used as a final refinement step for detection in aerial imagery [35]. Therefore, eliminating the background areas characterised by a monochromatic colour distribution. Others use colour and simple entropy to form a fingerprint for object detection from UAVs [10]. A further application is that of colour segmentation of breast infrared images using GMM [79].

Colour histogram is a representation of the colour distribution of an image, and the number of pixels that have colours in each of a fixed list of colour ranges. The common method of colour histograms is to represent the colour space as a three-dimensional space of RGB (red, blue, green). This is not optimal as it has a large separation of data, which requires more storage space. The method of colour quantization reduces the amount of data storage required. Colour quantization divides the colour space into a certain numbers of small intervals; each interval is called a bin. The number of pixels in each of the bins forms a one-dimensional colour histogram that is well suited to be used for GMM. Colour histogram is a commonly used feature set as it is relatively constant within a video sequence and the amount of information conveyed. However changes in perspective, illumination and scale causes variations in the colour. Kviatkovsky [80] explores aspects of colour structure that are invariant to illumination for person re-identification. In addition, some works consider colour spaces other than RGB, namely, and LAB [1, 42] and HSV (hue, saturation, value) [15]. RGB does not require transformations to perceive the colour and has a lower computational cost, however it is not always useful for object specification and difficult to determine specific colour in RGB model [81]. LAB is perceived as uniform but suffers from unintuitive. While for HSV, the hue and saturation components is the way humans perceive colour and well suited for image processing. Additionally, the hue component can be used for segmentation with better speed since only one component needs to be processed. However, undefined achromatic hue points are sensitive to value deviations of RGB and instability of hue [81].

2 Motivation and Research Objective

In the area of visual surveillance, detection, classification and tracking of objects are beneficial in numerous applications, as shown in previous works [1-5]. It has been applied to fixed [6-9] and mobile [1, 10-12] camera platforms for both image and video analysis. Furthermore aerial platforms have an additional advantage of larger surveillance scope and higher mobility. However, existing works have identified various factors that contribute to noise, namely; change in viewpoint, parallax errors, and low contrast between objects and background. Despite the challenges, the ever growing volume of data creates a need for automated interpretation tools that reduce human error and human workload. Visual surveillance assists in military and civil applications such as law enforcement, situational awareness, search and rescue, traffic monitoring and crowd surveillance [2, 4, 12, 13]. Furthermore, Unmanned Aerial Vehicle (UAV) surveillance platforms may be utilised. These are increasingly cost-effective, safer and, quick to set up and deploy. Several publications identify and address challenges in the use of UAVs for surveillance of multiple object types [14-17]. These publications have identified various disruptive factors emanating from such data streams, for example; moving background [13], unrestricted pose variation [2], illumination [4], and target occlusion [12]. To overcome these problems, classification has become a key factor for detection and tracking. For detection from aerial platforms, a common trend is either to add several different feature sets or to use complex classification methods. The use of multiple feature sets can significantly increase computational cost, and complex classification methods often require extensive expertise to configure and deploy. Furthermore, complex unsupervised classification methods are also used in the context of tracking. These methods require a set of labelled training samples and in most cases, there are insufficient instances for successful unsupervised learning. GMM classification simplifies the process. A parametric model of the data is created, and classification is performed on model parameters instead of the high-dimensional data. Furthermore GMM classification does not require extensive training sets and labelled samples. Therefore the objective is to investigate whether the specification and adoption of GMM based classifiers on commonly used feature spaces is beneficial in alleviating the challenges associated with detection and tracking in UAV video streams.

3 Contributions of Included Papers

The contributions of this dissertation are presented in two journal papers (Paper A and Paper B) which are included in Section II. Paper A focuses on the detection of ground based objects using GMM supervised classifiers, with UAV video streams. While Paper B focuses on the tracking of detected ground based objects using GMM online classifiers for UAV video streams.

3.1 Paper A

T. Pillay, B. Naidoo, "Gaussian Mixture Model classifiers for detection in UAV video streams", [Submitted for review to International Journal of Remote Sensing], 2017

Paper A: The paper aims to simplify the classification process in a reduced feature space. The objectives are demonstrated with a vehicle detector using a single feature space, namely Histogram of Oriented Gradients (HoG) with Gaussian Mixture Model (GMM) classifiers. A low-dimensionality information-preserving feature space is developed and reduced to a parametric mixture model, further decreasing complexity. The use of a likelihood function simplifies the classification process as the function provides likelihood estimate values for direct comparison. The proposed solution is tested on standard datasets, and performs well in comparison to related works.

3.2 Paper B

T. Pillay, B. Naidoo, "Gaussian Mixture Model classifiers for tracking in UAV video streams", [Submitted for review to International Journal of Remote Sensing], 2017

Paper B: This paper aims to simplify the classification process with a minimised set of unlabelled instances to reduce the problems experienced by complex classification. The objectives are demonstrated with a vehicle tracker using colour histograms, with Gaussian Mixture Model (GMM) classifiers and a Kalman filter. GMM classification is used to differentiate the tracked object from other elements in the scene using a likelihood function; and the Kalman filter provides a step-ahead estimate of the object location, thus reducing the search space. The GMM classification model is constantly updated with a limited set of instances obtained from recent frames. This allows the model to adapt to the changes in the appearance of the tracked object over a localised time-frame. GMM classification has resulted in a simplified classification process which is tested on standard datasets, and performs well in comparison to related works.

4 References

1. Rodríguez-Canosa, G.R., et al., *A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera.* Remote Sensing, 2012. **4**(4): p. 1090-1111.

2. Kimura, M., et al. *Automatic extraction of moving objects from UAV-borne monocular images using multi-view geometric constraints*. in *IMAV 2014: International Micro Air Vehicle Conference and Competition 2014, Delft, The Netherlands, August 12-15, 2014*. 2014. Delft University of Technology.

3. Kumar, R., et al., *Aerial video surveillance and exploitation.* Proceedings of the IEEE, 2001. **89**(10): p. 1518-1539.

4. Pokheriya, M. and D. Pradhan. *Object detection and tracking based on silhouette based trained shape model with Kalman filter*. in *Recent Advances and Innovations in Engineering (ICRAIE), 2014*. 2014. IEEE.

5. Yang, W., et al. *Real-time detection algorithm of moving ground targets based on Gaussian mixture model*. in *SPIE Remote Sensing*. 2014. International Society for Optics and Photonics.

6. Priya, K.R. and R. Ramachandiran, *Vehicle detection and tracking methods.* 2016.

7. Dave, S.A., D.M. Nagmode, and A. Jahagirdar, *Statistical Survey on Object Detection and tracking Methodologies‖.* International Journal of Scientific & Engineering Research, 2013. **4**(3).

8. Abdulrahim, K. and R.A. Salam, *Traffic Surveillance: A Review of Vision Based Vehicle Detection, Recognition and Tracking.* International Journal of Applied Engineering Research, 2016. **11**(1): p. 713-726.

9. Cho, H., P.E. Rybski, and W. Zhang. *Vision-based bicyclist detection and tracking for intelligent vehicles*. in *Intelligent Vehicles Symposium (IV), 2010 IEEE*. 2010. IEEE.

10. Mundhenk, T.N., et al. *Detection of unknown targets from aerial camera and extraction of simple object fingerprints for the purpose of target reacquisition*. in *IS&T/SPIE Electronic Imaging*. 2012. International Society for Optics and Photonics.

11. Cheraghi, S.A. and U.U. Sheikh. *Moving object detection using image registration for a moving camera platform*. in *Control System, Computing and Engineering (ICCSCE), 2012 IEEE International Conference on*. 2012. IEEE.

12. Siam, M. and M. ElHelw. *Robust autonomous visual detection and tracking of moving targets in UAV imagery*. in *Signal Processing (ICSP), 2012 IEEE 11th International Conference on*. 2012. IEEE.

13. Cao, X., et al., *Vehicle detection and motion analysis in low-altitude airborne video under urban environment.* IEEE Transactions on Circuits and Systems for Video Technology, 2011. **21**(10): p. 1522-1533.

14. Ma, Y., et al., *Pedestrian Detection and Tracking from Low-Resolution Unmanned Aerial Vehicle Thermal Imagery.* Sensors, 2016. **16**(4): p. 446.

15. Chen, X. and Q. Meng. *Robust vehicle tracking and detection from UAVs*. in *Soft Computing and Pattern Recognition (SoCPaR), 2015 7th International Conference of*. 2015. IEEE.

16. Teutsch, M. and W. Krüger. *Detection, segmentation, and tracking of moving objects in UAV videos*. in *Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on*. 2012. IEEE.

17. Jeon, B., et al. *Mode changing tracker for ground target tracking on aerial images from unmanned aerial vehicles (ICCAS 2013)*. in *Control, Automation and Systems (ICCAS), 2013 13th International Conference on*. 2013. IEEE.

18. Reynolds, D.A., T.F. Quatieri, and R.B. Dunn, *Speaker verification using adapted Gaussian mixture models.* Digital signal processing, 2000. **10**(1): p. 19-41.

19. Kapsalas, P., et al. *Regions of interest for accurate object detection*. in *2008 International Workshop on Content-Based Multimedia Indexing*. 2008. IEEE.

20. Xu, Q., et al. *Air-ground vehicle detection using local feature learning and saliency region detection*. in *Intelligent Control and Automation (WCICA), 2012 10th World Congress on*. 2012. IEEE.

21. Mao, H., et al., *Automatic detection and tracking of multiple interacting targets from a moving platform.* Optical Engineering, 2014. **53**(1): p. 013102-013102.

22. Teutsch, M. and W. Kruger. *Robust and Fast Detection of Moving Vehicles in Aerial Videos using Sliding Windows*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2015.

23. Reilly, V., B. Solmaz, and M. Shah, *Shadow casting out of plane (SCOOP) candidates for human and vehicle detection in aerial imagery.* International journal of computer vision, 2013. **101**(2): p. 350-366.

24. Li, N., et al., *Object Tracking with Multiple Instance Learning and Gaussian Mixture Model.* Journal of Information and Computational Science, 2015. **12**(11): p. 4465-4477.

25. Buch, N., et al., *A review of computer vision techniques for the analysis of urban traffic*, IEEE Transactions on Intelligent Transportation Systems, 2011. **12**(3): p. 920-939

26. Manikandan, R., et al., *Video object extraction by using background subtraction techniques for sports applications*, Digital Image Processing, 2013. **5(**9): p. 435-440

27. Wan, Q., et al., *Background subtraction based on adaptive non-parametric model*, in *World Congress on Intelligent Control and Automation (WCICA)*, 2008 *7th World Congress on*. 2008. IEEE.

28. Liu, Y., et al., *Optical flow based urban road vehicle tracking*, in *International Conference Computational Intelligence and Security (CIS)*, 2013 . IEEE

29. Maier, J., et al., *Movement Detection Based on Dense Optical Flow for Unmanned Aerial Vehicles*, International Journal of Advanced Robotic Systems, 2013. **10**(146): p. 1-11

30. Hasan, M., *Integrating Geometric, Motion and Appearance Constraints for Robust Tracking in Aerial Videos*, 2013

31. Tian B., et al., *Hierarchical and networked vehicle surveillance in its: A survey*, in *Trans. Intell. Transp. Syst*., 2015 IEEE

32. Viola, P. and Jones, M., 2001. *Rapid object detection using a boosted cascade of simple features*, in *Computer Vision and Pattern Recognition*, 2001 *Proceedings of the IEEE Computer Society Conference.* 2001. IEEE.

33. Guilmart, C., *Context-driven moving object detection in aerial scenes with user input*, in *ICIP*, 2011 IEEE

34. Nizar, T.N., N. Anbarsanti, and A.S. Prihatmanto. *Multi-object tracking and detection system based on feature detection of the intelligent transportation system*. in *System Engineering and Technology (ICSET), 2014 IEEE 4th International Conference on*. 2014. IEEE.

35. Gleason, J., et al. *Vehicle detection from aerial imagery*. in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. 2011. IEEE.

36. Cheng, H.-Y., C.-C. Weng, and Y.-Y. Chen, *Vehicle detection in aerial surveillance using dynamic Bayesian networks.* IEEE transactions on image processing, 2012. **21**(4): p. 2152-2159.

37. Carmi, A., et al., *The Gaussian mixture MCMC particle algorithm for dynamic cluster tracking*. Automatica, 2012. **48**(10), p.2454-2467

38. Kersten, J., *Simultaneous feature selection and Gaussian mixture model estimation for supervised classification problems.* Pattern Recognition, 2014. **47**(8): p. 2582-2595.

39. Zivkovic, Z. and B. Krose. *An EM-like algorithm for color-histogram-based object tracking*. in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. 2004. IEEE.

40. Santosh, D.H. and P.K. Mohan. *Multiple objects tracking using Extended Kalman Filter, GMM and Mean Shift Algorithm-A comparative study*. in *Advanced Communication Control and Computing Technologies (ICACCCT), 2014 International Conference on*. 2014. IEEE.

41. Xiao, J., et al., *Vehicle and person tracking in aerial videos*, in *Multimodal Technologies for Perception of Humans*. 2008, Springer. p. 203-214.

42. Khan, S.M., et al. *3D model based vehicle classification in aerial imagery*. in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. 2010. IEEE.

43. Qi, Y., et al., *Strategy of active learning support vector machine for image retrieval,* IET Computer Vision, 2016. **10**(1), p.87-94.

44. Feng, Y., Wu, X. and Jia, Y., *Multi-group–multi-class domain adaptation for event recognition*. IET Computer Vision, 2016. **10**(1), p.60-66.

45. Guo, Y.K., et al., *Uncommon adrenal masses: CT and MRI features with histopathologic correlation*, European journal of radiology, 2007. **62**(3), p.359-370.

46. Prince, S.J., 2012. Computer vision: models, learning, and inference. Cambridge University Press.

47. Sedai, S., et al., *Right ventricle landmark detection using multiscale HoG and random forest classifier*, in Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on 2015. IEEE.

48. Yu, B., et al., *Constructions detection from unmanned aerial vehicle images using random forest classifier and histogram-based shape descriptor*, Journal of Applied Remote Sensing, **8**(1), pp.083554-083554.

49. Muthusamy, H., K. Polat, and S. Yaacob, *Improved emotion recognition using gaussian mixture model and extreme learning machine in speech and glottal signals.* Mathematical Problems in Engineering, 2015. **2015**.

50. Wiest, J., et al. *Probabilistic trajectory prediction with gaussian mixture models*. in *Intelligent Vehicles Symposium (IV), 2012 IEEE*. 2012. IEEE.

51. Deng, S., et al., *An infinite Gaussian mixture model with its application in hyperspectral unmixing.* Expert Systems with Applications, 2015. **42**(4): p. 1987-1997.

52. Štěpánek, M., J. Franc, and V. Kůs. *Modification of Gaussian mixture models for data classification in high energy physics*. in *Journal of Physics: Conference Series*. 2015. IOP Publishing.

53. Dai, P., et al., *A new approach to segment both main and peripheral retinal vessels based on gray-voting and gaussian mixture model.* PloS one, 2015. **10**(6): p. e0127748.

54. Kermani, S., N. Samadzadehaghdam, and M. EtehadTavakol, *Automatic color segmentation of breast infrared images using a Gaussian mixture model.* Optik-International Journal for Light and Electron Optics, 2015. **126**(21): p. 3288-3294.

55. Yousefi, S., et al., *Unsupervised Gaussian Mixture-Model With Expectation Maximization for Detecting Glaucomatous Progression in Standard Automated Perimetry Visual Fields.* Translational vision science & technology, 2016. **5**(3): p. 2-2.

56. Ragothaman, S., et al. *Unsupervised Segmentation of Cervical Cell Images Using Gaussian Mixture Model*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2016.

57. Xiong, G., C. Feng, and L. Ji, *Dynamical Gaussian mixture model for tracking elliptical living objects.* Pattern Recognition Letters, 2006. **27**(7): p. 838-842.

58. Zhan, X. and B. Ma, *Gaussian mixture model on tensor field for visual tracking.* IEEE Signal Processing Letters, 2012. **19**(11): p. 733-736.

59. Chauhan, A.K. and P. Krishan, *Moving object tracking using gaussian mixture model and optical flow.* International Journal of Advanced Research in Computer Science and Software Engineering, 2013. **3**(4).

60. Kim, J., Z. Lin, and I.S. Kweon, *Rao-Blackwellized particle filtering with Gaussian mixture models for robust visual tracking.* Computer Vision and Image Understanding, 2014. **125**: p. 128-137.

61. Quast, K. and A. Kaup, *Shape adaptive mean shift object tracking using gaussian mixture models*, in *Analysis, Retrieval and Delivery of Multimedia Content*. 2013, Springer. p. 107-122.

62. Santosh, D.H.H., et al., *Tracking Multiple Moving Objects Using Gaussian Mixture Model.* International Journal of Soft Computing and Engineering (IJSCE), 2013. **3**(2).

63. Lee, D.-S., *Effective Gaussian mixture learning for video background subtraction.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005. **27**(5): p. 827-832.

64. Mukherjee, D., Q.J. Wu, and T.M. Nguyen, *Gaussian mixture model with advanced distance measure based on support weights and histogram of gradients for background suppression.* IEEE Transactions on Industrial Informatics, 2014. **10**(2): p. 1086-1096.

65. Zhou, D. and H. Zhang. *Modified GMM background modeling and optical flow for detection of moving objects*. in *2005 IEEE International Conference on Systems, Man and Cybernetics*. 2005. IEEE.

66. Xue, K., et al., *Panoramic Gaussian Mixture Model and large-scale range background substraction method for PTZ camera-based surveillance systems.* Machine vision and applications, 2013. **24**(3): p. 477-492.

67. Fradi, H. and J.-L. Dugelay. *Robust foreground segmentation using improved gaussian mixture model and optical flow*. in *Informatics, Electronics & Vision (ICIEV), 2012 International Conference on*. 2012. IEEE.

68. Permuter, H., J. Francos, and I. Jermyn, *A study of Gaussian mixture models of color and texture features for image classification and segmentation.* Pattern Recognition, 2006. **39**(4): p. 695-706.

69. Permuter, H., J. Francos, and I.H. Jermyn. *Gaussian mixture models of texture and colour for image database retrieval*. in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*. 2003. IEEE.

70. Tao, J., et al., *A study of a Gaussian mixture model for urban land-cover mapping based on VHR remote sensing imagery.* International Journal of Remote Sensing, 2016. **37**(1): p. 1-13.

71. Grabner, H., M. Grabner, and H. Bischof. *Real-time tracking via on-line boosting*. in *BMVC*. 2006.

72. Babenko, B., M.-H. Yang, and S. Belongie, *Robust object tracking with online multiple instance learning.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011. **33**(8): p. 1619-1632.

73. Zhang, K. and H. Song, *Real-time visual tracking via online weighted multiple instance learning.* Pattern Recognition, 2013. **46**(1): p. 397-411.

74. Zhang, K., L. Zhang, and M.-H. Yang. *Real-time compressive tracking*. in *European Conference on Computer Vision*. 2012. Springer.

75. Zhang, K., L. Zhang, and M.-H. Yang, *Fast compressive tracking.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014. **36**(10): p. 2002-2015.

76. Szeliski, R., 2010. Computer vision: algorithms and applications. Springer Science & Business Media.

77. Stavens, D. and S. Thrun. *Unsupervised learning of invariant features using video*. in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. 2010. IEEE.

78. Moranduzzo, T. and F. Melgani. *Comparison of different feature detectors and descriptors for car classification in UAV images*. in *2013 IEEE International Geoscience and Remote Sensing Symposium-IGARSS*. 2013. IEEE.

79. Kermani, S., et al., *Automatic color segmentation of breast infrared images using a Gaussian mixture model*, Optik-International Journal for Light and Electron Optics, 2015. **126**(21), p.3288-3294.

80. Kviatkovsky, I., A. Adam, and E. Rivlin, *Color invariants for person reidentification.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013. **35**(7): p. 1622-1634.

81. Collins, R., X. Zhou, and S.K. Teh. *An open source tracking testbed and evaluation web site*. in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*. 2005.

Part II

Included Papers

Paper A

Gaussian Mixture Model classifiers for detection in UAV video streams

T. Pillay and B. Naidoo

Submitted for review to International Journal of Remote Sensing

Abstract

Unmanned Aerial Vehicle (UAV) visual surveillance is widely applied, and still actively researched with regard to detection and classification. Although object detection has improved significantly, it continues to pose challenges for UAVs, due to moving background, unrestricted pose variation, illumination and low contrast. Past solutions have resorted to multiple feature sets, which results in redundant feature spaces and complex classification. This study aims to simplify the classification process in a reduced feature space. The objectives are demonstrated with a vehicle detector using a single feature space, Histogram of Oriented Gradients (HoG), with Gaussian Mixture Model (GMM) classifiers. GMMs provide a concise parametric representation of the HoG distributions. GMM parameters are computed during a training phase and are categorised at a subsequent classification phase. The training model parameters are compared to candidate parameters using a likelihood function, thus providing classification. Detection is achieved with a simple two-stage GMM classifier, the stages are: (1) initial detection: find regions of interest (ROI) from video frames, (2) final detection: classify ROIs from stage 1 and output detections. A simple feature space is used and this is reduced to a parametric mixture model, further decreasing complexity. The use of the likelihood function simplifies the classification process as the function provides likelihood estimate values for direct comparison. The proposed solution is tested on standard datasets, and performs well in comparison to related works.

1. Introduction

Aerial visual surveillance research is widely applied, and continues to develop the fundamental processing steps of detection and classification. Previous works have shown that detection and classification of objects are necessary steps in numerous applications [1-5]. These steps have been applied to fixed [6-9] and mobile [10-13] camera platforms for both image and video analysis. More specifically, unmanned aerial camera platforms have the advantage of broader surveillance scope and higher mobility. However, studies have identified various disruptive factors emanating from such data streams, for example; moving background [14], unrestricted pose variation [2], illumination [15], and low contrast between objects and background [12]. Despite these challenges, the growing volumes of data creates a need for automated interpretation tools that reduce human-operator workload and human error. Visual surveillance assists in military and civil applications such as; law enforcement, situational awareness, search and rescue, traffic monitoring and crowd surveillance [2, 4, 12, 14]. Several publications identify and address challenges in the use of unmanned aerial vehicle (UAV) surveillance [15-18].

An efficient object detector has to accurately determine the location, extent and shape of the objects of interest, despite the challenges faced by aerial platforms [19]. In the past, numerous published works addressed the challenges associated with detection and classification [20-23]. There are various approaches to the problem. However, each approach only addresses a part of the entire problem, therefore a common trend with later works is to combine different approaches. The most common approaches for vehicle detection are feature based and applies the proposed object detection framework by Viola and Jones [24]. The approaches generally compensate for poor performance by extracting a large number of features and then combining several weak classifiers with a cascade structure to form one strong classifier. While other approaches, such as knowledge based methods, require extensive training of classifiers and complex classification, which leads to higher computational cost.

It would be ideal to use a single feature set with simple classification, similar to [25], who uses Histogram of Oriented Gradients (HoG), with a State Vector Machine (SVM) classifier. This method can detect vehicles, motorcycles and people, implying that HoG can differentiate these objects well. However, the abovementioned method is applied to fixed camera imagery and is not sufficient for aerial platforms. Therefore, a common trend for aerial platforms is either to add several different feature sets or to use complex classification methods. Both Chen [17] and Gleason [26] utilise simple classification methods with multiple features sets, such as, colour, FAST (Features from Accelerated

Segment Test), Harris corner detectors and HoG. The use of multiple types of features are beneficial in aerial platforms but can have high computational cost. To overcome this problem some works use fewer features with advanced classifiers. An example of this is from [19], who proposed the use of only Harris corner features with unsupervised clustering and a cascade of boosted classifiers. The unsupervised clustering through k-means is required to assist the classifiers by grouping data, at the cost of additional computation. Another method by [27] also uses clustering by k-means to aid classification with Dynamic Bayesian Networks. Further challenges arise in their training, where the conditional probability tables of the Bayesian Network model are required and obtained via the Expectation Maximization (EM) algorithm.

An efficient alternative to clustering data through k-means is to use Gaussian Mixture Models (GMM), which will find data parameters while clustering the data, furthermore the training phase in [27] can be simplified with GMM classification. In addition, simpler low-dimensional feature spaces are required and the GMM is capable of representing distributions within these spaces as a parametric probabilistic model [28]. Since classification is a challenge in this area of research, it is worth investigating the use of GMMs for classification.

The GMM has gained recognition due to its ability to represent some classes of real-world data in an efficient and accurate manner [29]. They are capable of representing arbitrary univariate and multivariate distributions in a closed-form representation as a convex combination of Gaussian distributions. The GMM has been beneficial to numerous applications including other research areas, some examples include; emotion recognition [30], probabilistic trajectory prediction [31], spectral unmixing for multi-spectral data processing [32] and data classification in high energy physics [33]. It is further utilised for image processing of medical data [34-38]. In other image processing areas, including the current research space, GMMs are used to aid the tracking and detection process [39-44]. The use of GMMs extending into multi-disciplines is an indication of its ability to adapt and efficiently represent data.

A common approach with GMMs for detection and classification is background subtraction due to the ability to handle complex background scenes [45-49]. However, GMMs cannot properly model noisy or nonstationary backgrounds and requires additional methods for optimisation. A different approach by [50, 51], who applied GMMs on colour and texture features for image classification and segmentation. The classification with GMMs are achieved through Expectation Maximization (EM) and Maximum Likelihood algorithm. However, both works require careful initialisation of GMM parameters and optimal feature sets. To address this challenge, Tao [52] applied Figueiredo and Jain

(FJ) algorithm instead of EM, which does not require initialisation of parameters. They developed an optimized GMM classifier with FJ and SVM, applied to remote sensing images in urban areas. The methods from Permuter [50] and Tao [52] can be merged and adapted for detection from aerial videos. The application of FJ algorithm by Tao [52] can improve classification, while the methodology from Permuter [50] can simplify the classification process.

This paper focuses on detection of ground-based vehicles from UAV video streams using GMM classifiers. The specification and adoption of GMM based classifiers on commonly used feature spaces forms the principal contribution of this work. Variations of HoG descriptors are used to extract features from the frames of the UAV video streams. The distributions of these features are computed and represented by a parameterised GMM. Thereafter, classification is performed on the parameters and not on the data, thus simplifying the process. A two stage cascade of GMM classifiers are utilised to first detect potential vehicles, and secondly to validate the detections, thus reducing false positives. The proposed work is directly compared to related works, as testing is performed on the benchmark VIVID dataset [53].

The paper continues onto Section 2, which reviews the various parts of GMM and presents the proposed solution. Section 3, describes the test evaluation and shows the experimental results with comparisons with related works, while Section 4 presents the conclusions.

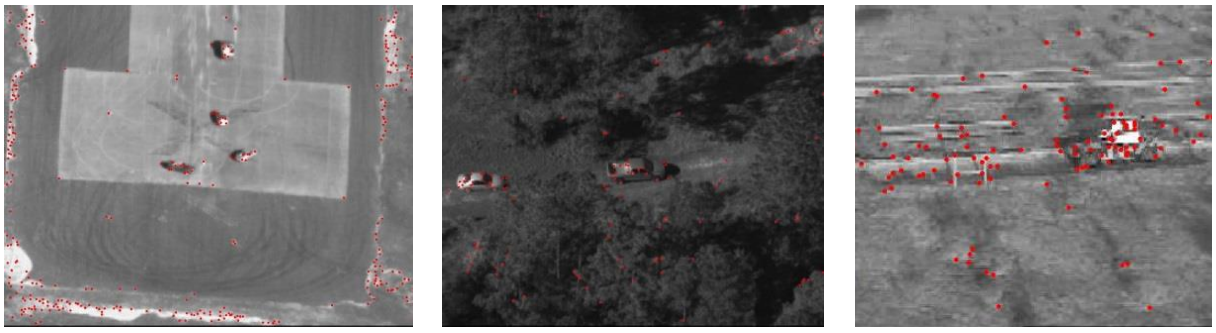
2. Gaussian Mixture Model Classification

The proposed method requires various elements, namely, feature extraction, GMM parameterisation, parameter classification and finally, detection functionality. GMM parameters are computed and stored during the training phase; whereas in the detection phase, current parameters are compared to trained parameters in order to classify the image regions as vehicle or background. This method is applied to both stages of the cascaded classifier to produce the final vehicle detection.

2.1. Feature Extraction

The feature extraction process generates data distributions, $X_{(N,D)}$ in the shape feature space, where the number of samples is N and the dimensionality of the space is D. These distributions are modelled with GMMs to create a parametric probabilistic model for the training and comparative process of classification. Since GMMs represent probability density functions (PDF) as a weighted sum of Gaussian density functions, the input data must be PDF as well. Histogram of Oriented Gradients (HoG) descriptors represents a shape feature space that procedures PDF. HoG descriptors contain the

attributes that identify and distinguish objects from the scene. The HoG-edge feature space represents the shape of vehicles, while HoG-corner feature space is used to detect potential vehicle regions. The HoG-edge is capable of obtaining features that differentiate between vehicle and background, whereas HoG-corner detects all corners in an image, including both vehicles and background. Furthermore, some background elements contain similar distributions as vehicles. However, since fewer data points are generated, this is ideal for initial detection, where the entire frame is considered. The more detailed HoG-edge considers small regions of the frame and is ideal for validation. Details of the HoG descriptor can be found in [25, 54], while, HoG-corner in [55]. Fig. A.1 (a) – (c), are visual representations of the HoG-corner applied to entire frames from the VIVID dataset. Dense corner features are extracted in order to reduce false negatives. However, this can increase the false positive rate. Therefore, classification is used to filter out false positives. Fig. A.2 shows illustrations of HoG-edge features extracted from regions of interest (ROI). HoG visualisation for vehicles is shown in Fig. A.2 (a)-(b) is a representation of a vehicle, while in Fig A.2 (c)-(d) for background elements. The illustrations show the difference in distribution between the two items however, some overlap does exist.



(a) (b) (c)

Fig. A.1 *Visual representation of HoG-Harris corner (a)-(c) from VIVID dataset frames*

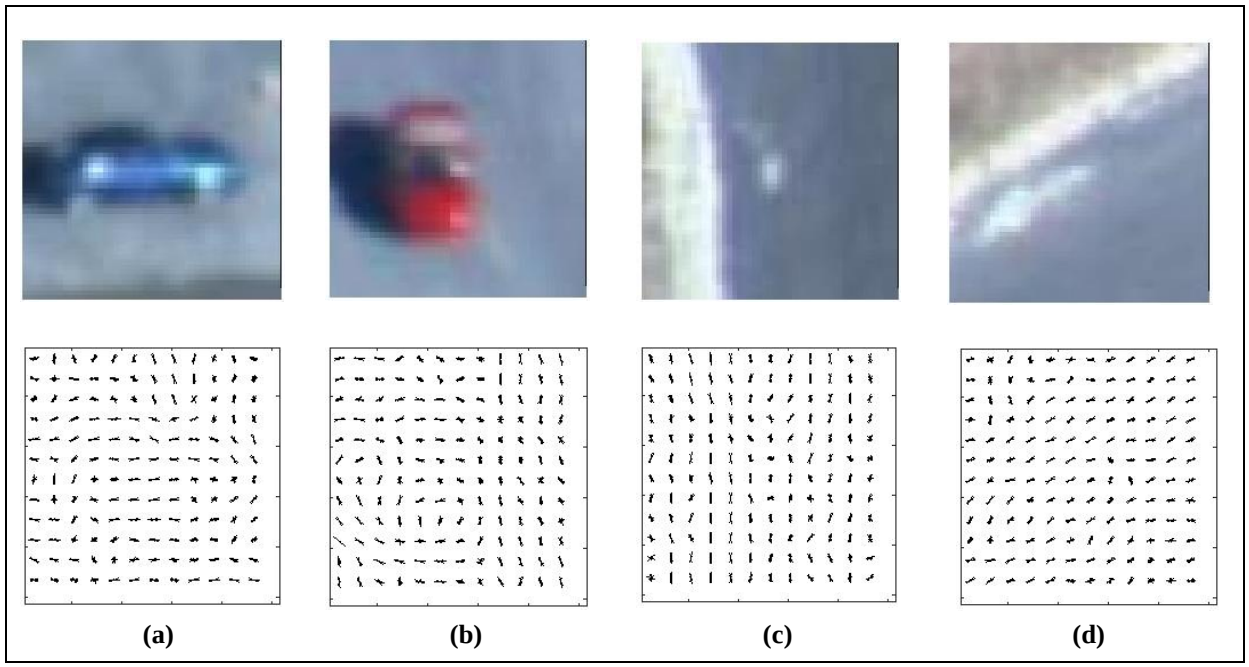


Fig. A.2 *Visual representation of HoG-edge (a)-(b) – HoG of vehicles, (c)-(d) – HoG of background*

2.2. Gaussian Mixture Models

The general assumption is that, when simple natural data is represented as a PDF, the PDF is usually Gaussian in nature. However, complex real world data distributions often approximate linear combinations of Gaussian distributions. Thus, the motivation for using GMMs is to represent $X_{(N,D)}$ as multiple Gaussians in an efficient and accurate way. Assume that $X_{(N,D)} = X\{x_1, \dots, x_N\}$ is a set of N independent and identically distributed samples in D dimensions; and its PDF approximates a multivariate Gaussian distribution, then the PDF of sample x may be represented as [56]:

$$p(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\sigma^2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{1}$$

where, μ and σ^2 are the mean and variance parameters respectively. If given sufficient training samples, GMMs are capable of representing the PDF. The arbitrary set $X_{(N,D)}$, all $p(x|\mu, \sigma^2)$, can be approximated by a weighted sum of K Gaussian density functions, as illustrated below [56]:

$$p(X|\mu, \sigma^2) = \sum_{k=1}^{K} \alpha_k p(X|\mu_k, \sigma_k^2) \tag{2}$$

where, $\alpha_k (k = 1,2, \dots, K)$ are the prior probabilities (mixture weights) of the components k. The aim of GMMs is to estimate the parameters μ, σ^2 and α, which is achieved by applying the Maximum Likelihood (ML) estimation to equation (2), [56]:

$$\log p(X|\mu, \sigma^2) = \log \prod_{n=1}^{N} p(x_n|\mu, \sigma^2) = \sum_{n=1}^{N} \log \sum_{k=1}^{K} \alpha_k p(x_n|\mu_k, \sigma_k^2) \tag{3}$$

2.3. Expectation Maximisation Algorithm

The ML estimation, equation (3) does not converge to a close form solution, which is not ideal for computational purposes. A commonly used solution to this is the Expectation Maximisation (EM) algorithm, an iterative process that finds the local maxima of $\log p(x|\mu, \sigma^2)$. The algorithm maximises the likelihood function with respect to the parameters. The conditions that must be satisfied at a maximum of the likelihood function are found by setting the derivatives with respect to μ and σ^2 in equation (3) to zero. The required equations for the expectation and maximization steps are derived by multiplying the solution by σ_k^{-2} [57]:

$$\gamma(x_k) = \frac{\alpha_k p(x_n|\mu_k, \sigma_k^2)}{\sum_{j=1}^{K} \alpha_j p\left(x_n|\mu_j, \sigma_j^2\right)} \tag{4}$$

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(x_k) x_n \tag{5}$$

$$\sigma_k^2 = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(x_k)(x_n - \mu_k)\,(x_n - \mu_k)^T \tag{6}$$

$$N_k = \sum_{n=1}^{N} \gamma(x_k) \tag{7}$$

$$\alpha_k = \frac{N_k}{N} \tag{8}$$

where N_k is the effective number of points assigned to component k and $\gamma(x_k)$ is the posterior probability which represents $p(x_k = 1|x)$ and can be found using Bayes theorem. The two main steps of EM is (i) evaluate posterior probability using the current parameter values, with equation (4) and

(ii) re-estimate the parameters using current posterior probabilities with equations (5), (6) and (7). The main steps are repeated until the solution converges by evaluating the log likelihood and checking for convergence of either the parameters or the log likelihood. If the convergence criteria are not satisfied, the main steps are repeated until the solution converges, provided that there is sufficient data.

2.4. Improved EM Algorithm

Although EM converges with sufficient data, the final model parameters are highly dependent on the training data and the initialisation of μ, σ^2 and K. If the initialisation is not optimal, the GMMs will not fit the data well and the output will vary, even for the same set of data. Furthermore, in the case of classification, a common initialisation for all datasets are required, thus increasing the difficulty in finding optimal parameters. A method to prevent this problem is the improved EM algorithm by Figueiredo and Jain [58], which is well suited for classification. The method does not require careful initialisation and is capable of selecting K. The algorithm only requires a minimum and maximum initial estimate for K, which is K_{min} and K_{max}. Here, EM algorithm is used in the traditional way, however the difference is in the likelihood function. An additional criterion is added to ML estimate and is derived by first considering the Minimum Message Length (MML), which aims at finding the "best" overall model instead of the "model-class/model" approach used in EM. According to Shannon theory [56], for $p(X|\mu, \sigma^2)$, the shortest code length is the ceiling of $\lceil -\log p\,(X|\mu, \sigma^2) \rceil$, since μ and σ^2 are unknown, the entire coding length is:

$$\text{Length}((\mu, \sigma^2), X) = \text{Length}(\mu, \sigma^2) + \text{Length}(X|\mu, \sigma^2) \tag{9}$$

The minimum encoding length criteria for MML is that the parameter estimate is the one minimizing $\text{Length}((\mu, \sigma^2), X)$. A finite code-length can be obtained by quantising μ and σ^2 to finite precision, the quantised version is denoted as $\hat{\mu}$ and $\widehat{\sigma^2}$. If a fine precision is used, then $\text{Length}\left(\hat{\mu}, \widehat{\sigma^2}\right)$ is large, which implies that $\text{Length}(X|\hat{\mu}, \widehat{\sigma^2})$ can be made small therefore, $\hat{\mu}$ and $\widehat{\sigma^2}$ can come close to the optimal value. The quantised version is as follows [56]:

$$\left(\widehat{\mu}, \widehat{\sigma^2}\right) = \arg\min_{\mu,\sigma^2}\left\{-\log p(\mu,\sigma^2) - \log\ p(X|\mu,\sigma^2) + \frac{1}{2}\log|I(\mu,\sigma^2)| + \frac{D}{2}\left(1 + \log\frac{1}{12}\right)\right\} \tag{10}$$

where, $I(\mu,\sigma^2) \equiv -\mathrm{E}\left[D^2_{\mu,\sigma^2}\log p(X|\mu,\sigma^2)\right]$ is the Fisher information matrix and $|I(\mu,\sigma^2)|$ is the determinant. Since $I(\mu,\sigma^2)$ cannot be determined analytically for mixtures, the expression is replaced by the complete-data Fisher information matrix with a block-diagonal structure. Therefore,

$I_c(\mu,\sigma^2) \equiv -\mathrm{E}\left[D^2_{\mu,\sigma^2}\log p(X,Z|\mu,\sigma^2)\right]$, which is the upper-bounds of $I(\mu,\sigma^2)$ [56].

$$I_c(\mu,\sigma^2) = n\ \mathrm{block} - \mathrm{diag}\left\{\alpha_1 I^{(1)}(\mu_1,\sigma_1^2), \ldots, \alpha_k I^{(1)}(\mu_k,\sigma_k^2), (\alpha_1\alpha_2 \ldots \alpha_k)^{-1}\right\} \tag{11}$$

The final criterion is derived from equation (10) and (11), the full derivation is found in [58], the additional criterion for ML is as follows:

$$\mathcal{L}((\mu,\sigma^2),X) = \frac{F}{2}\sum_{k:\alpha_k>0}\log\left(\frac{n\alpha_k}{12}\right) + \frac{K_{nz}}{2}\log\frac{n}{12} + \frac{K_{nz}(F+1)}{2} - \log p\,(X|\mu,\sigma^2) \tag{12}$$

where, K_{nz} is the number of non-zero-probability components, which is initially equal to the maximum initial estimate for K. While F is the number of free parameters in $p(x|\mu,\sigma^2)$. The aim is to obtain the minimum value of equation (12) to meet the new likelihood criterion. After convergence of EM, there is no guarantee that $\mathcal{L}((\mu,\sigma^2),X)$ is minimised. This is solved by excluding the least probable component of α_k and rerunning the EM algorithm until it converges. The process is repeated until $K_{nz} = K_{min}$, then each $\mathcal{L}((\mu,\sigma^2),X)$ is compared to find the minimum value. The model parameter set with the minimum value is chosen as the optimal set, $[K_{best}, \alpha_{best}, \mu_{best}, \sigma^2_{best}]$.

2.5. Gaussian Mixture Model Classification

GMM parameterisation enables the transformation of high-dimensional input spaces into lower-dimensional GMM parameter spaces, while retaining adequate object description for subsequent classification. Classification in lower-dimensional parameter space is simpler. The GMM model is a set of K parameter triplets $\{(\alpha_1, \mu_1, \sigma_1^2), \dots, (\alpha_K, \mu_K, \sigma_k^2)\}$. The first stage classifier is built to detect regions of interest (ROI) across the frame, which consists of vehicle detections and some false positives (background). The classifier is trained with a set of positive and negative samples. HoG-corner features are first extracted from all the positive samples to form $X_{(N_{pos},D)}$, while $X_{(N_{neg},D)}$ is generated for all the negative samples, where N_{pos} and N_{neg} are the total number of positive and negative samples data points respectively. Thereafter, GMMs are modelled on the sets $X_{(N_{pos},D)}$ and $X_{(N_{neg},D)}$ to establish model parameters for each set, $(\alpha_{pos}, \mu_{pos}, \sigma_{pos}^2)$ and $(\alpha_{neg}, \mu_{neg}, \sigma_{neg}^2)$. Each set represents the two different classes, vehicle and background, with the posterior probabilities as class labels. This classifier generates cropped images (ROIs) of potential detections. That are obtained during the comparative process. These ROIs are classified by the second classifier, to reduce false positives and verify vehicle detections. A similar process is applied to build the second stage classifier, however ROIs are used instead of frames and HoG-edge is used for feature extraction. The final comparative process produces the final vehicle detection locations.

GMM classification is achieved with a likelihood function as in [50]. The classification model is defined on space $\mathbb{C}$ that maps from the image domain to a set of C classes with each class, c, that corresponds to a ROI. Therefore, each classification $v \in \mathbb{C}$, assigns $c = v(fp) \in C$ to each feature data point. Optimal classification is chosen by using a loss function and by defining the posterior probability distribution on $\mathbb{C}$. Furthermore, each ROI is divided into B blocks, with individual blocks denoted as b; which corresponds to the block size used in the feature extraction. The likelihood of any ROI given the classification v, equation (13), and posterior probability of v given a ROI, equation (14), is as follows [50]:

$$Pr(ROI|v) = \prod_{b \in B} Pr(ROI_b|v_b) \qquad (13) \qquad Pr(v|ROI) = \prod_{b \in B} Pr(v_b|ROI_b) \qquad (14)$$

To derive estimates of the v, the loss function is used, illustrated in equation (15), while the expected value of the loss function [32] is shown in equation (16):

$$L(v^*, v) = -\sum_{b \in B} \prod_{b' \in P(b)} \delta(v_b^*, v_{b'}) \quad (15) \qquad \langle L \rangle (v^*) = -\sum_{b \in B} \left[\prod_{b' \in P(b)} Pr(v_{b'} = v_b^* | ROI_{b'}) \right] \quad (16)$$

where, v is the true classification with known posterior probability $Pr(v|ROI)$, while v^* is the proposed classification. The classification rule is formulated by minimising the mean loss and using the posterior probability from equation (14) [50]:

$$\hat{v}_b = \arg\max_{c \in C} \left[\prod_{b' \in P(b)} Pr(ROI_{b'} | v_{b'} = c) \right] \quad (17)$$

The classification rule $\hat{v}_b$ implies that the probability of the neighbourhood patch $P(b)$ of block b is maximised if all the blocks in the patch $P(b)$ had class c, of which, class c is assigned to block b. The full derivation of the classification rule with the likelihood function is shown in [50].

2.6. Detection Algorithm

The algorithm considers a set of frames from UAV video streams, and detection of vehicles are performed on each frame independently. The algorithm contains a main loop that iterates through all the frames and preforms the two-stage classification. Firstly, HoG-corner features are extracted from the entire frame and represented as GMM parameters. Thereafter, the parameters are classified based on the trained model parameters. The output is a matrix with all data points classified as either 'vehicle' or 'background'. Then data points denoted as vehicles are mapped back to the corresponding positions on the frame and stored in a matrix of positions. Overlapping positions within the same area are averaged and merged as one, then updated in the position matrix. An inner loop iterates through the matrix while defining ROIs within a box of fixed size that is centred on each positions. If the box size is not big enough, multiple boxes are assigned to a vehicle. The ROIs are cropped from the frame

and used by the second stage classifier. In this stage, HoG features are extracted from the ROIs and GMMs are used to obtain model parameters. Thereafter the ROIs are classified as either 'vehicle' or 'background', the output is 'true' if a vehicle is found and 'false' if it is a background element. If the classifier output is 'true', the ROI is highlighted on the frame, thus indicating vehicle detections. The detailed detection algorithm is illustrated as pseudocode in Fig. A.3.

Input: Set of video frames: $f = \{f_1, f_2, \dots, f_n\}$, n is the total number of frames
Output: Vehicle detections as boxed regions: detection = (x, y, width, height)

1. Main Loop: loop for all frames: **for** i = 1 to n **do**
2. Input single frame with loop number as index: $I = input(frame(i))$
3. Extract HOG-corner features from frame: $X_s = HOGcorner(I)$
4. Obtain GMM: $[\alpha_f, \mu_f, \sigma_f^2] = GMM(K_{min}, K_{max})$
5. Perform 1st stage GMM parameter classification, result matrix with class outcomes for all data points: $result1 = GMMclass1(\alpha_f, \mu_f, \sigma_f^2)$
6. Map data points with vehicle outcomes to corresponding positions on frame
7. Store positions: $pos_{(x,y)} = \{pos_{(x_1,y_1)}, pos_{(x_2,y_2)}, \dots pos_{(x_m,y_m)}\}$, m is the total number of detections
8. Merge overlapping positions and store new set of positions and m: $pos_{(x,y)} = pos_{(x,y)}^{new}$, $m = m^{new}$
9. Inner Loop: loop for all detection position: **for** j = 1 to m **do**
10. Define box around position with loop number as index: $box = (pos_{(x_j,y_j)}, width, height)$
11. Crop box to obtain ROI: $ROI = crop(box)$
12. Extract HOG-edge features from ROI: $X_{ROI} = HOGedge(ROI)$
13. Obtain GMM: $[\alpha_R, \mu_R, \sigma_R^2] = GMM(K_{min}, K_{max})$
14. Perform 2nd stage GMM parameter classification, result 'true' if classified as vehicle: $result2 = GMMclass2(\alpha_R, \mu_R, \sigma_R^2)$
15. **if** result = 'true' **then**
16. Highlight ROI with box on frame at current position
17. **end if**
18. **end inner loop**
19. **end main loop**

Fig. A.3 *Proposed detection algorithm*

3. Experimental Results

The proposed solution is implemented in Matlab and evaluated on the DARPA Video Verification of Identity (VIVID) dataset [53] ("egtest01", "egtest02", "egtest05" and "redteam"). The videos are captured from a single camera mounted on an aerial vehicle at 30 frames per second (fps) at a resolution of 640 × 480 pixels. All of the targets in the sequences are motor vehicles on the ground. The datasets provide a wide variety of troublesome scenarios including arbitrary and abrupt camera motion, out-of-focus video, target occlusions, multiple target interactions, moving background,

unrestricted pose variation, changes in illumination, and low contrast between objects and background. Thus, creating an extensive test evaluation.

3.1. Classifier Training

Training data required for the classifiers are obtained from a subset of the total frames, furthermore, only small ROIs from the frames are used. The training data is further separated from the test data by training the classifiers with a set that differs from the current test set (e.g. train with "egtest01", then tested on "egtest02". Although, it is possible to use one set. Approximately 5% of the test set is sampled across the whole sequence and used as training data. If too much data is given, the performance is hindered, due to the overlap in data. It is also important to 'balance' the classifier in order to prevent class bias. Thus, for stage one, an equal number of vehicle and background samples of the same size are used. The same applies to stage two, but samples of different vehicle orientations are included (front, back, side and 45° angle views). The training data used directly influences the output, thus the output is not predictable and depends on the training data quality and class assignment. Therefore multiple configurations may exist for optimal solutions and this is worth exploring further.

3.2. Test Evaluation Indicators

The evaluation of the two stage GMM classifier for detection of vehicles are performed on all frames of the chosen test sets. The stage one classifier, performing the initial detection, is first tested without the second stage classifier, which performs final validation. Then testing with both stages is conducted to generate final detection results. Qualitative results are illustrated by images highlighting vehicle detections with false positives, and quantitative results are represented by performance indicators. Two indicators are used, the Detection Rate (DR) and the False Alarm Rate (FAR), illustrated in equation (18) and (19) respectively. Where, True Positive (TP) is the detected regions that correctly correspond to vehicles, False Positive (FP): detected regions that falsely correspond to a vehicle, and False Negative (FN): failure to detect a vehicle. With the ideal being, *DR* = 1 and *FAR* = 0. For each dataset ("egtest01", "egtest02", "egtest05" and "redteam"), the total number of TP, FP and FN is applied to equation (18) and (19) respectively, to calculate the DR and FAR.

$$DR\% = \left(\frac{TP}{TP + FN}\right) \times 100 \qquad (18) \qquad FAR\% = \left(\frac{FP}{TP + FP}\right) \times 100 \qquad (19)$$

3.3. Stage One – Initial Detection Results

The stage one GMM-based classifier considered entire frames to provide initial detections for the next stage of classification. Since the feature space here does not differentiate between vehicle and background well, the FP detections are high. However the importance of this stage is to capture all potential positive detections. If a vehicle is not detected (FN), then the next stage will not be able to correct this error, as stage two only considers TP and FP from stage one. The qualitative results in Fig. A.4 illustrates all the vehicle detections (blue boxes) with a number of background elements being classified as vehicles. For the quantitative results, it is expected that both DR and FAR yields high values as indicated in Table A1. The increased sensitivity of the HoG corner feature set ensures high vehicle detection. The classifier is configured to except a low likelihood value. Furthermore, the feature space only represents locations, while the clusters of the corner detections are classified. Since the positions vary and the clusters differ for the same object, there is not enough data for full detection. The FAR is calculated based on TP and FP, since the FP values are high, the overall FAR yield high values. Higher FAR values are recorded in “egtest05” and “redteam” due to the increased number of FP caused by vegetation. The DR is based on TP and FN, therefore DR is high due to the acceptance of most detections (including FP). In some cases, where a vehicle is not detected (FN), this is caused by low contrast between objects and background, which may be caused by rapid camera movement and bad focus. As a result, shape detectors are unable to locate edges at object boundaries. This is especially evident in the end of “egtest02” where the camera moves rapidly from side to side and goes out of focus. Nonetheless, the overall DR recorded is relatively high. A 100% DR is recorded in “redteam” because there is no FN, which is ideal for detection.

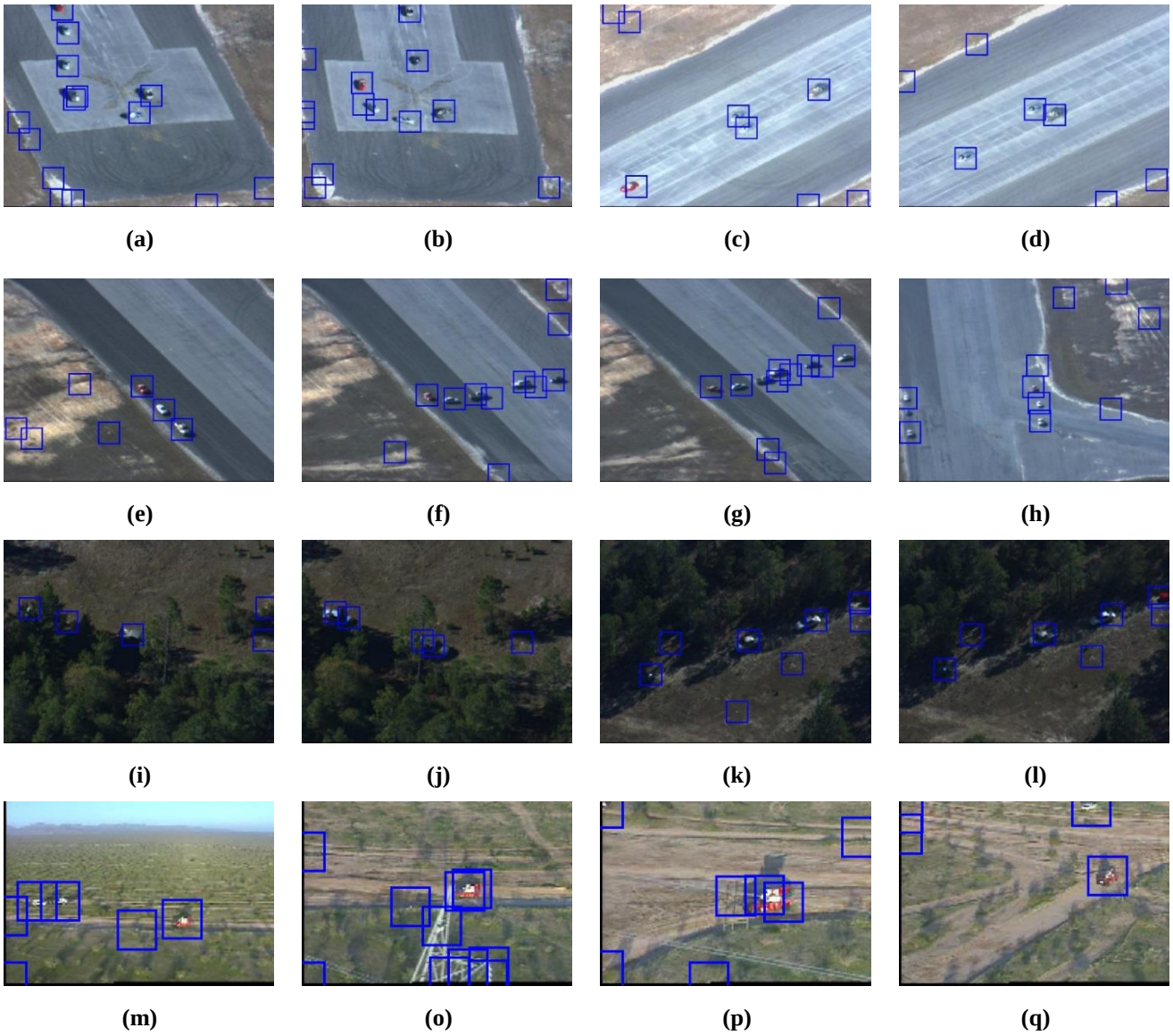


Fig. A.4. *Results from stage 1 classifier (initial detection) on VIVID datasets. (a)-(d): "egtest01", (e)-(h): "egtest02", (i)-(l): "egtest05" and (m)-(q): "redteam". As indicated, the classifier detects vehicles with many false positives.*

Table A.1: Quantitative results of stage 1 classifier on VIVID datasets

Test Data Sets	Detection Rate (DR) %	False Alarm Rate (FAR) %
Egtest01	96.36	42.71
Egtest02	92.73	49.14
Egtest05	94.28	53.88
Redteam	100	67.19

3.4. Stage Two – Final Detection Results

In this stage, the classifier solely considers ROIs generated from stage one, therefore the only potential improvement lies in the FAR. DR does not increase because there are no new TP detections. The classifier is capable of differentiating between vehicles and background, due to the feature space used and GMM classification. This classifier can be implemented on an entire frame, however, the feature space generates a large volume of data points which increases computational cost. The main purpose here is to reduce the high number of FP generated from stage one. Qualitative results visually illustrated this reduction, shown in Fig. A.5, while it is evident that the FAR is significantly reduced in the quantitative results, as indicated in Table A2. This result is due to the large difference between the GMM parameters, for vehicles and background. To further highlight the difference during classification, the ROIs are classified solely based on the likelihood of the prior probability representing the shape component. As a result, vehicles will always be classified correctly, while FPs occur for background elements containing shape components. However the algorithm tries to reduce multiple allocations for a single vehicle, to one box representing the detection. Therefore, in "egtest 02", where heavy overlap between vehicles occur, two vehicles are denoted as one, this causes a slight decrease in DR for "egtest02", as shown in Table A2 and illustrated in Fig. A.5, row 2. Note that multiple detections for the same object are considered as a single TP or FP for both stages. For "egtest01" and egtest02, FP are significantly reduced as in Table A2, while in "redteam" FP are from powerline structures, as shown in Fig. A.5, row 2. In "egtest05", a higher number of false positives are reported, due to highly dense vegetation in the scene, while FN are caused by shadows. The proposed solution is compared to two different methods that use the same dataset. Furthermore, the other methods also use feature extraction with classification. The method by Cheng [27] uses background colour removal, Harris Corner and Canny edge detection as features, then Dynamic Bayesian Networks for classification. While Xu [20] uses K-means clustering, Saliency region detection for features, then feature pooling with SVM classification. Table 3 shows the average DR and FAR for the VIVID datasets. The proposed work out performs both methods in terms of detection rate (DR).

Table A.2 Quantitative results of stage 2 classifier on VIVID datasets

Test Data Sets	Detection Rate (DR) %	False Alarm Rate (FAR) %
Egtest01	96.36	1.23
Egtest02	92.06	1.92
Egtest05	94.28	5.18
Redteam	100	2.04

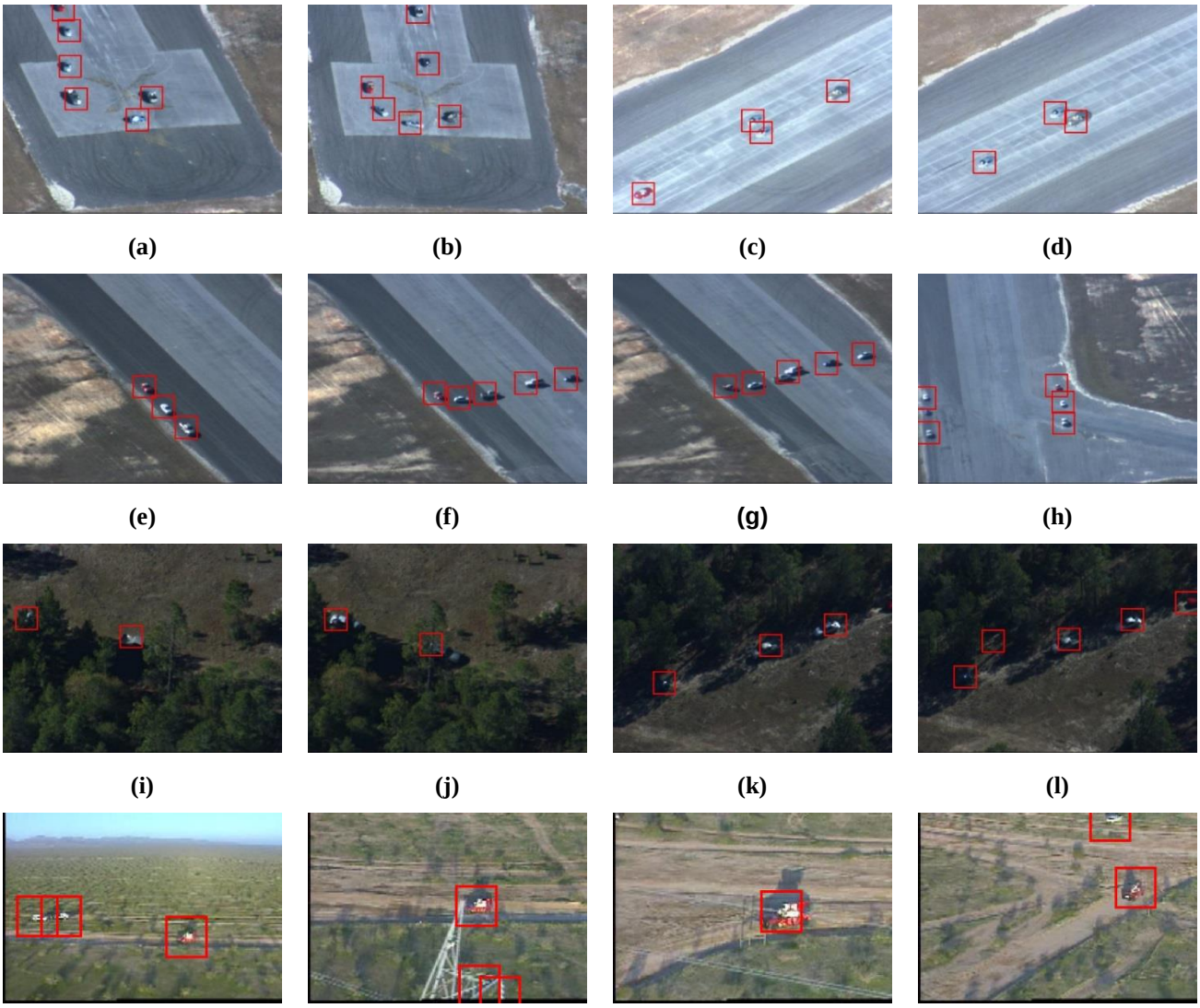


Fig. A.5. *Results from stage 2 classifier (validation) on VIVID datasets, (a)-(d): "egtest01", (e)-(h): "egtest02", (i)-(l): "egtest05" and (m)-(q): "redteam". As indicated, the classifier significantly reduced false positives from stage 1 as illustrated in Fig A.4*

Table A.3 Comparison with related works

Method	**Detection Rate (DR) %**	**False Alarm Rate (FAR) %**
Harris + Canny + DBN [27]	92.31	0.278
Saliency regions + SVM [20]	94.0	5.4
Proposed Solution	95.68	2.59

4. Conclusion

The paper presents an approach for detecting ground based vehicles from UAV video streams using GMM classifiers. A two-stage cascade of GMM classifiers is developed, the first stage initially detects potential vehicle ROIs, then passes them to the second stage classifier, which validates the detections while reducing false positives. Stage one utilises HoG-corner feature space for frame-wide detections, while the second stage uses HoG-edge which provides a finer level of detail for classification. An improved algorithm for fitting GMM models is combined with a likelihood function to form the GMM classification. GMMs are used to form model parameters from the training data, which are then compared to GMMs of current candidate test parameters with the likelihood function. The function yields numeric factors for each class, thus providing confidence levels for classifications. The specification and adoption of GMM based classifiers on commonly used feature spaces formed the principal contribution of the work. The training process highlighted the sensitivity of training data and class configuration. Therefore, multiple configurations need to be explored to find optimal solutions that may exist. The method presented here is tested on the commonly used DARPA VIVID dataset, and proved to be comparable with related works. Overall, the detector has proven to be tolerant to moving background, changes in illumination and target occlusion. Unrestricted pose variation is compensated for by including different vehicle orientations in the training data. Abrupt camera motion and out-of-focus video caused a high number of FNs, indicating that shape features are not tolerant against low contrast between objects and background. GMM classification has been beneficial in reducing dimensionality of feature spaces, and classification is performed on the parameters instead of the data. Furthermore, the use of the improved EM algorithm, eliminated the need for parameter initialisation. Additionally, the likelihood function simplified the overall classification process. The proposed method performed well in comparison to related works in terms of Detection Rate, but falls sightly short for False Alarm Rate. However, this can be improved with better training data and additional classes for vehicles. The method outlined here can easily be modified to detect arbitrary objects and be can applied in other research areas.

5. References

1. Khan, S.M., et al. *3D model based vehicle classification in aerial imagery*. in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. 2010. IEEE.

2. Kimura, M., et al. *Automatic extraction of moving objects from UAV-borne monocular images using multi-view geometric constraints*. in *IMAV 2014: International Micro Air Vehicle Conference and Competition 2014, Delft, The Netherlands, August 12-15, 2014*. 2014. Delft University of Technology.

3. Kumar, R., et al., *Aerial video surveillance and exploitation.* Proceedings of the IEEE, 2001. **89**(10): p. 1518-1539.

4. Pokheriya, M. and D. Pradhan. *Object detection and tracking based on silhouette based trained shape model with Kalman filter*. in *Recent Advances and Innovations in Engineering (ICRAIE), 2014*. 2014. IEEE.

5. Yang, W., et al. *Real-time detection algorithm of moving ground targets based on Gaussian mixture model*. in *SPIE Remote Sensing*. 2014. International Society for Optics and Photonics.

6. Priya, K.R. and R. Ramachandiran, *VEHICLE DETECTION AND TRACKING METHODS.* 2016.

7. Dave, S.A., D.M. Nagmode, and A. Jahagirdar, *Statistical Survey on Object Detection and tracking Methodologies‖.* International Journal of Scientific & Engineering Research, 2013. **4**(3).

8. Abdulrahim, K. and R.A. Salam, *Traffic Surveillance: A Review of Vision Based Vehicle Detection, Recognition and Tracking.* International Journal of Applied Engineering Research, 2016. **11**(1): p. 713-726.

9. Cho, H., P.E. Rybski, and W. Zhang. *Vision-based bicyclist detection and tracking for intelligent vehicles*. in *Intelligent Vehicles Symposium (IV), 2010 IEEE*. 2010. IEEE.

10. Mundhenk, T.N., et al. *Detection of unknown targets from aerial camera and extraction of simple object fingerprints for the purpose of target reacquisition*. in *IS&T/SPIE Electronic Imaging*. 2012. International Society for Optics and Photonics.

11. Cheraghi, S.A. and U.U. Sheikh. *Moving object detection using image registration for a moving camera platform*. in *Control System, Computing and Engineering (ICCSCE), 2012 IEEE International Conference on*. 2012. IEEE.

12. Siam, M. and M. ElHelw. *Robust autonomous visual detection and tracking of moving targets in UAV imagery*. in *Signal Processing (ICSP), 2012 IEEE 11th International Conference on*. 2012. IEEE.

13. Rodríguez-Canosa, G.R., et al., *A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera.* Remote Sensing, 2012. **4**(4): p. 1090-1111.

14. Cao, X., et al., *Vehicle detection and motion analysis in low-altitude airborne video under urban environment.* IEEE Transactions on Circuits and Systems for Video Technology, 2011. **21**(10): p. 1522-1533.

15. Jeon, B., et al. *Mode changing tracker for ground target tracking on aerial images from unmanned aerial vehicles (ICCAS 2013)*. in *Control, Automation and Systems (ICCAS), 2013 13th International Conference on*. 2013. IEEE.

16. Ma, Y., et al., *Pedestrian Detection and Tracking from Low-Resolution Unmanned Aerial Vehicle Thermal Imagery.* Sensors, 2016. **16**(4): p. 446.

17. Chen, X. and Q. Meng. *Robust vehicle tracking and detection from UAVs*. in *Soft Computing and Pattern Recognition (SoCPaR), 2015 7th International Conference of*. 2015. IEEE.

18. Teutsch, M. and W. Krüger. *Detection, segmentation, and tracking of moving objects in UAV videos*. in *Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on*. 2012. IEEE.

19. Kapsalas, P., et al. *Regions of interest for accurate object detection*. in *2008 International Workshop on Content-Based Multimedia Indexing*. 2008. IEEE.

20. Xu, Q., et al. *Air-ground vehicle detection using local feature learning and saliency region detection*. in *Intelligent Control and Automation (WCICA), 2012 10th World Congress on*. 2012. IEEE.

21. Mao, H., et al., *Automatic detection and tracking of multiple interacting targets from a moving platform.* Optical Engineering, 2014. **53**(1): p. 013102-013102.

22. Teutsch, M. and W. Kruger. *Robust and Fast Detection of Moving Vehicles in Aerial Videos using Sliding Windows*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2015.

23. Reilly, V., B. Solmaz, and M. Shah, *Shadow casting out of plane (SCOOP) candidates for human and vehicle detection in aerial imagery.* International journal of computer vision, 2013. **101**(2): p. 350-366.

24. Jones, P.V.a.M., *Rapid object detection using a boosted cascade of simple features.* IEEE Computer Vision and Pattern Recognition, 2001. **1**: p. 511-518.

25. Nizar, T.N., N. Anbarsanti, and A.S. Prihatmanto. *Multi-object tracking and detection system based on feature detection of the intelligent transportation system*. in *System Engineering and Technology (ICSET), 2014 IEEE 4th International Conference on*. 2014. IEEE.

26. Gleason, J., et al. *Vehicle detection from aerial imagery*. in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. 2011. IEEE.

27. Cheng, H.-Y., C.-C. Weng, and Y.-Y. Chen, *Vehicle detection in aerial surveillance using dynamic Bayesian networks.* IEEE transactions on image processing, 2012. **21**(4): p. 2152-2159.

28. Kersten, J., *Simultaneous feature selection and Gaussian mixture model estimation for supervised classification problems.* Pattern Recognition, 2014. **47**(8): p. 2582-2595.

29. Reynolds, D.A., T.F. Quatieri, and R.B. Dunn, *Speaker verification using adapted Gaussian mixture models.* Digital signal processing, 2000. **10**(1): p. 19-41.

30. Muthusamy, H., K. Polat, and S. Yaacob, *Improved emotion recognition using gaussian mixture model and extreme learning machine in speech and glottal signals.* Mathematical Problems in Engineering, 2015. **2015**.

31. Wiest, J., et al. *Probabilistic trajectory prediction with gaussian mixture models*. in *Intelligent Vehicles Symposium (IV), 2012 IEEE*. 2012. IEEE.

32. Deng, S., et al., *An infinite Gaussian mixture model with its application in hyperspectral unmixing*. Expert Systems with Applications, 2015. **42**(4): p. 1987-1997.

33. Štěpánek, M., J. Franc, and V. Kůs. *Modification of Gaussian mixture models for data classification in high energy physics*. in *Journal of Physics: Conference Series*. 2015. IOP Publishing.

34. Xiong, G., C. Feng, and L. Ji, *Dynamical Gaussian mixture model for tracking elliptical living objects.* Pattern Recognition Letters, 2006. **27**(7): p. 838-842.

35. Dai, P., et al., *A new approach to segment both main and peripheral retinal vessels based on gray-voting and gaussian mixture model.* PloS one, 2015. **10**(6): p. e0127748.

36. Kermani, S., N. Samadzadehaghdam, and M. EtehadTavakol, *Automatic color segmentation of breast infrared images using a Gaussian mixture model.* Optik-International Journal for Light and Electron Optics, 2015. **126**(21): p. 3288-3294.

37. Yousefi, S., et al., *Unsupervised Gaussian Mixture-Model With Expectation Maximization for Detecting Glaucomatous Progression in Standard Automated Perimetry Visual Fields.* Translational vision science & technology, 2016. **5**(3): p. 2-2.

38. Ragothaman, S., et al. *Unsupervised Segmentation of Cervical Cell Images Using Gaussian Mixture Model*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2016.

39. Zhan, X. and B. Ma, *Gaussian mixture model on tensor field for visual tracking.* IEEE Signal Processing Letters, 2012. **19**(11): p. 733-736.

40. Chauhan, A.K. and P. Krishan, *Moving object tracking using gaussian mixture model and optical flow.* International Journal of Advanced Research in Computer Science and Software Engineering, 2013. **3**(4).

41. Kim, J., Z. Lin, and I.S. Kweon, *Rao-Blackwellized particle filtering with Gaussian mixture models for robust visual tracking.* Computer Vision and Image Understanding, 2014. **125**: p. 128-137.

42. Quast, K. and A. Kaup, *Shape adaptive mean shift object tracking using gaussian mixture models,* in *Analysis, Retrieval and Delivery of Multimedia Content*. 2013, Springer. p. 107-122.

43. Santosh, D.H.H., et al., *Tracking Multiple Moving Objects Using Gaussian Mixture Model.* International Journal of Soft Computing and Engineering (IJSCE), 2013. **3**(2).

44. Li, N., et al., *Object Tracking with Multiple Instance Learning and Gaussian Mixture Model.* Journal of Information and Computational Science, 2015. **12**(11): p. 4465-4477.

45. Lee, D.-S., *Effective Gaussian mixture learning for video background subtraction.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005. **27**(5): p. 827-832.

46. Mukherjee, D., Q.J. Wu, and T.M. Nguyen, *Gaussian mixture model with advanced distance measure based on support weights and histogram of gradients for background suppression.* IEEE Transactions on Industrial Informatics, 2014. **10**(2): p. 1086-1096.

47. Zhou, D. and H. Zhang. *Modified GMM background modeling and optical flow for detection of moving objects*. in *2005 IEEE International Conference on Systems, Man and Cybernetics*. 2005. IEEE.

48. Xue, K., et al., *Panoramic Gaussian Mixture Model and large-scale range background substraction method for PTZ camera-based surveillance systems.* Machine vision and applications, 2013. **24**(3): p. 477-492.

49. Fradi, H. and J.-L. Dugelay. *Robust foreground segmentation using improved gaussian mixture model and optical flow*. in *Informatics, Electronics & Vision (ICIEV), 2012 International Conference on*. 2012. IEEE.

50. Permuter, H., J. Francos, and I. Jermyn, *A study of Gaussian mixture models of color and texture features for image classification and segmentation.* Pattern Recognition, 2006. **39**(4): p. 695-706.

51. Permuter, H., J. Francos, and I.H. Jermyn. *Gaussian mixture models of texture and colour for image database retrieval*. in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*. 2003. IEEE.

52. Tao, J., et al., *A study of a Gaussian mixture model for urban land-cover mapping based on VHR remote sensing imagery.* International Journal of Remote Sensing, 2016. **37**(1): p. 1-13.

53. Collins, R., X. Zhou, and S.K. Teh. *An open source tracking testbed and evaluation web site*. in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*. 2005.

54. Stavens, D. and S. Thrun. *Unsupervised learning of invariant features using video*. in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. 2010. IEEE.

55. Moranduzzo, T. and F. Melgani. *Comparison of different feature detectors and descriptors for car classification in UAV images*. in *2013 IEEE International Geoscience and Remote Sensing Symposium-IGARSS*. 2013. IEEE.

56. Cover, T.M. and J.A. Thomas, *Elements of information theory*. 2012: John Wiley & Sons.

57. Moon, T.K., *The expectation-maximization algorithm.* IEEE Signal processing magazine, 1996. **13**(6): p. 47-60.

58. Figueiredo, M.A.T. and A.K. Jain, *Unsupervised learning of finite mixture models.* IEEE Transactions on pattern analysis and machine intelligence, 2002. **24**(3): p. 381-396.

Paper B

Gaussian Mixture Model classifiers for tracking in UAV video streams

T. Pillay and B. Naidoo

Submitted for review to International Journal of Remote Sensing

Abstract

Manual visual surveillance systems are subject to a high degree of human-error and operator fatigue. The automation of such systems often employs trackers and classifiers as fundamental building blocks. Tracking and classification are especially useful and challenging in Unmanned Aerial Vehicle (UAV) based surveillance systems. Previous solutions have addressed the challenges with complex classification, however these methods require a set of labelled instances for separating the tracked object and there is insufficient instances for online learning. This paper aims to simplify the classification process with a minimised set of unlabelled instances to reduce the problems experienced by classification. The objectives are demonstrated with a vehicle tracker using colour histograms, with Gaussian Mixture Model (GMM) classifiers and a Kalman filter. GMMs provide a concise parametric representation of the histograms. Subsequent classification is used to differentiate the tracked object from other elements in the scene using a likelihood function. While the Kalman filter provides an initial estimate of the location, thus reducing the search space. The GMM classification model is constantly updated with a limited set of instances obtained over time. This allows the model to adapt to the changes in the appearance of the tracked object with fewer instances. GMM classification has resulted in a simplified classification process which is tested on standard datasets, and performs well in comparison to related works.

1 Introduction

Tracking is an active research area within visual surveillance, more specifically, tracking from aerial platforms such as (unmanned aerial vehicles) UAVs. These camera platforms have the advantage of broader surveillance scope and higher mobility. However, previous studies have identified numerous challenges; moving background [1], unrestricted pose variation [2], illumination [3], and low contrast between objects and background [4]. Despite these challenges, there is a motivated need for this technology [1, 2, 4, 5]. In addition, automated tracking systems reduce human-operator workload and human error.

The aim of object tracking is to locate and associate the position of an object over time from consecutive video frames. In multiple object tracking [6, 7], the association of each object is crucial, whereas for selected object tracking [8] it is important to differentiate the selected object. Some commonly used elements for both are; background subtraction [9], motion detection [10], segmentation [6] and foreground detection [11].

A common approach is to use static tracking algorithms, such as, Kalman and particle filters. The Kalman filter estimates the object position in the next frame, using previously estimated states and current measurements to recursively estimate the next state [12], which is used by Cheraghi [13] to track regions from UAVs. While the particle filter sequentially estimates the latent state variables from a sequence of observations using Monte Carlo sampling techniques [12], used by Cao [14] to track colour and Hu features from aerial imagery.

Another feature based method demonstrated by Chen [15], uses SIFT features and classification instead of tracking algorithms. There are other proposed methods that treat the tracking problem as a classification task [16-20]. Despite the success they have demonstrated, numerous issues remain to be addressed. Firstly, these methods need a set of labelled training instances (samples) to determine the decision boundary for separating the target object. Secondly, in most cases, due to change in appearance, there may be insufficient instances. However Gaussian Mixture Models (GMM) classification is an efficient unsupervised alternative that is capable of labelling instances and requires fewer instances. Since classification is a challenge for tracking, it is worth investigating the use of GMM classification.

GMM is a closed-form representation of arbitrary univariate and multivariate distributions as a convex combination of Gaussian distributions. Thus GMM has gained recognition due to its ability to represent some classes of real-world data in an efficient and accurate manner [21].

A common method for using GMMs in tracking is background subtraction as a pre-processing step, due to the ability to handle complex background scenes [22-27]. However, GMM background subtraction requires additional methods of optimisation for noisy or nonstationary backgrounds. Other approaches incorporate tracking algorithms with GMM. Quast [28], proposed a shape adaptive object tracker with GMM and the mean shift algorithm. Whereas, Kim [29], developed a robust visual tracker by combining GMM with a particle filter. The tracking algorithm methods above, searches a localised area around the previous state with no prior knowledge of the next state. Prior knowledge can consist of predictions of the next state, which the Kalman filter provides. Xiong [30], used the Kalman filter to estimate the state for parameters of GMM, for tracking elliptical living objects. These methods produces good results in their respective fields, however for UAVs, additional methods such as classification are required.

A GMM classification approach by Permuter [31, 32], who applied GMM on colour and texture features for image classification and segmentation. The classification with GMM was achieved through Expectation Maximization (EM) and Maximum Likelihood (ML) algorithm but careful initialisation of GMM parameters are required. To overcome this problem, Tao [33] applied Figueiredo and Jain (FJ) algorithm instead of EM, which does not require initialisation of parameters. They developed an optimised GMM classifier with FJ and SVM, applied to VHR remote sensing images in urban areas. The methods from Permuter [31] and Tao [33] can be merged and adapted for classification within tracking from aerial videos. The FJ algorithm can improve classification, while ML can simplify the classification process.

This paper demonstrates the specification and adoption of GMM classification to track a user selected ground vehicle from UAV video streams. The user selection initialises the problem, then the tracker continues in an unsupervised manner. Colour histograms forms the feature space for vehicles, which is represented as GMM parameters. Then GMM classification is used to differentiate the selected object from other elements in the scene. The classification is conducted with a likelihood function on model parameters and not on the data, thus simplifying the process. The model parameters are constantly updating in order to adapt to changes in the appearance of the tracked vehicle. The GMM classifier is attached to a Kalman filter for next state estimations, which reduces the subsequent search space. The solution is limited to local tracking of ground vehicles, with the assumption that the location of the vehicle and the model does not change significantly between successive and successful image captures. The paper continues onto Section 2, which reviews the various parts of GMM and presents the proposed solution. Section 3, describes the test evaluation and shows the experimental results with comparisons with related works, while Section 4 presents the conclusions.

2 Gaussian Mixture Model Classification and Tracking

The proposed method requires various elements, namely, colour feature extraction (Section 2.1), GMM parameterisation (Section 2.2), GMM parameter classification (Section 2.3), tracking algorithm functionality (Section 2.4) and finally Kalman filter estimation, (Section 2.5). Fig B.1, shows a block diagram of the proposed solution.

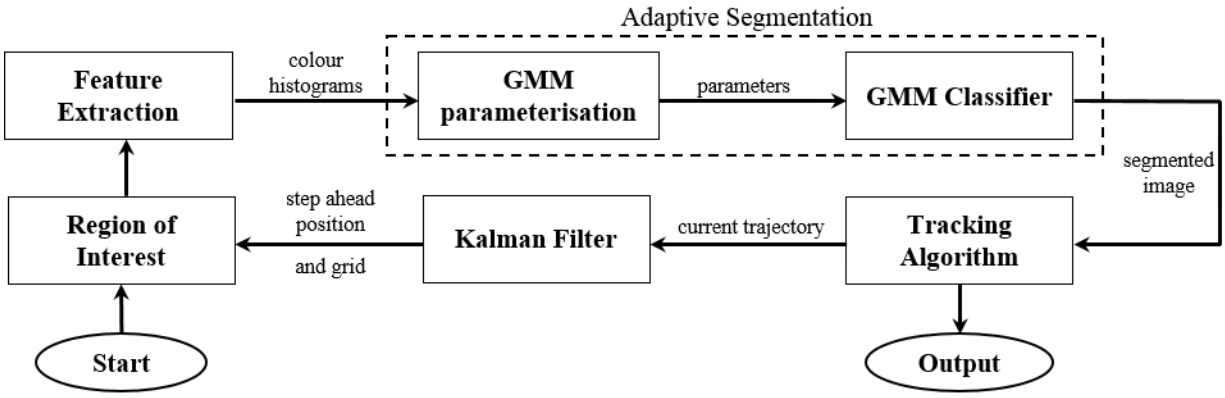


Fig. B.1 *Proposed solution block diagram*

2.1 Colour Feature Extraction

The feature extraction process represents the attributes that identify and distinguish objects. Colour histograms are used, as they depict a rich source of information. However changes in perspective, illumination and scale causes variations in the colour. Multiple colour spaces can represent a wider spectrum, thus more tolerant to change. Some works consider colour spaces other than RGB, namely, and LAB [31, 32] and HSV (hue, saturation, value) [15, 34]. RGB does not require transformations to perceive the colour and has a lower computational cost, however it is difficult to determine specific colour in the RGB model [35]. LAB is perceived as uniform but is not intuitive. While for HSV, the hue and saturation components is the way humans perceive colour and well suited for image processing. However, undefined achromatic hue points are sensitive to value deviations of RGB and instability of hue [35]. Therefore both RGB and HSV were combined for better performance. Let $H_{(C,D)} \in \{H^{RGB}, H^{HSV}\}$, where $H_{(C,D)}$ is a set of colour histograms, C is the number of colour spaces and D is the number of dimensions representing the data. While H^{RGB} are colour histograms in RGB and H^{HSV} are histograms in HSV colour space. Here $C = 2$ and $D = 6$, since RGB and HSV has 3 channels each.

2.2 Gaussian Mixture Models

The general assumption is that, real world data distributions form multivariate Gaussian distributions when a set of data is represented as a probability density functions (PDF). The multivariate distributions can be approximated by a linear combination of Gaussian distributions with the use of GMMs. Because GMMs provide a concise parametric representation of the distributions. The data to be represented is produced by the colour histograms of the feature extraction step, $H_{(C,D)} = H\{h_1, \dots, h_D\}$, where the individual histograms forms multivariate Gaussian distributions. Therefore the PDF of a sample h is approximated as a convex function of multiple components using GMM [36]:

$$p(h|\theta) = \sum_{k=1}^{K} \alpha_k p(h|\theta_k) \tag{1}$$

where, $\theta_k = (\mu_k, {\sigma^2}_k)$ is defined as a vector, with μ and σ^2 are the mean and variance parameters respectively. Each component k has a mixture weight α_k (prior probabilities) assigned, all α_k sum to 1 [36].

$$\sum_{k=1}^{K} \alpha_k = 1 \tag{2}$$

Generally, the Expectation Maximisation (EM) algorithm is used to estimate the vector θ and parameters α and to converge equation (1) to a close form solution [36]. However, the convergence of the EM algorithm is highly dependent on the data samples and the initialisation of θ and α. If the initialisation is not optimal, parameter estimation may not converge. To eliminate the need for initialisation, the improved EM algorithm by Figueiredo and Jain (FJ algorithm) [37] is used. The FJ algorithm is able to select an appropriate value for K, provided that minimum (K_{min}) and maximum (K_{max}) estimates for K are stipulated.

The FJ algorithm utilises expectation and maximisation steps of EM, but uses a different approach for the likelihood function.

2.3 Gaussian Mixture Model Classification

Model representation through GMM parameters provides a simplified form for classification. This is due to the entire set $H_{(C,D)}$ represented by (α, θ) parameters and comparisons are made only between these parameters. Furthermore, model updates simply requires new data samples to be added to $H_{(C,D)}$, then GMM applied to obtain updated model parameters. The GMM model is a set of K parameters $\{(\alpha_1, \theta_1), \ldots, (\alpha_K, \theta_K)\}$. Next the likelihood function is used for GMM classification similar to the method used in [31]. This method is used to segment the adaptive foreground object from the background. Successive GMM models track the adaptation of the foreground and enables the classifier to adaptively segment. The classification model regions of interest (ROI) is assigned to class c, which is a subset of C classes. Therefore the model is defined on space $\mathbb{C}$, which maps from the image domain to the set of C classes. This implies that, each classification $v \in \mathbb{C}$, assigns $c = v(p) \in C$ to each pixel p. If the posterior probability distribution is defined the on $\mathbb{C}$ and by using a loss function, optimal classification is achieved. If the ROIs are divided into B blocks, with individual blocks, b; the likelihood of a ROI given the classification v is defined in equation (3). The posterior probability of v given a ROI is defined in equation (4) [31]:

$$Pr(ROI|v) = \prod_{b=1}^{B} Pr(ROI_b|v_b) \qquad (3) \qquad Pr(v|ROI) = \prod_{b=1}^{B} Pr(v_b|ROI_b) \qquad (4)$$

Equation (5) shows the loss function used to derive estimates of the v, and equation (6) shows the expected value of the loss function, $L(v^*, v)$ and expected value of this loss function, $\langle L \rangle(v^*)$, [39]:

$$L(v^*, v) = -\sum_{b=1}^{B} \prod_{b'=1}^{P(b)} \delta(v_b^*, v_{b'}) \qquad (5) \qquad \langle L \rangle(v^*) = -\sum_{b=1}^{B} \left[\prod_{b'=1}^{P(b)} Pr(v_{b'} = v_b^* | ROI_{b'}) \right] \qquad (6)$$

The true classification with known posterior probability is v, whereas the proposed classification is v^* and b' is a function limited to block b. Using the posterior probability from equation (4) and minimising the mean loss, the classification rule is formulated [39]:

$$\hat{v}_b = \arg\max_{c \in C} \left[\prod_{b'=1}^{P(b)} Pr(ROI_{b'} | v_{b'} = c) \right] \quad (7)$$

$P(b)$ is the neighbourhood patch of block b, which is maximised if all the blocks in the patch $P(b)$ had class c, of which class c is assigned to block b. This defines the classification rule $\hat{v}_b$ [38] contains the full derivation with the likelihood function. The final outcome of the classifier is a segmented image of the selected vehicle to be tracked. Although classification is applied to fixed grids, the segmentation is not restricted to an individual grid, as each pixel is individually classified as either background or foreground.

2.4 Tracking Algorithm

The first step is to input the video frames $f = \{f_1, f_2, \ldots, f_n\}$, and an initial ROI. Next, a classification model is built by extracting colour histograms, $H_{(C,D)}$, from the ROI, and applying GMMs to $H_{(C,D)}$ to form model parameters $M_c = (\alpha_K, \theta_K)$. Thereafter the GMM classifier defines pixels as foreground or background and outputs the new segmented ROI, which indicates the new position of the vehicle. This new position and the model parameters are given to the Kalman filter to update the state variable that encodes the predicted (step ahead) location of the object. The Kalman filter estimates the next state (i.e. location). From this predicted location, a future 3x3 search grid is defined and colour histograms, $H_{c_{(C,D)}}$, around the new region are extracted on arrival of the next frame. Thereafter GMM classification is applied to $H_{c_{(C,D)}}$ to determine which pixels belong to the selected object with the likelihood function. Once the objects new location is found, it marked with a boarder and the classification model is updated. The model is updated by adding the data points from $H_{c_{(C,D)}}$ to $H_{(C,D)}$, then recalculating the GMM parameters. The final step is to update the state of the Kalman filter with the new parameters. The process is repeated for every frame, while the update occurs in every 5 frames. The detailed tracking algorithm is illustrated as pseudocode in Fig B.2.

Input: Set of video frames: $f = \{f_1, f_2, \ldots, f_n\}$, and ROI

Output: Boarder around tracked object $track = (x, y, width, height)$

1. Input first frame: $frame = f_1$
2. Input ROI: $ROI = (x, y, width, height)$
3. Extract colour histograms from ROI: $H_{(C,D)} = \text{colourHistogram}(ROI)$
4. Obtain classification model parameters from $H_{(C,D)}$: $M_c(\alpha_K, \theta_K) = GMM(H_{(C,D)})$
5. GMM classifier outputs new ROI position: $ROI_{new} = (x, y, width, height)$
6. Kalman filter to update state: $\text{UpdateState}(M_c(\alpha_K, \theta_K))$
7. Loop through all frames: **for** i = 1 to n **do**
8. Extract next ROI using updated state: $pos(x, y) = \text{NextState}(M_c(\alpha_K, \theta_K))$
9. Extract colour histogram around $pos(x, y)$: $H_{c_{(C,D)}} = \text{ColourHistogram}(ROI)$
10. GMM classify pixels in new ROI: $ROI_{(classF,classB)} = \text{GMMclassify}(H_{c_{(C,D)}})$
11. Output track boarder: $track(x, y, width, height) = Output(ROI_{(classX,classY)})$
12. Update classification model: $M_c(\alpha_K, \theta_K) = GMM(H_{(C,D)} + H_{c_{(C,D)}})$
13. Kalman filter to update state: $\text{UpdateState}(M_c(\alpha_K, \theta_K))$
14. **end** loop

Fig. B.2 *Proposed tracking algorithm*

2.5 Kalman Filter Estimation

The Kalman filter is used to predict the next state (i.e. location) of the selected vehicle, these predictions are used as the initial parameters of the next frame. The state represents a physical location, while state updates represent predicted motion. The motivation for the filter's use is to reduce the search space in the subsequent frame, thus lowering number of iterations and reducing computational cost. Kalman filters operate as a set of recursive mathematical equations that implement a predictor–corrector type estimator. The objective is to predict the object's position for the next frame from the previous frame. This is achieved if states satisfy the linear time-invariant model, defined as [30]:

$$\begin{Bmatrix} s(t+1) = \Theta s(t) + u(t) \\ m(t) = \Psi s(t) + v(t) \end{Bmatrix} \quad (8)$$

where, $s(t)$ and $m(t)$ are the state vector and measurement vector at time t, respectively. Θ is a matrix relating $s(t)$ to $s(t+1)$, Ψ is a matrix relating states and measurements. $u(t)$ and $v(t)$ zero mean and covariance of Gaussian white noise. The final estimates are predicted by the following steps:

1. Initialise error covariance matrix $E(0|-1) = \Pi_0$
2. Initialise state value $s(0|-1) = 0$
3. Calculate the Kalman gain $K(t)$ and $G(t)$:
   $$G(t) = \Psi E(t|t-1)\Psi^T + v(t) \quad (9)$$
   $$K(t) = E(t|t-1)\Psi^T \quad (10)$$
4. Calculate the estimated state vector $\hat{s}(t+1|t)$:
   $$\hat{s}(t+1|t) = \Theta\hat{s}(t|t-1) + \Theta K(t)G(t)^{-1} \times (m(t) - \Psi\hat{s}(t|t-1)) \quad (11)$$
5. Update the error covariance matrix $E(t+1|t)$:
   $$E(t+1|t) = \Theta E(t|t-1)\Theta^T + u(t) - \Theta K(t)G(t)^{-1}K(t)^T \quad (12)$$
6. Return to step 2

3 Experimental Results

The proposed solution is implemented on MatLab and evaluated on the DARPA Video Verification of Identity (VIVID) dataset [56]. VIVID is an open source evaluation and tracking testbed. Tests are performed on the following test sets: “egtest01”, “egtest02”, “egtest04” and “egtest05”. The videos are captured from a single low resolution camera mounted on an aerial vehicle. The datasets provide an extensive test evaluation, as it includes arbitrary and abrupt camera motion, out-of-focus video, target occlusions, multiple target interactions, moving background, unrestricted pose variation, changes in illumination, and low contrast between objects and background.

3.1 Kalman Estimation

The Kalman filter does not enhance the classification accuracy, rather it improves the computation cost by reducing the search space, thus reducing the number of iterations. The filter estimates the position of the object in the next frame, while GMM classification is used to detect the object and provide the exact position. Kalman filters provide the state (position) estimates for the next frame while the GMM classifier provides exact or detected location in the current frame. This is repeated for

all frames, thus accurate and up to date information is exchanged between the two parts. In addition, when GMM classification cannot find a suitable fit, the filter's prediction is used instead. This occurs during full occlusion of the object and if insufficient data points are extracted for the colour histograms, as can happen during extreme camera defocusing. Additionally, during rapid camera motion, the Kalman filter may cause a loss track due to the incorrect estimates of the position.

3.2 Classification Model Update

In order to build a strong classification model that represents the selected object, the model has to be updated as more instances are provided over time. This allows the model to adapt to the changes in the appearance of the selected object caused by scale, pose variation and illumination. The model is updated every 5 frames, which forms a model that represents the object accurately while keeping computational cost low. A matrix with first-in-first-out (FIFO) principle is implemented to store a current list up to 50 instances that are updated over time. This simplifies the updating process as classification models are formed from the matrix at different points in time. Although it is beneficial to have more instances, there is a point where too many samples causes the performance to deteriorate. Model quality decreased with extreme changes in the appearance of the object, therefore older samples are no longer applicable. The addition of this historical storage matrix proves to be beneficial, however an optimal solution would be one that can dynamically change the size of the matrix and the model update depending on the current scenario. This can be based on error observations between current and previous tracks.

3.3 Test Evaluation Indicators

The test evaluation of the proposed method is performed on all frames of the chosen test sets. Tracking results are compared to the ground truth values. To evaluate the performance, tracking rate (TR) is computed by:

$$TR = \frac{NFST}{NT} \tag{13}$$

where, $NFST$ is the number of times that vehicles are successfully tracked, while NT is the number of times that a vehicle appears in the sequence. Therefore, there can only be one successful track per single frame. A successful track is defined by the precision (P) and recall (R), which is denoted as [14]:

$$P = E \cap \frac{G}{E} \quad (14) \qquad R = E \cap \frac{G}{G} \quad (15)$$

where, E is the estimated area of the bounding box and G is the ground truth of the selected vehicle. A track is only denoted as successful if the precision and recall are both above 0.5.

3.4 Tracking via GMM Classification

The algorithm requires a user selected the object to be tracked in the first frame. This selection effects the overall performance. Therefore, for the purpose of uniform testing, the selection is predetermined and provided by the data test sets. In addition, the K_{min} and K_{max} values chosen for the FJ algorithm has an influence on the output. A larger K_{max} value leads to more iterations, while, if K_{max} is too small, the representation of data is inefficient and inaccurate. For the current application, $K_{min} = 1$ and $K_{max} = 10$.

The test results for all frames of "egtest01", "egtest02", "egtest04" and "egtest05", are represented quantitatively with tracking rate (TR) from equation (13) while the qualitative results highlights challenging events within the video sequences.

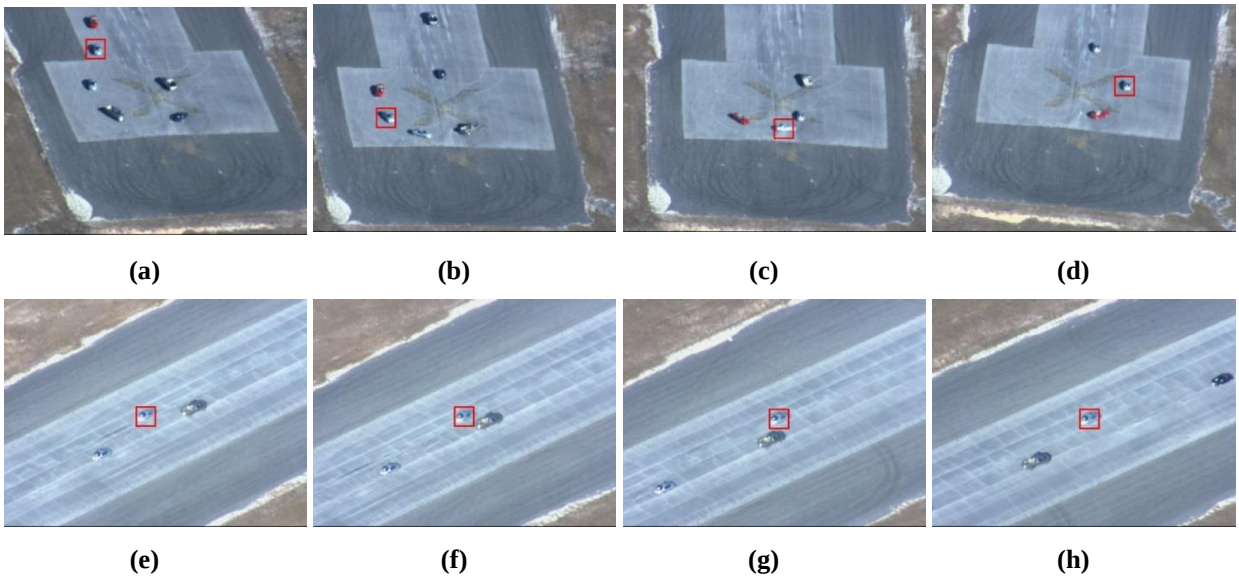


Fig. B.3 *Tracking sequences from "egtest01", (a)-(d) illustrates pose variation and changes in illumination as vehicles circle around, and (e)-(h) illustrates vehicle interaction as tracked vehicle overtakes another vehicle.*

The challenging events in "egtest01" occur when the vehicles make a U-turn, causing pose variation and changes in illumination as the sun reflects off different planes of the vehicles. A visual inspection of the frames in Fig B.3 (a)-(d) illustrates the colour variation. However the GMM classifier is able to

overcome this problem with the aid of the model update. Other challenging events occur when the tracked vehicle overtakes other vehicles, as shown in Fig B.3 (e)-(h). In this interaction where vehicles move closer to each other can cause the wrong vehicle to be tracked.

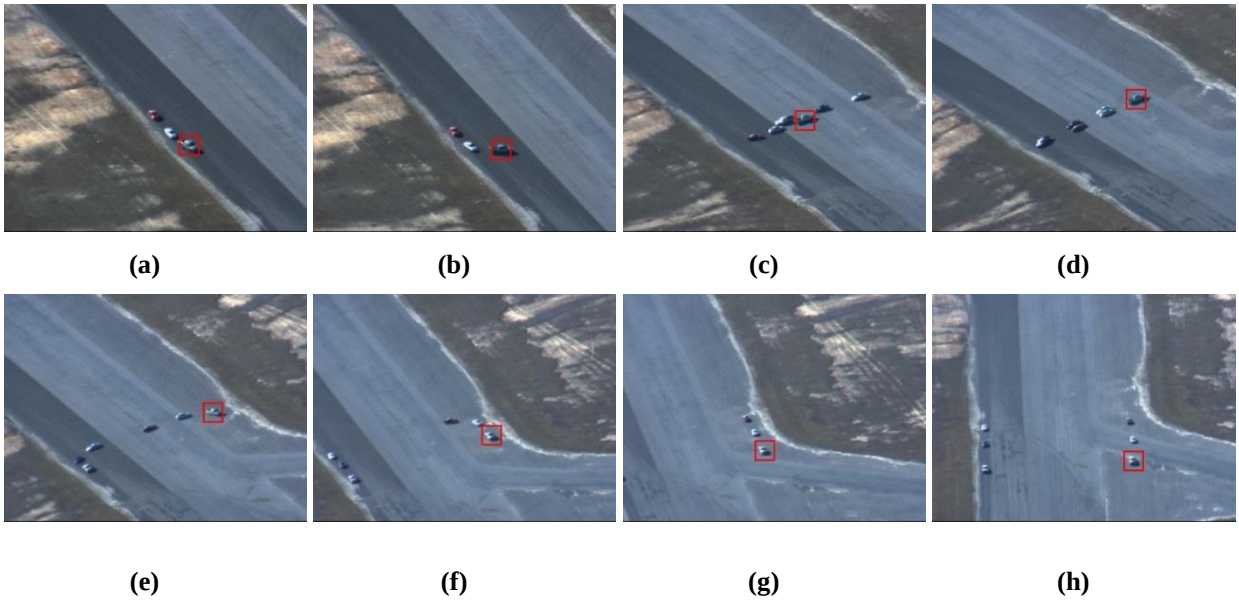


Fig. B.4 *Tracking sequences from "egtest02", (a)-(d) illustrates pose variation and vehicle interaction as vehicles pass each other, and (e)-(h) illustrates change of scale and rapid camera movement.*

In "egtest02" the vehicle interaction is increased when vehicles pass each other from opposite ends, causing vehicles to overlap and appear as a single vehicle, as illustrated in Fig B.4 (a)-(d). This reduces the quality of the classification model as other vehicles are included in the model update which decreases the precision. Fig B.4 (e)-(h) illustrates the change in scale and rapid camera movement. Scale does not affect the GMM classification, however the Kalman filter is unable to account for the camera movement and thus estimates incorrectly.

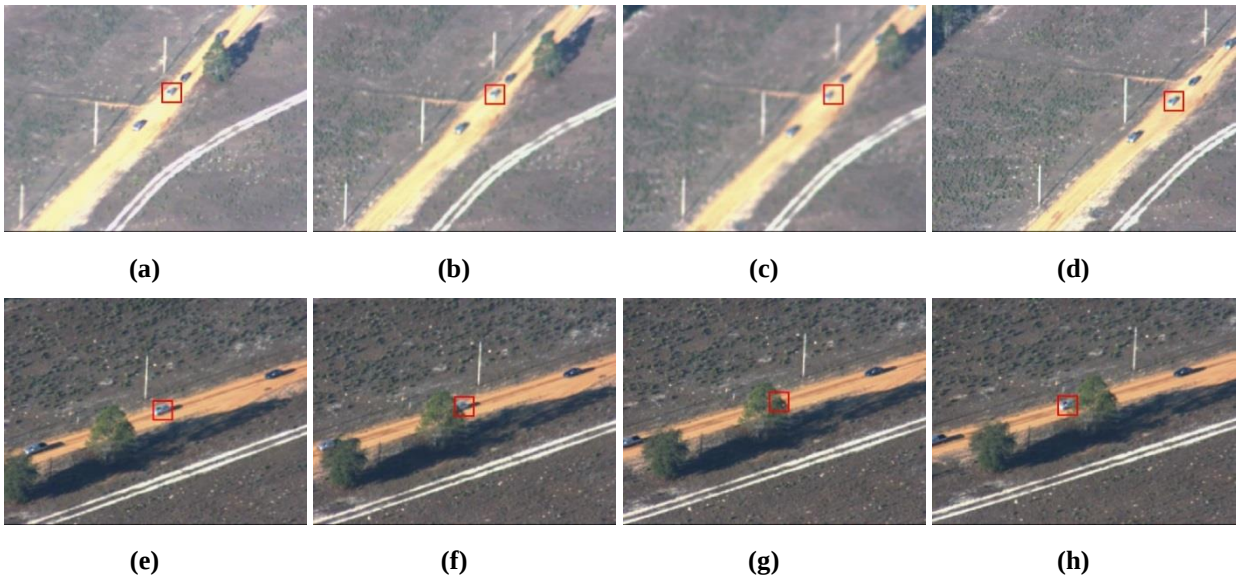


Fig. B.5 *Tracking sequences from "egtest04", (a)-(d) illustrates camera defocusing and dropped frames which are duplicated in the sequence (no motion), and (e)-(h) illustrates full occlusion as tracked vehicle passes trees.*

The challenges in "egtest04" are illustrated in Fig B.5 (a)-(d). The camera defocuses and there are some frames that are dropped, causing no motion, followed by a sudden discontinuity. The GMM classifier does not cope well with camera defocus, because (i) object and background are less distinguishable, (ii) the clearly focused historic data does not represent the defocused current frame. Whereas for cases of dropped frames, the filter fails. For the sudden discontinuity, both means fail because the tracker only preforms tracking within a localised region. Other difficulties are full occlusion which occurs when the vehicle passes trees, as illustrated in Fig B.5 (e)-(h).

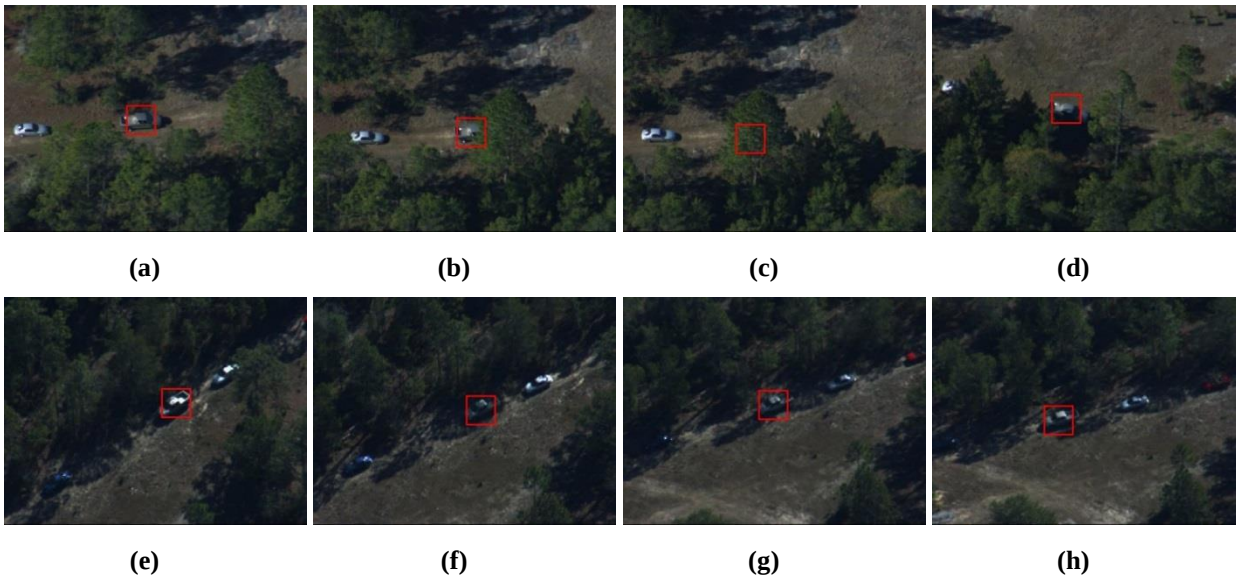


Fig. B.6 *Tracking sequences from "egtest05", (a)-(d) illustrates full occlusion as tracked vehicle passes trees, and (e)-(h) illustrates changes in illumination as vehicles pass in and out of tree shadows.*

A further increase of full occlusion events are represented in "egtest05", as vehicles pass through highly dense vegetation, shown in Fig B.6 (a)-(d). In addition, the test set contains events of extreme illumination changes, as vehicles pass in and out of areas with shadows created by trees, illustrated in Fig B.6 (e)-(h). These events cause variations between the colour histograms for each frame. However the GMM classifier overcomes the problem with the model update which includes instances that incorporate the changes in illumination.

The quantitative results obtained from the test sets are directly compared to related works that have used the same VIVID tests, as shown in Table B.1. Mao et al [9] uses background subtraction to extract moving objects, then uses data association to evaluate overlap rates between the moving objects. Whereas, Hasan et al [40] first detects motion regions from stabilised videos then identifies targets of interest around the motion regions using appearance based pre-trained classifiers. The classifier uses a finite state machine (FSM) that incorporates both motion detection and target classification into a Kalman filter. These methods as well as the proposed solution considers all the frames within each test.

Table B.1: Quantitative results with track rate on VIVID datasets and comparisons with related works

Method	**"egtest01"** ($TR\%$)	**"egtest02"** ($TR\%$)	**"egtest04"** ($TR\%$)	**"egtest05"** ($TR\%$)
Proposed Solution	**98.65**	92.69	73.14	84.52
Mao et al [9]	95.00	**93.02**	60.00	**88.89**
Hasan et al [40]	96.00	92.00	**82.00**	85.50

The "egtest04" test set, provided the most challenging scenarios for all works, however, the proposed solution still performs well in relation to the related works.

4 Conclusion

The paper presents an approach for tracking a selected ground based vehicle from UAV video streams using GMM classification with a Kalman filter. In this study, the GMM classification is simplified with the use of the likelihood function, while the GMM process is improved with the Figueiredo and Jain algorithm. The algorithm only requires the minimum and maximum number of Gaussian components to be initialised. If this is not chosen correctly, the performance is affected this is prevented provided that the initialisation is constant throughout all processes. The model update allows the tracker to adapt to changes in the appearance caused by scale, pose variation and illumination. However, too many model instances weakens the performance, as older instances become irrelevant. To overcome this problem a fixed number of instances is chosen. However an optimal solution would be to dynamically change the number of instances depending on the current scenario. The current work performs well and is comparable with related works. Furthermore, it is tolerant of moving background, pose variation, changes in illumination and scale. However arbitrary and abrupt camera motion, out-of-focus video, full occlusion and multiple target interactions poses challenges, as it may lose track. A method to overcome these challenges is to apply global tracking to require lost tracks and/or a method that stores historic model parameters with variance for a re-initialise step. The main contribution is the specification and adoption of GMM classification for local tracking of a user selected ground vehicle from UAV video streams. GMM classification has resulted in a simplified classification and minimises the number of required training instances.

5 References

1. Cao, X., et al., *Vehicle detection and motion analysis in low-altitude airborne video under urban environment.* IEEE Transactions on Circuits and Systems for Video Technology, 2011. **21**(10): p. 1522-1533.

2. Kimura, M., et al. *Automatic extraction of moving objects from UAV-borne monocular images using multi-view geometric constraints*. in *IMAV 2014: International Micro Air Vehicle Conference and Competition 2014, Delft, The Netherlands, August 12-15, 2014*. 2014. Delft University of Technology.

3. Jeon, B., et al. *Mode changing tracker for ground target tracking on aerial images from unmanned aerial vehicles (ICCAS 2013)*. in *Control, Automation and Systems (ICCAS), 2013 13th International Conference on*. 2013. IEEE.

4. Siam, M. and M. ElHelw. *Robust autonomous visual detection and tracking of moving targets in UAV imagery*. in *Signal Processing (ICSP), 2012 IEEE 11th International Conference on*. 2012. IEEE.

5. Pokheriya, M. and D. Pradhan. *Object detection and tracking based on silhouette based trained shape model with Kalman filter*. in *Recent Advances and Innovations in Engineering (ICRAIE), 2014*. 2014. IEEE.

6. Mundhenk, T.N., et al. *Detection of unknown targets from aerial camera and extraction of simple object fingerprints for the purpose of target reacquisition*. in *IS&T/SPIE Electronic Imaging*. 2012. International Society for Optics and Photonics.

7. Yang, X. and S. Wang, *Fast deformable structure regression tracking.* IET Computer Vision, 2016. **10**(2): p. 115-123.

8. Zivkovic, Z. and B. Krose. *An EM-like algorithm for color-histogram-based object tracking*. in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. 2004. IEEE.

9. Mao, H., et al., *Automatic detection and tracking of multiple interacting targets from a moving platform.* Optical Engineering, 2014. **53**(1): p. 013102-013102.

10. Rodríguez-Canosa, G.R., et al., *A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera.* Remote Sensing, 2012. **4**(4): p. 1090-1111.

11. Santosh, D.H. and P.K. Mohan. *Multiple objects tracking using Extended Kalman Filter, GMM and Mean Shift Algorithm-A comparative study*. in *Advanced Communication Control and Computing Technologies (ICACCCT), 2014 International Conference on*. 2014. IEEE.

12. Abdulrahim, K. and R.A. Salam, *Traffic Surveillance: A Review of Vision Based Vehicle Detection, Recognition and Tracking.* International Journal of Applied Engineering Research, 2016. **11**(1): p. 713-726.

13. Teutsch, M. and W. Krüger. *Detection, segmentation, and tracking of moving objects in UAV videos*. in *Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on*. 2012. IEEE.

14. Cao, X., et al., *Ego motion guided particle filter for vehicle tracking in airborne videos.* Neurocomputing, 2014. **124**: p. 168-177.

15. Chen, X. and Q. Meng. *Robust vehicle tracking and detection from UAVs*. in *Soft Computing and Pattern Recognition (SoCPaR), 2015 7th International Conference of*. 2015. IEEE.

16. Grabner, H., M. Grabner, and H. Bischof. *Real-time tracking via on-line boosting*. in *BMVC*. 2006.

17. Babenko, B., M.-H. Yang, and S. Belongie, *Robust object tracking with online multiple instance learning.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011. **33**(8): p. 1619-1632.

18. Zhang, K. and H. Song, *Real-time visual tracking via online weighted multiple instance learning.* Pattern Recognition, 2013. **46**(1): p. 397-411.

19. Zhang, K., L. Zhang, and M.-H. Yang. *Real-time compressive tracking*. in *European Conference on Computer Vision*. 2012. Springer.

20. Zhang, K., L. Zhang, and M.-H. Yang, *Fast compressive tracking.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014. **36**(10): p. 2002-2015.

21. Reynolds, D.A., T.F. Quatieri, and R.B. Dunn, *Speaker verification using adapted Gaussian mixture models.* Digital signal processing, 2000. **10**(1): p. 19-41.

22. Santosh, D.H.H., et al., *Tracking Multiple Moving Objects Using Gaussian Mixture Model.* International Journal of Soft Computing and Engineering (IJSCE), 2013. **3**(2).

23. Lee, D.-S., *Effective Gaussian mixture learning for video background subtraction.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005. **27**(5): p. 827-832.

24. Mukherjee, D., Q.J. Wu, and T.M. Nguyen, *Gaussian mixture model with advanced distance measure based on support weights and histogram of gradients for background suppression.* IEEE Transactions on Industrial Informatics, 2014. **10**(2): p. 1086-1096.

25. Zhou, D. and H. Zhang. *Modified GMM background modeling and optical flow for detection of moving objects*. in *2005 IEEE International Conference on Systems, Man and Cybernetics*. 2005. IEEE.

26. Xue, K., et al., *Panoramic Gaussian Mixture Model and large-scale range background substraction method for PTZ camera-based surveillance systems.* Machine vision and applications, 2013. **24**(3): p. 477-492.

27. Fradi, H. and J.-L. Dugelay. *Robust foreground segmentation using improved gaussian mixture model and optical flow*. in *Informatics, Electronics & Vision (ICIEV), 2012 International Conference on*. 2012. IEEE.

28. Quast, K. and A. Kaup, *Shape adaptive mean shift object tracking using gaussian mixture models*, in *Analysis, Retrieval and Delivery of Multimedia Content*. 2013, Springer. p. 107-122.

29. Kim, J., Z. Lin, and I.S. Kweon, *Rao-Blackwellized particle filtering with Gaussian mixture models for robust visual tracking.* Computer Vision and Image Understanding, 2014. **125**: p. 128-137.

30. Xiong, G., C. Feng, and L. Ji, *Dynamical Gaussian mixture model for tracking elliptical living objects.* Pattern Recognition Letters, 2006. **27**(7): p. 838-842.

31. Permuter, H., J. Francos, and I. Jermyn, *A study of Gaussian mixture models of color and texture features for image classification and segmentation.* Pattern Recognition, 2006. **39**(4): p. 695-706.

32. Permuter, H., J. Francos, and I.H. Jermyn. *Gaussian mixture models of texture and colour for image database retrieval*. in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*. 2003. IEEE.

33. Tao, J., et al., *A study of a Gaussian mixture model for urban land-cover mapping based on VHR remote sensing imagery.* International Journal of Remote Sensing, 2016. **37**(1): p. 1-13.

34. Jyothi, T.N., S. Vasavi, and V.S. Rao. *Moving object classification in a video sequence*. in *Advance Computing Conference (IACC), 2015 IEEE International*. 2015. IEEE.

35. Ibraheem, N.A., et al., *Understanding color models: a review.* ARPN Journal of Science and Technology, 2012. **2**(3): p. 265-275.

36. Moon, T.K., *The expectation-maximization algorithm.* IEEE Signal processing magazine, 1996. **13**(6): p. 47-60.

37. Figueiredo, M.A.T. and A.K. Jain, *Unsupervised learning of finite mixture models.* IEEE Transactions on pattern analysis and machine intelligence, 2002. **24**(3): p. 381-396.

38. Deng, S., et al., *An infinite Gaussian mixture model with its application in hyperspectral unmixing.* Expert Systems with Applications, 2015. **42**(4): p. 1987-1997.

39. Collins, R., X. Zhou, and S.K. Teh. *An open source tracking testbed and evaluation web site*. in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*. 2005.

40. Hasan, M., *Integrating Geometric, Motion and Appearance Constraints for Robust Tracking in Aerial Videos.* 2013.

Part III

Conclusion

1. Conclusion

Detection, tracking and classification are especially useful and challenging in Unmanned Aerial Vehicle (UAV) based surveillance systems due to the wide surveillance scope and mobility of the platform. Previous solutions have addressed the challenges with complex classification. Therefore GMM based classifiers have been applied to simplify the process. Data are represented in lower dimensionality as model parameters and classification is performed on the parameter-space instead of actual data. The specification and adoption of GMM based classifiers on the UAV visual tracking feature space formed the principal contribution of the work. This was achieved with two main contributions in the form of submitted ISI accredited journal papers.

The first paper demonstrated objectives with a vehicle detector incorporating a two stage GMM classifier applied to a single feature space, namely Histogram of Oriented Gradients (HoG). The first stage initially detects potential vehicle ROIs using the HoG-corner feature space; and then passes them to the second stage classifier which validates the detections while reducing false positives using the HoG-edge feature space. The training process highlighted the sensitivity of training data and class configuration. Therefore, multiple configurations were explored to find potentially optimal solutions. Overall, the detector has proven to be tolerant to moving background, changes in illumination, and target occlusion. Unrestricted pose variation is compensated for by including different vehicle orientations in the training data. Abrupt camera motion and out-of-focus video caused a high number of FNs, indicating that shape features are not tolerant of low contrast between objects and background. The proposed method performed well in comparison to related works in terms of Detection Rate, but falls slightly short for False Alarm Rate. However, this can be improved with better training data and additional classes for vehicles.

The second paper demonstrated objectives with a vehicle tracker using colour histograms (in RGB and HSV), with Gaussian Mixture Model (GMM) classifiers and a Kalman filter. GMM classification was simplified with the use of the likelihood function, while the GMM process is improved with the Figueiredo and Jain algorithm. The algorithm only requires the minimum and maximum number of Gaussian components to be initialised. If this is not chosen correctly, the performance is affected. This is prevented provided that the initialisation is constant throughout all processes. The model update allows the tracker to adapt to changes in appearance caused by scale, pose variation and illumination. However, too many model instances weakens the performance, as older instances become irrelevant. To overcome this problem a fixed number of instances is chosen. However an optimal solution would be to dynamically change the number of instances depending on the current scenario. The current

work performs well and is comparable with related works. Furthermore, it is tolerant of moving background, pose variation, changes in illumination and scale. However, extreme circumstances such as arbitrary and abrupt camera motion, out-of-focus video, full occlusion and multiple target interactions still pose challenges, and result in loss of track. A method to overcome these challenges is to apply global tracking to re-acquire lost tracks and/or to broadly apply historic model parameters that span the scope of object variance in a re-initialise step.

GMM classification has resulted in a simplified classification that minimises the number of required training instances and reduces the dimensionality of the problem representation.

2. Future Work

Both Paper A and Paper B, address issues within the tracking problem for aerial platforms. In the tracking domain for such platforms, tracking alone is not sufficient. Detection and classification assists in reducing the search space, establishment of knowledge priors and building of detailed representations. This improves performance and robustness as shown in the existing works. Detection and classification are addressed in Paper A, while Paper B addresses tracking with classification. The test evaluation from both papers demonstrates the use of GMM classification in different type of scenarios for objects with various appearances and behaviour. In addition, the evaluation of different feature sets provide useful information about the features behaviour and performance. The results highlight the benefits and shortfalls of each feature set across various scenarios, and shows the need to combine features to improve performance. The papers reveal the benefits of GMM classification and show how it can be used for different parts of the problem. Furthermore, different types of GMM classifiers are developed; offline learning for global surveillance in Paper one and online learning for local surveillance in Paper two. A combination of the two methods can form a stronger overall system with both global and local surveillance. This can be used to form either a two-mode tracker or tracking through detection methods. The appropriate combination of methods may offer benefits because each on its own addresses distinct aspects of the problem.