Surveillance Alarm Making

JUN SHEN

A thesis submitted to Auckland University of Technology

in partial fulfillment of the requirement for the degree of

Master of Computer and Information Science (MCIS)

2017

School of Engineering, Computer and Mathematical Sciences

Abstract

Computer vision based surveillance systems have become increasingly important to society. This thesis presents a new approach for computer vision based alarm making systems which detect abnormal events in fixed camera circumstances. The approach contains four functions: namely (1) detecting, (2) tracking, (3) recognizing and (4) alarming. In line with these functions, based on the results of detecting, tracking and recognition, the system will be able to generate alarms automatically. Through the experiments, the related methods and algorithms applied to the proposed approach provide better performance for the purpose of alarm making, thus it could be helpful in reducing the manual labor of security staff. The contributions of this thesis are: Firstly, the shortcomings and deficiencies of the traditional surveillance and alarm systems have been studied. Secondly, computer vision techniques have been utilized to allow the system to work with different environments. Thirdly, dual artificial neural networks have been innovatively deployed for abnormal events detection and to improve the accuracy of alarming to reduce false alarms. The overall result for the false alarm rate of the system developed in this project is 13.8% which is lower that the mainstream 15.27% and also helpful for the management of traffic environments. In future, the improvement of the system will be the working direction for the researcher such as using more training datasets to make the abnormal events alarming system more efficient in terms of abnormal event detection and reduction of the false alarm rate.

Keywords: intelligence surveillance, alarm, traffic, Kalman filter, GMM, ANN, HOG, LBP, INRIA

Table of Contents

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly define in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Signature  Date 17/02/2017

Acknowledgments

I would like to express my lofty respect and sincere gratitude to my primary supervisor Dr. Wei Qi Yan. Dr. Yan owns profound knowledge, rich experience, a responsible teaching attitude and a rigorous scientific research attitude as well. During the thesis study, he always gave me careful guidance and meticulous help for both my study and my life. My achievements are inseparable from his supervision. I also thank my second supervisor Dr. William Liu. This thesis would not have been completed on time without his encouragement.

I would also like to thank my friends for their companionship. They have accompanied me through the wonderful time of this period of study. I hope our friendship will last forever.

Finally, I would like to thank my parents and family for their care and support. They always gave me the confidence and courage to face difficulties and meet challenges, especially during this period of study time. I will always love them.

Chapter 1
Introduction

The first chapter of this thesis presents and elaborates the research question and issues related to alarm making included the research process and analysis. The background of the research topic and the motivation are introduced. Also, the objective of the system developed in this research project is described. The last part of this chapter shows the structure of the thesis.

1.1 Background and Motivation

The surveillance system is one of the most widely used systems in public security. Many different types of sensor have been designed, manufactured and implemented in the public environment to capture the sensory data of our living environment to meet the demand of surveillance by our communities. The most common way is to use a camera to capture the video information for a monitored area (Steenweg et al., 2016). The video information was captured by the camera, transmitted by an analog signal through a cable and stored on tapes in the early stage of video-based surveillance system development. This solution of a video-based surveillance system is called Closed Circuit Television (CCTV) (Kruegle, 2011). The limitations of this solution are very apparent. Firstly, it requires a high cost for the implementation of the cable laying. Secondly, there is a lack of management mechanism for video information which results in limited capacity and poor-quality playback. Finally, it requires a human to sit in front of the monitor to watch for changes in the monitoring area. As a result, many screens must be implemented to display the information captured by the cameras – a waste of social resources. Also, security guards must face a wide range of monitor screens (Tseng, Lin, & Smith, 2002), and this goes far beyond acceptable for humans. Lee, Romano, and Stein (2000) showed that the human eye will ignore more than 95% of moving objects when the person is looking at a video on a screen for more than 22 minutes. Due to these limitations, using computers to replace humans for surveillance has become increasingly important. Surveillance systems are required to continuously conduct the work of monitoring for our living environment 24 hours a day and seven days a week. The task of event detection from surveillance video is to detect and identify the patterns in the video automatically and continuously. Hence, the patterns will be generated from these repeating works and embodied through different types of events

(Luckham, 2002).

For detected events, especially for abnormal events, the system must have an effective way to send a message to the users and let them know something has happened. In general, an alarm signals the occurrence of some undesirable event (William, 1992). The alarm system is a critical component of the surveillance system consisting of various sensors, detectors and actuators to contribute to monitoring a region and make sure of security of the monitored region. According to the data which are collected by the different types of sensors from the natural environment, an alarm system is able to determine whether to trigger an alarm or not (Ding, Cooper, & Pasquina, 2011). Also, the alarm system can produce different types of alarms to alert the users if any monitored parameters are out of range. In this thesis, the camera has been adopted as the sensor to capture the event information and decide whether to trigger the alarm or not. The alarm making based on processing and analyzing of surveillance video is mainstream for the intelligence surveillance system because a surveillance system should be able to detect events from the video information collected by a surveillance camera. Therefore, the prerequisite of the event detection is to detect objects moving in the view of a surveillance camera. Hence, moving object detection is a key technology of video processing (Hampapur et al., 2005). The technology of moving object detection includes the theoretical knowledge of multiple disciplines such as image processing, signal processing, pattern recognition, control theory, biology and so on. Some methods can be applied for moving object detection such as background subtraction, temporal differencing and optical flow (Chien, Ma, & Chen, 2002). Optical flow is the pattern of apparent motion of objects and surfaces, and Gibson first introduced edges into the perspectives of two-dimensions and the concept of optical flow in 1950 (Horn & Schunck, 1981). The method of optical flow is to map the moving objects and background from

the three-dimensional space of a two-dimensional scene. (Beauchemin & Barron, 1995). Temporal differencing method finds the difference between two or three consecutive frames in a surveillance video for moving object detection; therefore, the temporal differencing method is also called a frame differencing method (Zhou & Zhang, 2005). Similar to the temporal differencing method, background subtraction is another moving object detection method to construct a model for background without any moving object in this background model and then subtract the current frame and background model to detect the moving objects in the current frame. (Shaikh, Saeed, & Chaki, 2014).

The other necessary technology applied to this system is called multi-object tracking. The surveillance system will be able to detect and recognize an event or a behavior of one or more moving objects in the surveillance video based on the technology of multi-object tracking. The goal of multi-object tracking is to detect the location of moving objects in each frame of the video sequence (Wang, Doherty, & Van Dyck, 2000). There is a multi-object tracking algorithm based on the Kalman filter (Zhou, Wu, & Zhu, 2016) which can improve multi-object tracking when some static objects cover the target. The algorithm requires object detection using the background subtraction method based on the Gaussian Mixture Model (GMM) and then establishes a relational matrix to matching the objects by local spatial correlation in each frame.

Artificial Neural Networks (ANN) simulate the biological neural networks of humans or animals, and it will be applied to this thesis for the recognition of moving objects. The multilayer neural network is an attractive solution for the application of neural networks currently. Most neural network models and algorithms are based on multilayered structures. The BP neural network is one of the most popular algorithms using multilayered structures. It was developed

by a team of scientists in 1986 (Møller, 1993) and is widely used in many fields, especially in the area of intelligence surveillance.

According to the researching and analyzing for the methods and algorithms introduced above, the system will be able to make alarms for traffic surveillance with a better performance to improve the security of the traffic environment. Therefore, the research question of this thesis is established as What kind of computer vision techniques and methods are eligible to be utilized for the implementation of a computer vision based alarm making system in the traffic scene?

1.2 Objectives

The focus point of this thesis is the methods and techniques of computer vision related to pedestrian detection and alarm making. Based on the theories and principles related to the project, the possible methods and algorithms are implemented, tested and evaluated in this thesis.

The thesis intends to develop a computer vision based alarm making for the traffic environment and improve its security. The system developed in this thesis should be able to detect moving objects in surveillance and distinguish moving objects. In light of the position relationship between pedestrians and vehicles, once an abnormal event has been detected, it may trigger an alarm to notify its user. The working process can be defined as follows: firstly, the inputted video will be pre-processed such as image enhancement, binarization and so on, then the moving objects will be detected from the background and assigned to tracks. For each detected object, the feature will be extracted and imported into ANN for recognition. Based on the results and predefined rules, the abnormal event could be detected and corresponding alarms could be generated.

Finally, the techniques and methods applied to the experiment will be compared and evaluated. The testing results for each part will also be discussed and compared.

1.3 Contributions

There is a new schema presented in this thesis for the implementation of a computer vision based alarm making system. The system is able to detect moving objects and track their trajectories. For the detected objects, the trained artificial neural network is utilized to verify the type of moving object and, based on the recognition result, to make alarm for abnormal events. In terms of the procedures provided in this thesis, the system can be developed step by step for detecting abnormal events in the video source mentioned in the previous section. There are four parts contained in the system proposed in this thesis, and each part is responsible for different tasks such as moving object detection, moving object tracking, pedestrian recognition and alarm making. There are a number of methods and techniques which can be applied to these four parts of the system. The details of the methods and techniques for the implantation of these four parts in the system will be described in a later chapter.

In order to get a better result implemented in this thesis, related methods and algorithms have been analyzed. The advantages and disadvantages of the related methods and algorithms are also compared in this thesis. Because the system is a multiple objective oriented system, the segmentation of objects from the video will have an effect on alarm making. Therefore, the segmentation is also a focus point of this thesis to improve the result of alarm making, and it will be described in the chapter on experiment design.

Furthermore, there is an improvement for the using of the artificial neural networks in this experiment. Due to the shortcomings of artificial neural networks, it is very hard to implement one artificial neural network to satisfy the requirement for multiple object recognition. In overcoming the shortcomings of artificial neural networks, two artificial neural networks have been utilized. The recognition objects are grouped into two classes: pedestrians and vehicles. The two artificial neural networks are trained separately and work together to improve performance. The details of the training and implementation of these artificial neural networks will be described in the chapter on experiment design and will be compared with other methods in the results and analysis chapter.

Lastly, the implementation of the background modelling, Kalman filtering for tracking and recognition of moving objects using the artificial neural networks in Matlab are also provided as the future work of this thesis.

1.4 The Structure of This Thesis

The structure of the thesis will be introduced in this section. There are six chapters in this thesis, the contents of each being shown as follows:

Chapter 2 provides a systematic knowledge structure to support the research and development of this thesis. There is some literature related to the objective of the thesis including moving object detection, multi-object tracking and artificial neural networks. Also, reports of other researchers will be reviewed in this chapter.

In Chapter 3, the methodology of this thesis will be explained. The chapter firstly introduces the limitations and issues of the current surveillance system and then presents the research question of the thesis, and the potential solution

to the research question is provided. Useful resources including the datasets for the experiments will be also described in this chapter.

In Chapter 4, the details of the experiment will be presented. The implementation of the experiment will be addressed in a number of steps. The ANNs for alarming will also be demonstrated in this chapter. According to the details of each step, the experiment will be able to be implemented.

In Chapter 5, the testing data and results developed in the experiment are described and analyzed. Specifically, the frame differencing and GMM (Gaussian Mixture Model) for moving object detection have been compared. The features HOG (Histogram of Oriented Gradient) and LBP (Local Binary Pattern) for pedestrian recognition are also compared. The solutions applied to achieve the objectives will be also discussed in this chapter.

In Chapter 6, the limitations of the experiment will be described and future work will be suggested.

Chapter 2
Literature Review

This chapter provides a systematic structure to support the research and development of this thesis. It shows an understanding of the literature relevant to computer vision, decision making and machine learning. Studies of other researchers related to the issue of alarm making will also be reviewed in this chapter.

2.1 Event Detection

The primary mission of an intelligence surveillance system is to detect events from the surveillance video automatically and continuously. Before the researchers detail the concept event, they have to understand what an event is. Generally speaking, an event is something that happens in the real world. It can be a concert or a ball game. However, the concept of an event in the field of scientific research is not as simple as in the general sense. Accordingly, the conventional definition of an event is an abstract symbol of happenings in the dimension of time and space (spatio-temporal) of the real world (Xie, Sundaram, & Campbell, 2008). Under the framework of spatio-temporal, the objects in the reality can be known as entities which include humans, animals, cars and so on. All objects in the real world are able to trigger the event (Snoek & Worring, 2005). The entities are able to trigger a meaningful event through two ways: changing the status of the objects, and movement of an object in the real world (Yan & Weir, 2011).

According to the information recorded by the attributes of an event, the detected events are classified into different classes from the videos in surveillance. Based on the categories of an event, the detected events will be grouped as normal events and abnormal events. In the field of modern intelligence surveillance, detected events can be arranged in categories of normal events and abnormal events in line with how often a kind of event occurs by attributing analysis of events. If an event happens frequently, then assume that this kind of event is a regular event (Cong, Yuan, & Liu, 2011). Usually, an ordinary event itself is harmless for our human and living environment. Therefore, abnormal events are more concerning in surveillance. The definition of an abnormal event in the Oxford English Dictionary is that "deviating from the ordinary type, especially in a way that is undesirable or

prejudicial; contrary to the normal rule or system; unusual, irregular, aberrant" (Yin, Liu, & Mao, 2015). According to Kwon and Lee, an abnormal event is relative to a normal event, abnormal event is the irregular event from the normal event, and the occurrence frequency of an abnormal event is much lower than the normal event, but resulting in a high causality (Kwon & Lee, 2012). Due to its low occurrence frequency, it is very difficult to describe the details of an abnormal event. An abnormal event can be subtle, and it is difficult to predict the occurrence of an abnormal event. In order to describe and detect an abnormal event, the abnormal event is described based on normal events. (Huang et al., 2007) Because an abnormal event is significantly different from a normal event and is associated with activity and also normal events have commonality in terms of behavior. So, abnormal events can be described by normal events.

Detection of an abnormal event by video surveillance plays a pivotal role in the field of modern intelligence surveillance. Events will be able to be detected by extracting the features in the dimension of spatio-temporal (Romer, 2006). Different features can be applied to event detection from a surveillance video such as trajectory and time (Radinsky & Horvitz. 2013). A trajectory is the path of a moving object through space as a function of time. The position of a moving object in the video source is constantly changing with the changing of time during the movement. The event triggered by the moving of an experimental object is detected using the trajectory (Shotton, Rodrigues, & Trelles, 2000). The features are modelled by the Hidden Markov Model (HMM) to improve the detection of abnormal events (Jiang, Wu, & Katsaggelos, 2007).

Based on the model of a normal event group, the model can be applied to testing video data. The trajectory feature of a pedestrian in the testing video data will be extracted from the video frames and compared with the model of

the normal event group. The compared result will be recorded as the likelihood, in line with the similarity between the features extracted from the testing video and normal event group to define the event in testing, whether the video is a normal event or an abnormal event (Rock, 1986). Moreover, the time interval is also an important feature for event detection.

Due to these facts, it not only facilitates the surveillance system to record, but also allows us to conduct further analysis for the detected events like alarm making. Alarming is an important process based on detected events (Pustejovsky et al., 2003). When an abnormal event has been detected, the surveillance system should provide a warning to the user. The content of surveillance alarm making will be described in detail in a future section.

2.2 Surveillance Alarms

Alarms are widely implemented in the field of intelligence surveillance to improve the security of commercial, industrial, military and personal protection (Weber, 1985). Based on the processing and analyzing of a surveillance video, computers are able to detect events contained in the video source and generate alarms through an alarm system. This mechanism is called event-driven alarm making (Bandini, Bogni, & Manzoni, 2002). A number of examples are used for the description of event-driven alarm making such as border intrusion detection, human behavior detection, smoke and flame detection and so on. For intrusion detection, the intrusion of a predefined area by objects like pedestrians or cars will be detected as an abnormal event to trigger an alarm. For human behavior detection, the unusual behaviors of human targets such as falling, a sudden change in walking speed and hovering in the detecting area will be detected as an abnormal event and trigger an alarm. Smoke and flame detection can be applied in scenarios like a warehouse, museum, forest and so on. The alarm will be triggered by the appearance of

smoke or flame in the surveillance video.

According to the types of triggering conditions, alarms can be grouped into three categories: rule based alarms, probability based alarms and system based alarms (Lomi, Tonetto, & Vangelista, 2003). The rule based alarm is a relatively simple manner for alarm making. It is represented as "IF <condition> THEN <alarm> END" (Gomez & Dasgupta, 2002). The rule based alarm system is designed to monitor a specific condition to trigger an alarm, but it is inflexible. For example, the alarm system is set as someone opens the door, to make an alarm. The original intention is to prevent strangers, but how about when the person who opens the door is the owner of the property? So, the shortcomings of rule-based alarms will be considered in this project. The probability based alarm adopts the method of statistics; it generates an alarm by calculating the expectations of risks or errors (Huanxiong et al., 2011). For system based alarms, a type of system based alarm is the event-driven alarm. System based alarms such as event-driven alarms are able to generate the alarm in line with a rational analysis of different events. Therefore, it will be the developing trend of alarm making. The alarm can be released through a different way of expression. A solution in traditional surveillance alarm making applied lights, sirens and a loudspeaker to inform the occurrence of events (Withington, 1999). With the development of communication networks, people expect to receive the alarm promptly when something happens, no matter whether they are on the site or not.

Whether to generate the alarm for a detected event is an acute problem that needs to be handled seriously during the design processes of an alarm system. According to the demand of alarm making, decision making was used as a mechanism to improve the logic of alarm making (Artikis et al., 2014). Decision making is an interdisciplinary approach. A simple description of

decision making in the field of management is that decision making is an approach for selecting a feasible solution from two or more options.

In intelligence surveillance, a decision will be made through a relational network and the Decision Tree is a good example to describe the application of simple decision making in the intelligence surveillance. A Decision Tree is a decision support tool that uses a tree-like graph or model for decisions and their possible consequences, including chance event outcomes, resource costs, and utility (Kohavi, 1996). The Decision Tree originated from the 1960s to the late 1970s. Each branch on the tree represents a testing point of the attribute of data, also known as a node. When the testing data arrives at a node, it will predict the result according to the classification criteria (Kohavi, 1996). There are mainly three algorithms existing for Decision Tree algorithms: ID3, C4.5, and C5.0. The basic idea of these three types of Decision Tree algorithms is almost same: to contribute the node and classify the data according to the best result of attributes until arriving at the top of the tree. When the input data are partly random for decision making, the decision making will become complex. For complex decision making, the Markov Decision Processes (MDPs) has been introduced. The MDPs is a modelling tool for decision making based on a mathematical framework which is able to solve the problem of complex decision making (KallenberT, 2000). In addition, artificial neural networks can be applied for decision making and they simulate the process of decision making like a human.

With the development of surveillance systems, their function and applicability of the surveillance system are continuously improving. However, the false alarm is still an intractable problem that effects the accuracy of an alarm system and cannot be avoided. On the other hand, the missing of alarms should be more important to people. The term false alarm exists in many different

applications of alarm systems. The false alarm is triggered by an event other than the predefined trigger event (Hubballi & Suryannarayanan, 2014). The triggering of a false alarm is unnecessary and meaningless for an alarm system. Also, if an alarm system always generates a false alarm, that will cause the system to no longer be trusted by a user (Ray, 2013). A false alarm will not only reduce the trust in the alarm system by people, but also it will be result in the high cost of handling unforeseen circumstances. The United States Department of Justice estimates that between 94% and 98% of all alarms sent to law enforcement are false alarms (Sampson, 2011). Related to false alarms, the missing of alarms results in more serious consequences than false alarms (Alm & Osvalder, 2012). Compared with the high cost resulted by a false alarm, the missing of alarms causes harm to the lives of humans when the alarm system fails at the occurrence of abnormal events. Therefore, the reduction of false alarms to improve the reliability of an alarm system will be a major task for the design of an alarm system (Tibor et al., 2011). Before a solution is developed for the reduction of false alarms and missing alarms, the reasons why false alarms and the missing alarms are caused must be understood.

Numerous reasons will cause false alarms. Firstly, the selection and implementation of the sensor will be one reason for the cause of a false alarm. For example, applied ultrasonic wave sensors in the area have a high probability of generating a false alarm. Secondly, changes of environment such as air flow, illumination and magnetic field also affect an alarm system (Hu & Yi, 2016). Thirdly, the aging and failure of surveillance circuits and facilities are factors for producing false alarms. Finally, the most important reason of the causing for a false alarm is the human factor. Improper operations of a user will trigger the alarm device accidentally (Hu & Yi, 2016). Once the causes of false alarms are recognized, solutions will be able to be developed.

The fundamental method to reduce false alarms and improve the adaptability and reliability of alarm systems, is to improve the performance of sensors (Huicong, 2002). Improving the training of users is also an imperative way for reduction of false alarms. In the technique part, there are some algorithms developed to solve this problem. There is an algorithm (Liang, 2011) called non-linear model (NLM). The mechanism of NLM is that if an alarm system generates too many false alarms, the alarms will be automatically inhibited by the algorithm to reduce the further false alarms. The introduction of the non-linear model can reduce the false alarm rate and the cost of system resource, but the disadvantage is that the NLM results in a high missing report when applied to the mechanism for inhibition of false alarms (Mark, 2005). The linear convergence model (LCM) is a simplified version of the non-linear model and is more efficient for low configuration systems (Xiaoli, 2009). In addition, the method of machine learning such as Naive Bayes, Decision Trees, SVM and ANN can also be applied to reduce false alarms (Baumgartner, Rodel, & Knoll, 2012).

2.3 Artificial Neural Networks

Artificial Neural Networks (ANN) are a series of models to simulate the biological neural networks of human or animals (Yegnanarayana, 2009). The main objective for the development of artificial multilayer neural networks is to simulate the mechanism of the real brain of a human to let a machine be more intelligent by using machine learning (Marcus, 2015). The basic structure of a multilayer neural network is a collection of neurons, each distributed in several levels and connected to construct a layered network structure (He & Xu, 2010). Therefore, ANN is a simulation of the structure and working methods of the real biological brain. The advantage of a working ANN is that the researcher can simply input source data to the network for training and then the result of the analysis will be output without any user

interventions because the ANN will automatically calculate the result on each neuron based on the obtained weights (Hayashi, Setiono, & Azcarraga, 2016). The training process determines the value of the weight on each neuron. Similar to a real brain, ANN requires a large amount of training data and computations. Based on these characteristics, ANNs are able to make complex decisions.

A working artificial neural network requires a training process to achieve full functionality (Dreiseitl & Ohno-Machado, 2002). The training of an ANN is based on inputting of external data to adjust the weight value of each neuron based on the learning rules, and finally to make the network have some expected output behaviors. The most common learning rules are the Hebb learning rule, competitive and cooperative learning, Randomly Connected Learning and so on (Jain, Mao, & Mohiuddin, 1996). Based on the different learning rules, the ANN will be trained.

According to the difference in training environments, the training of an ANN can be divided into mainly three types: supervised learning, unsupervised learning and semi-supervised learning. Supervised learning requires a set of training samples known as a training set. The training set contains the known data and the corresponding outputs. Usually, the known data is stored in a vector and the expected output will be a continuous value or a tag on the vector (Mitchell, 1999). The training data will be input to the ANN in training, and finally, the best model of this dataset will be obtained. During the training process, the error between the output value and the expected output value will be checked. If the value of an error is too big, a new training process will start adjusting the parameters of ANN and the process will be repeated until the error reaches a minimum value.

For unsupervised learning, there is no need for a training sample. The neural network observes the data in the light of the extraction of statistical features and learns by itself. Also, the result will not be checked during the training process.

Semi-supervised learning is a type of learning method between supervised learning and unsupervised learning (Zhu, 2011). For the application of this training method, supervised learning is able to classify the targets to predefined categories, and it is usually applied for pattern recognition. Unsupervised learning is usually applied for clustering by calculating the similarity of targets (Wang, 2003). A number of different models of ANNs have been developed such as Perceptron, BP neural network, Hopfieldd model and ART network.

In recent years, deep learning has become more popular in the field of computer vision (Hinton et al., 2006). The introduction of the deep learning method is based on the research of artificial neural networks. Actually, the deep learning method is a branch of artificial neural networks. For a normal artificial neural network, there are usually three to four hidden layers existing in the network (Murata, Yoshizawa, & Amari, 1994). For a deep learning neural network, the hidden layer of the neural network has the ability to reach eight to ten layers due to the use of special training methods. The root cause of this distinction is the difference in training algorithms. The algorithms developed for deep learning are based on the layer-wise training mechanism (Jin et al., 2000). The reason is the traditional training mechanism for artificial neural networks is based on back propagation. Back Propagation (BackProp) uses the iterative algorithm to train the entire network like chain rules, randomly sets the initial value, calculates the current network output and changes the parameters of the previous layers according to the difference between outputs until convergence. However, with the increasing number of

layers, especially for a deep network which is more than seven layers, the residual propagation to the front of the layers has become indistinguishable. This results in a diffusion gradient to the deep network and affects the accuracy. Similar to other machine learning methods, deep machine learning methods also have supervised learning and unsupervised learning. In accordance with the different learning frameworks, the function and characteristics of the trained network will be also different. For example, the Convolutional Neural Networks (CNNs) is a famous and popular deep learning model based on supervised learning. The Deep Belief Nets (DBNs) is another deep learning model utilizing the unsupervised learning mechanism. With the introduction of deep learning methods, the computer will be able to cope with more complex issues and data in fields like computer vision, decision making, nature language processing and so on.

2.4 Related Work

With rapid growth in the number of vehicles, the safety and effectiveness of transportation environments are also facing enormous challenges. To ensure the safety and effectiveness of transportation systems is a broadly concerned topic is a main objective for the development of an intelligent surveillance system (Lu et al., 2008).

Before proposing solutions to improve the safety of a transportation system, an understanding of the components is necessary. The elements of a transportation system have multiple types. Broadly speaking, any persons or objects that are running on the road can be considered as participants of a transportation system. To be more specific, the participants of a transportation system can be divided into two main categories. One category contains all the pedestrians; the other category comprises the vehicles including cars, trucks, buses and so on. There are a large number of pedestrians and vehicles existing

in the urban traffic environment. This will cause the environment of the transportation system to be more complex with increasing numbers of pedestrians and vehicles. Also, it will cause some problems that impact on the safety and effectiveness of the traffic transportation system by the complexity of the environment. The main problem that impacts on the safety and effectiveness of a transportation system is violation of traffic rules. There are multiple situations being considered such as the traffic violations like running a red light and speeding. The traffic violation is an illegal behavior and it will not only cause traffic accidents, but also it will be result in traffic jams. Therefore, detection of traffic violations would be the core issue of an intelligent surveillance system in the traffic scene. To be more specific, an intelligent surveillance system or a smart transportation system should be able to detect wrong things at a particular time and location. The wrong things are also known as abnormal events, and the detection object will be the participants of a transportation system including both pedestrians and vehicles.

At present, the safety and effectiveness of a traditional transportation system is expected to be improved through technical means. The concept of a smart transportation system has been introduced. The intelligent transportation system (ITS) is a main trend for the development of transportation systems. The intelligent transportation system is able to reduce traffic load and environmental pollution, ensure traffic safety and improve transport efficiency with timeliness and accuracy.

Intelligent transportation can be considered as a refinement branch of intelligent surveillance in the traffic scene. Intelligent surveillance includes a few components to ensure the normal operation of the system. As an important component of intelligent surveillance, alarm making is able to make the alarm respond to those abnormal events which exist in the traffic scene but it is very

hard to develop a system to make all alarms. Therefore, the only solution is to detect abnormal events under predefined rules. In order to provide alarms for all abnormal events, the detection of events has become a key point to achieving the goal (Krumm et al., 2000). As mentioned in Chapter 1, the events are triggered by an object moving or changing status. Thus, the detection of a moving object will be the prerequisite for alarm making.

A number of technologies and methods can be applied for the detection of moving objects based on different sensors such as infrared sensors (IR), radar, sonar, camera and so on. A camera-based surveillance system has advantages in compatibility, feasibility, stability, reliability, robustness, better cost-effectiveness and so on. Based on the advantages of the system, it has gradually become a main trend to achieve the purpose of moving object detection. Some methods can be applied to achieve moving object detection such as background subtraction, optical flow and frame difference (Robert, 2000). In this project, a camera will be implemented as the sensor to capture the video as the input source for experiment. If there are objects moving across the view of the camera, the system will be able to decide and make corresponding alarms and notify the user automatically.

When the system is setup, the video of traffic will be input to the system and the moving objects will be separated from the background (Rui, 2009). A video is an actually a set of static frames, and the location of a moving object should be different in each frame but the location of a static object and the background will be same in each frame. So, the video will be converting into a number of frames and processing each of them. For each frame, the original frame will convert to a gray image by changing the proportion of RGB. The gray image will remove all the color information and contain different levels of gray information from 0 to 255 which is much easier for a computer to process. But

the gray image still has a lot of noise. The noise is blemishes that exist in the images or videos and it is usually caused by electromagnetic interference and bad light conditions. In order to remove noise, the process of image enhancement is required to make the image clearer with better quality. The moving objects will be detected from the preprocessed frames using background subtraction and bounding boxes will be drawn around the detected objects. There is a new method called three-frame difference. It is an improvement of frame difference (Grzegorz, 2004) to improve the result of moving object detection. Moving object tracking methods will also be implemented. The tracking method is to select Kalman filtering to predict the location of a moving object in the next frame based on the information collected from previous frames. The information of trajectory for moving objects will be recorded. From the information of trajectories, the tracked moving objects that include both vehicles and pedestrians will be able to trigger a set of events such as object turn-up and object lost (Koller, Daniilidis, & Nagel, 1993). These events belong to normal events. The important point is to distinguish which object is a car and which is a pedestrian. In order to solve this problem, the feature of a detected object will be extracted and input to an artificial neural network to distinguish the type of moving object by the machine learning method. The artificial neural network will be trained using the training dataset gained from the website of the computer vision organization. The features extracted from the tracked moving object will be compared with the features that are utilized for the training process of the artificial neural network. There are different features extracted from the training sample and testing sample such as HOG and LBP. Therefore, the selection of features and the performance of the classification method will be tested in this experiment. In order to reduce false alarms, it is expected the performance of the sensors will be improved (Huicong, 2002), and some algorithms developed. The non-linear model can reduce the false alarm rate

and also reduce the cost of system resource but will result in a high missing report (Mark, 2005). The linear convergence model is a simplified version of the non-liner model and is more efficient for low-configuration systems (Xiaoli, 2009). In addition, the method of data mining such as Naive Bayes, Decision Trees, SVM and *k*-NN can also be applied for the reduction of false alarms (Baumgartner, 2012). The algorithms and threshold value for triggering alarms will be tested by WEKA in this project.

Chapter 3
Methodology

This chapter provides an articulate description of the methodologies and methods adopted in this thesis. Firstly, the limitations and issues of the current surveillance system are outlined and then the research question for these shortages and issues in this thesis is proposed. Potential solutions to solve the research question are made and related resources are described.

3.1 Research Question

The main objective of this thesis is to develop a computer vision based alarm making approach to improve the safety of vehicles and pedestrians in traffic scenes. The system should be able to detect the events triggered by moving objects in the video and make alarms to notify users when an abnormal event is detected. As a fundamental module, the video processing module is responsible for detection and tracking of moving objects in the video. Based on the results of moving object detection and tracking, the system will be able to recognize the type of moving object through another module called the recognition module. The rate of recognition is very important for effective event detection and alarm making. Finally, the alarming module must be able to work correctly.

From reviewing and learning of computer vision techniques and methods from the literature, there is a clear and systematic overview of the implementation of a computer vision based surveillance application. However, there are so many techniques and methods existing in the field of computer vision, it is important to select the most appropriate and effective one for this project, because the utilization of an appropriate method will not only be able to improve the efficiency of the program but also improve the accuracy of the system. Therefore, the main research question of this thesis is established as:

Question:
What kind of computer vision techniques and methods are eligible to be utilized for the implementation of a computer vision based alarm making system in the traffic scene?

In order to propose a solution to the research question proposed above,

technologies and methods of moving objects detection, especially pedestrian detection, and the related techniques of computer vision and artificial neural networks, should be evaluated and selected before they are implemented to the alarm making system. Therefore, this project intends to analyze and test the relevant methods for the development of the desired system.

3.2 Research Method

The experimentation research method is a main research method for natural science research. The researchers have been able to organize the relationship between theory and practice based on the adoption of an experimentation research method. The experimentation research method is a controllable research method involving the changing of one or more variables to predict the impact on the experimental object. The six main steps for the experimentation research method procedure are as follows:

(1) Identify the research question according to the actual demand and literature review.
(2) Make a logical speculation based on the theory and put forward the hypothesis propositions.
(3) Design the research procedures and methods.
(4) Collect relevant data from the experiment.
(5) Data analysis. Use the relevant data collected from the experiment to test the proposed hypothesis propositions.
(6) Explain the results of the data analysis, summarize the conclusions and suggest further research and improvements.

The advantages of the experimentation research method are mainly in the following aspects: Firstly, the experimental researchers have independent autonomy and are able to decide the study variables, design variables and so on according to the hypothesis made. Secondly, the experiments can take place over time and data can be collected at different times. Thirdly, the results

obtained from the experimentation research method are more convincing and allow the reader to assess the believability of the experiment results. The fourth point is that the experimentation research method is able to effectively control the effect of the variables. Therefore, the experimentation research method has been selected as the research method in this research experiment. In order to facilitate and unify the description of the various experimental designs, the experimental design is often indicated by different symbols.

O——Experimental Object

X——Experimental Variable

Y——Measurement Result

C——Overall Result

Based on the formal research approach explained above, the experiment design in this research is shown as:

The research question is "What kind of computer vision techniques and methods are eligible to be utilized for the implementation of a computer vision based alarm making system in the traffic scene". Against the research question, it is assuming that the artificial neural network is able to improve the recognition result of pedestrian detection and reduce the false alarm rate in the traffic scene. Which means the results of experiment with different parameters will be compared. If C1 < C2, then X1 < X2 the parameter will be more suitable for the experiment. There are two video sources capturing for the experiment and they will be explained in the following sections. Take the video clips as the research objects, named O1 and O2. Apply a set of video processing and morphological processing such as moving objects detection, tracking and separating. The processing result will be input to the different pre-trained artificial neural networks with different layer numbers, training times

and sample numbers. The recognition results will be recorded by the number of frames. For each variable, the result will be recorded as Y, and then the data of Y will be adding together and divided by the number of variables to get an overall result to measure the difference in performance. The calculations of the results for each group in the experiment are shown in the Table 3.1.

Table 3.1 The Wheel Set of Experiments

Objects \ Variables	X1	X2	X3
O1	Y11	Y12	Y13
O2	Y21	Y22	Y23
O3	Y31	Y32	Y33
Overall Result	C1	C2	C3

According to Table 3.1. the results of the experiment indicate the difference between the impact for different X1, X2 and X3 where:

$$C1=(Y11+Y21+Y31)/3$$

$$C2=(Y12+Y22+Y32)/3$$

$$C3=(Y13+Y23+Y33)/3$$

Then compare the data from different groups and the variance analysis on the mean values. The experimentation research method still has limitations for the experiment in this research. Firstly, the results depend on the capacity of the training database: with a different training sample, the result will be different. Secondly, the training process of an artificial neural network is uncontrollable. During the training process, the nodes are self-organized which means a different organization of nodes in an ANN will impact on the data consistency and reduce the conclusion validity.

3.3 Processing Environment

As a foundation and a necessary condition to ensure the smooth progress of the experiment for the system development, the run-time environment of the experiment will be briefly introduced in this section, and it is helpful for measuring the performance. A computer was used as the main equipment in this study. This computer has an Intel Core i7-3720QM CPU installed with quad-core and eight threads which allow multitasking in 2.60 GHz at the default setting and with turbo boost to 3.60 GHz. This ensures the smooth running of the program developed in this project. As a computer vision based experiment, an independent graphic chip allows the program to run much faster than integrated graphic cards. For internal storage, the 8G DDR3 memory will be able to store the temporary data during the running of the program. For external storage, a 250G SSD was installed for the storage of sample data and testing data. The SSD can provide a better access speed for the experiment data. There are mainly three software which will be utilized in this experiment. Developing the platform and testing the software to measure the performance of the computer vision based alarm making system will be taken into consideration in this project. First of all, the computer was installed the Windows 10 Professional Operating System which is an X64 version to allow the installation of our developing software and testing software. Secondly, the popular software, Matlab, will be utilized as the developing platform for our experiments. There are different versions of Matlab available to download and install, but the functions and modules being slightly different between the different vision of Matlab software. In this project, the Matlab R2015b was selected as the developing platform to developing and testing the methods and algorithms that implemented in this thesis. The reason for using the Matlab R2015b in this experiment is that there is a toolbox integrated into this version of Matlab named Computer Vision Toolbox. The Computer Vision

Toolbox is able to provide a set of built-in functions about video and image processing for the developer to use and it is helpful to simplify the developing progress and allow the code to run with higher performance. Also, an Artificial Neural Network Toolbox is allowed to build, train and connect the artificial neural network to the video processing module for recognition. Lastly, in order to test and compare the processing results of the artificial neural network and other methods, another software was utilized as a testing tool in this project, named WEKA. WEKA was born in New Zealand, the full name of WEKA being Waikato Environment for Knowledge Analysis, and it is a popular suite of machine learning software written in Java, developed at the University of Waikato in New Zealand. As a public data mining platform, there are so many algorithms integrated into WEKA for data mining and machine learning tasks. Therefore, the results will be tested and compared with this tool. In addition, there is a Java environment required for the running of WEKA.

3.4 Data Acquisition

The acquisition of the video source and the dataset will be described in this section. There are two videos captured for the experiment of video processing. The two videos captured should meet the requirements for our experiments. The first video sample was captured by the built-in camera of an iPhone 5 mobile phone. The iPhone 5 mobile phone was equipped with an iSight camera which has eight megapixels and an f/2.4 aperture and allows capture of the video at a resolution of 1920 × 1080 (1080P) and a speed of 30 frames per second. The video captured for this part has a person walking across the view of the camera from the right-hand side to the left-hand side with a normal walking speed of about 5 kilometers per hour. The video is captured in the daytime for better illumination conditions, and the background is static without any other moving objects except the walking pedestrian in the video. The camera is also fixed without shaking. Figure 3.1 shows several frames of the

video for the evaluation of moving object detection and alarm making.

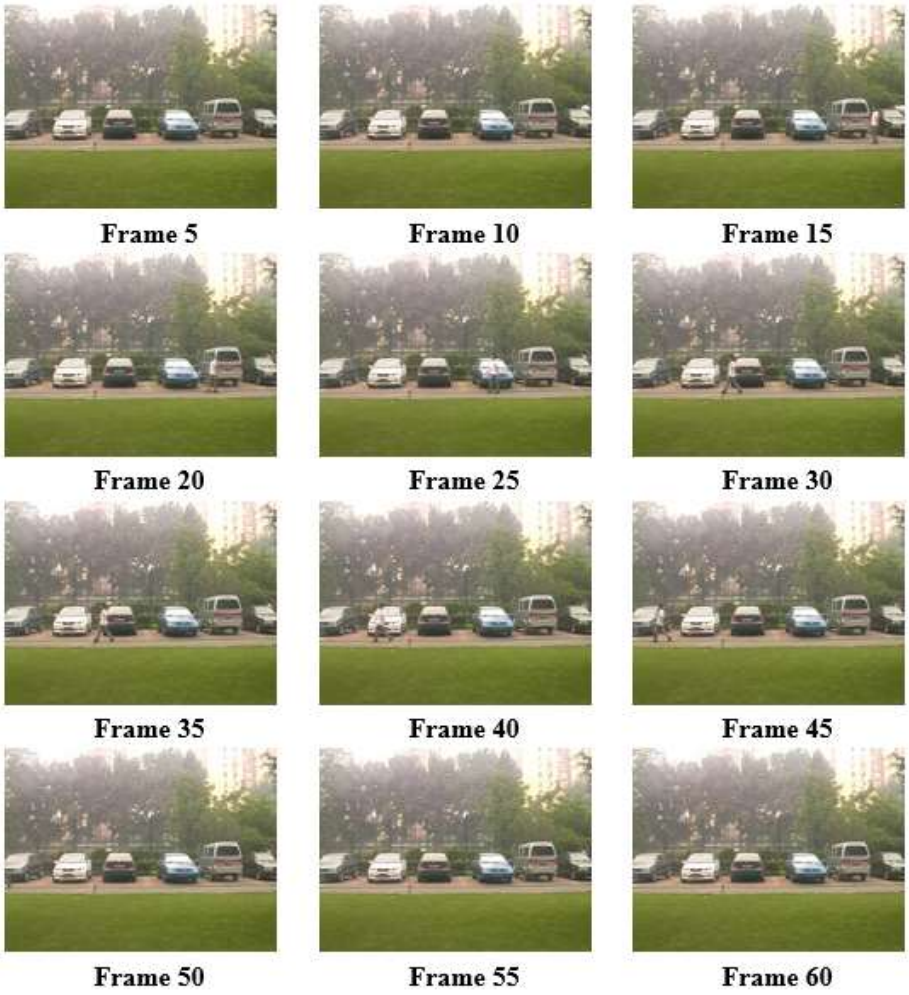

Fig. 3.1 Frames of Sample Video 1

The second video sample was captured by a Canon 5D Mark III SLR camera. The camera was able to capture the video at a resolution of 1920 × 1080 (1080P) and a speed of 25 frames per second. Because this video will be utilized for the moving object tracking, the video has multiple moving objects. The video is captured for an intersection at the corner of a very tall building. There are moving objects in this video including cars, buses and pedestrians. The objects in the video are moving randomly. The vehicles in the video are moving along the road at different speeds and the pedestrians in the video are

moving in all directions. Because the video is captured in a real situation, the vehicles in the video have different motion states including stopping, increasing speed, constant speed and decreasing speed. Also, the shooting direction leads to the vehicles overlapping one and another. The pedestrians in the video also have the opportunities to overlapping with each other. In addition, outside interferences also exist in the video such as swinging trees, flying birds and flags and so on. Figure 3.2 shows the frames of the video for the evaluation of moving object tracking etc.

Fig. 3.2 Frames of Sample Video 2

Another part of the system to be implemented in the experiment is the artificial neural network. An artificial neural network is a powerful machine learning method requiring a vast quantity of data training the network to achieve the

goal of pedestrian recognition. The dataset collected for training and testing the artificial neural network will be described in this section. Plenty of datasets are available and utilized for the pedestrian recognition.

The INRIA dataset was selected for the experiment, and the INRIA person dataset was selected as part of research work on detection of people in images and videos. The dataset of the INRIA person dataset is available to download from the website of INRIA (http://pascal.inrialpes.fr/data/human/), and there are 2436 images collected in the dataset with a size of 980MB. The resolution of all the images has been modified to a size of 64 × 128. The INRIA person dataset contains two groups of images: positive samples and negative samples. Figure 3.3 shows a part of the positive samples in the INRIA person dataset. In the selected examples, there are 614 images and 2416 pedestrians. The pedestrian images collected in this data set are all standing and view angles are various including front, back and from various sides. In addition, the people who are riding a bicycle or skateboard are also grouped into the class of pedestrian.

Fig. 3.3 Example of Positive Sample in INRIA Dataset

Figure 3.4 shows some of the negative samples in the INRIA person dataset. The selected examples include 1218 images, and these images have non-

pedestrian objects such as cars, a toll machine and buildings. The negative samples will be input into the artificial neural network for training.

Fig. 3.4 Example of Negative Sample in INRIA Dataset

3.5 Performance Metrics

Performance is an important measurement of the quality of the approach for developing in the experiment. Usually, the performance and quality of the system cannot be evaluated arbitrarily. On the contrary, rational analysis and evaluation of the results generated in the experiment should be provided. In this experiment, the machine learning methods will be utilized and implemented for pedestrian recognition. Therefore, a confusion matrix in pattern recognition will be selected as the tool to calculate the value of precision, recall and accuracy for the performance evaluation of the system. The details of the confusion matrix will be described in this section.

The experiment will show different results including precision, recall and accuracy to measure the performance of the methods tested in this project for the classification. In order to calculate the value of precision and recall, the confusion matrix will be applied to this classification. As shown in Table 3.2, the confusion matrix is very helpful for understanding and calculating the precision and recall values.

Table 3.2 The Confusion Matrix

	Predict True	Predict False
Actual True	True Positive	False Negative
Actual False	False Positive	True Negative

The confusion matrix shown in the Table 3.2, shows the true positive (*TP*), true negative (*TN*), false positive (*FP*) and false negative (*FN*). The *TP* value predicts true and actually true, the *FP* value predicts true but actually false, the *FN* value predicts false but actually true and the *TN* value predicts false and actually false. Then use eq. (3.1) to calculate the value of precision, eq. (3.2) to compute the value of recall and eq. (3.3) to count the value of accuracy for the performance measurement. In addition, cross-validation is required in this experiment for the validation of accuracy.

$$Precision = \frac{tp}{tp+fp} \tag{3.1}$$

$$Recall = \frac{tp}{tp+fn} \tag{3.2}$$

$$Accuracy = \frac{tp+tn}{tp+tn+fp+fn} \tag{3.3}$$

In order to describe the performance of the system developed in the experiment more intuitively, the *F* value was introduced to this experiment to help us compare this work with others. The *F*-value is calculated as the value of precision and recall obtained from the testing results.

$$F = 2\times\frac{precision\times recall}{precision+recall} \tag{3.4}$$

According to the comparisons between these values, the performance of the

methods tested in this experiment will be easily understood and compared with the work of other researchers.

Chapter 4
Design and Implementation

The implementation of the experiment is presented in this chapter and the research processes of alarm making are demonstrated with originality and innovation. The research process has been conducted in several steps. The ANNs for the abnormal event detection and alarm making approach will be demonstrated in this chapter. By reviewing the details of each step, the experiment results will be reproduced.

4.1 Video Processing

The project detects abnormal events and makes alarms from the surveillance video source. The detection of abnormal events from a surveillance video source is a fundamental process for alarm making. In the light of the correct detection of abnormal events, the accuracy of alarm making for abnormal events will be better. As a computer vision based surveillance system, video processing will be the first component of the system. There are mainly three stages to detect the events from the surveillance video source: video pre-processing, moving objects detection and multiple objects tracking. The details of video processing will be described in this section. By reviewing the details of the implementation described in this section, the experiment will be able to be repeated step by step.

4.1.1 Video Pre-processing

A camera captures the input from the surveillance video source. The details of the camera include the model and resolution, the aperture being introduced in the previous section. Also, the properties of the video files are described. Actually, the video is a collection of a number of frames, and the frames are a static image of the view of the camera at a specific time. When the frames are formed as a contiguous sequence and played continually, the objects captured in the video will be ‘moving’. Based on the characteristics of the video, the computer programs deal with the video file frame-by-frame rather than processing the whole video file. Therefore, the first step of the video pre-processing is to separate the surveillance video into a series of frames and tackle each of them. In this project, the frames from the surveillance video file are also known as the original image; the moving objects will be able to be extracted from the video for event detection. Figure 4.1 shows an example of an original image converted from the surveillance video.

Fig. 4.1 Example of Original Video Frames

The second step for the pre-processing of the frames is to convert a color image to grayscale. The original image is a color image, but our system is based on grayscale imaging to find the location of moving objects. The original intention of the conversion of the color image to a grayscale image is to reduce the amount of data for processing and improve the performance of the program. The weighted average method is a most popular method and the conversion of the RGB color image to a grayscale image by the weighted average method can be defined as:

$$Gray = 0.30 \times R + 0.59 \times G + 0.11 \times B \tag{4.1}$$

For the implementation, the command rgb2gray can be applied in Matlab to complete the conversion. Figure 4.2 shows the gray image converted from the original image in Matlab.

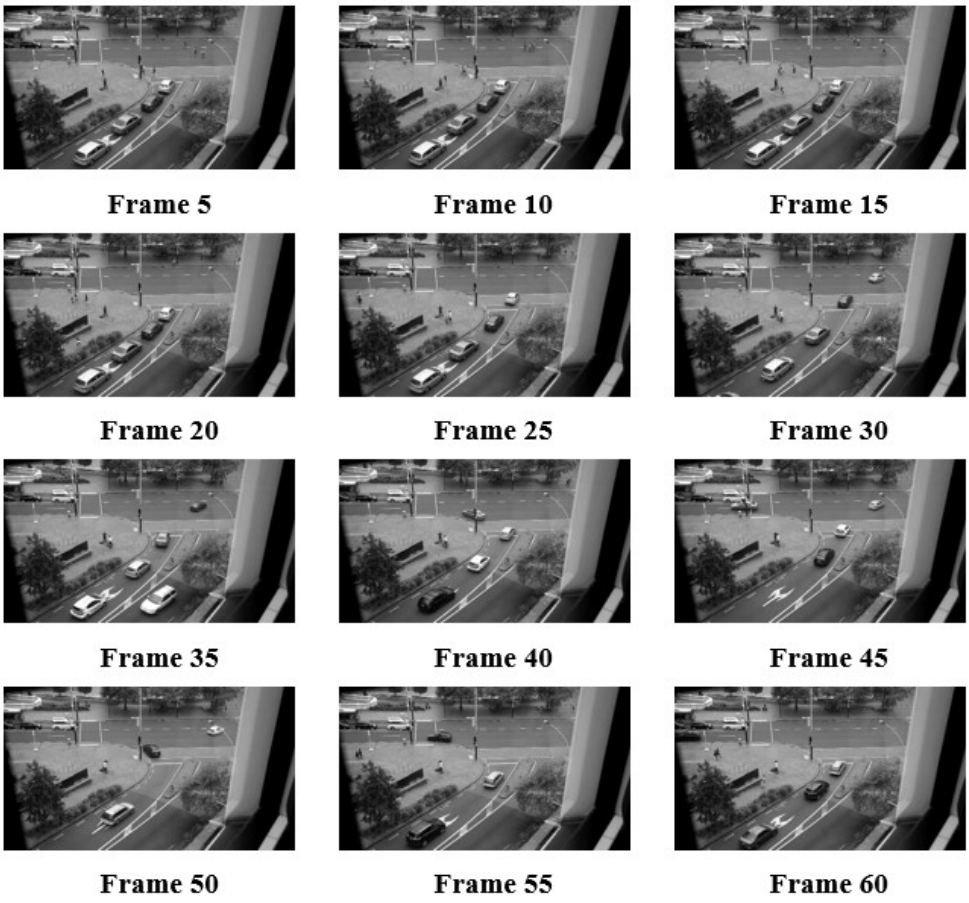

Fig. 4.2 Samples of Gray Scale Frames

In order to remove the noise in the grayscale image and improve the quality, an image enhancement has to be applied to the grayscale image, making the grayscale image clearer.

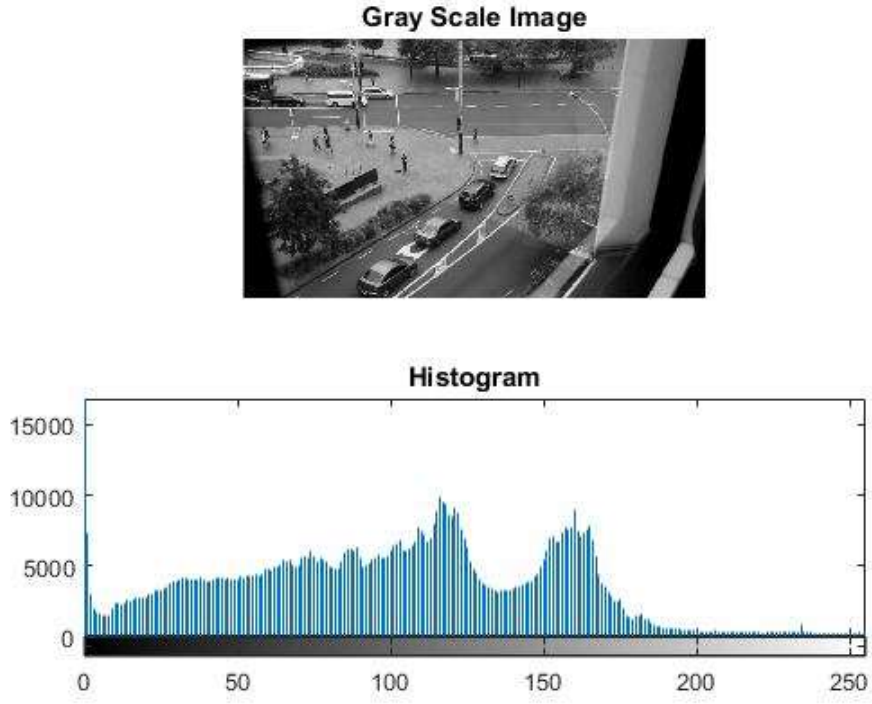

Fig. 4.3 Histogram of Gray Scale Image

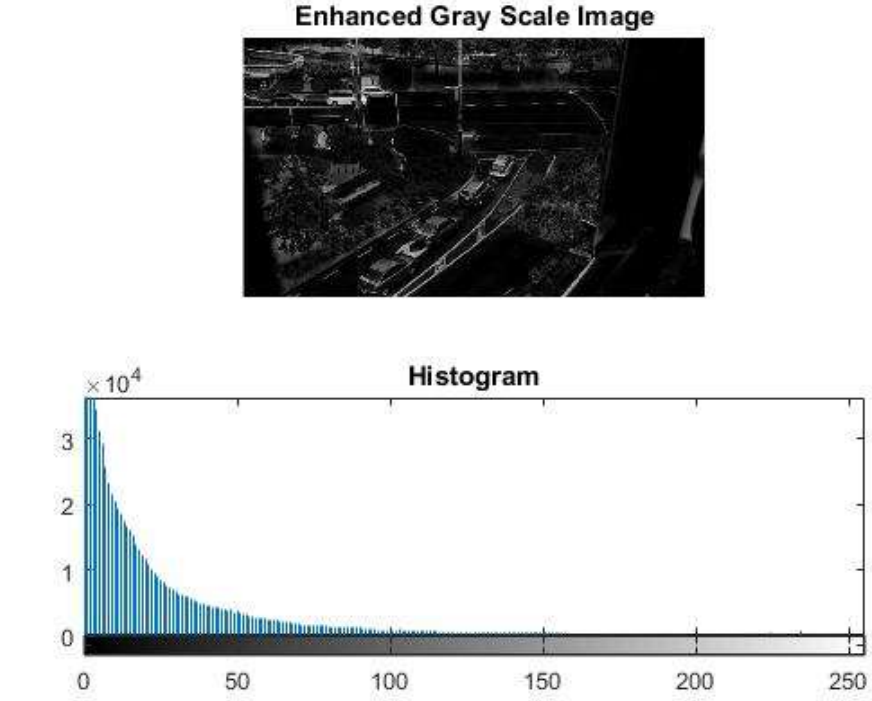

Fig. 4.4 Histogram of Enhanced Gray Scale Image

Based on the grayscale image, a background image was made, and then the

background image and grayscale image were superimposed to get an enhanced grayscale image. The useful information in the enhanced grayscale image will be more prominent than in the original gray scale image. Figures 4.3 and 4.4 show the comparisons between a grayscale image and an enhanced grayscale image. They also show the histograms of the gray scale image and the enhanced grayscale image.

Image binarization is the last step of the video pre-processing. Image binarization is processed based on the gray scale image. The gray level of a pixel in the grayscale image is set between 0 and 255, where 0 represents black, and 255 represents white. Any other levels from 1 to 254 are represent by a different level of gray. The process of binarization is to make a boundary between white and black. The calculation of threshold value for binarization is to add up the highest and lowest gray scale value of an image and divided by two. The binarization is helpful in making the bright object clearer in the image and remove the dark objects. The bright object has a higher probability to represent the moving objects which is helpful for the moving objects detection in the next step. Figure 4.5 shows an example of an image after binarization processing.

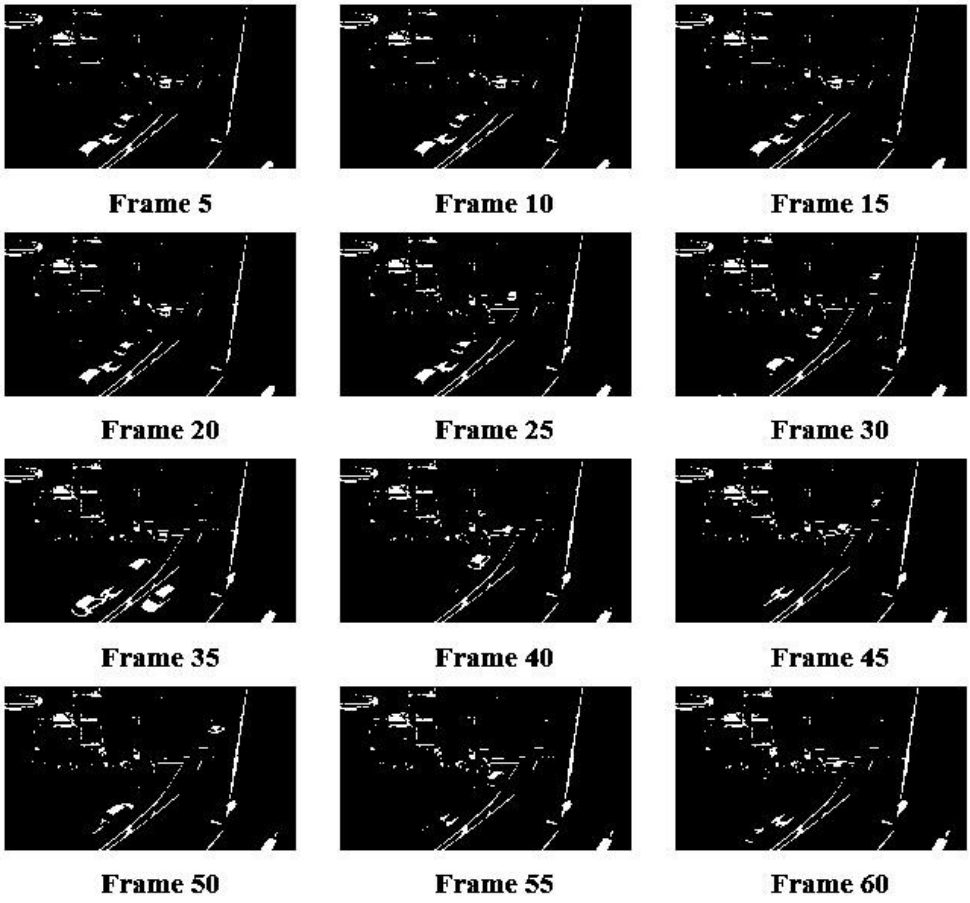

Fig. 4.5 Samples of Binary Frames

4.1.2 Moving Objects Detection

Events can be triggered by moving objects. Therefore, in order to detect the events that are contained in the surveillance video, moving objects should be extracted from the surveillance video first.

Background differencing requires modelling of the background image (Liu, Ai, & Xu, 2001). Based on the different modelling methods, the researchers have presented a large number of methods for the process of background differencing: average model, median model, Single Gaussian Model, Gaussian Mixture Model and so on (Li et al., 2003). The Gaussian Mixture Model (GMM) was selected as the background modelling method for moving objects detection, also known as foreground detection. GMM is able to create a

number of models to represent the features of each pixel. The principle of the GMM based foreground detector is, for foreground detection, each pixel in the current frame will be matched with each pixel in the GMM, and these two pixels should have exactly the same position (Mittal & Paragios. 2004).

In Matlab, there is a foreground detector built into the Computer Vision System Toolbox for moving objects detection based on GMM. Rather than immediately processing the entire video, the foreground detector requires a certain number of video frames to initialize the GMM. The first ten frames were selected as the training frame to initialize the GMM which is helpful for video processing to get better performance (Piccardi, 2004). The number of the GMM was set at five which is the most popular parameter for background modelling by a GMM (Pilet, Strecha, & Fua, 2008). The procedures of GMM moving objects detection can be described as:

1) Feature extraction from a number of training frames,
2) Application of extracted features for background modelling,
3) Feature matching between the current frame and GMM,
4) Foreground detected and displayed.

$$\frac{|x - \mu_i|}{\sigma_i} < K \tag{4.2}$$

In the course of object detection, the background and static objects will be removed from the video, and the moving objects including pedestrians and vehicles will be shown in the video. Figures 4.6 and 4.7 show a comparison between the detected foreground by frame differencing and the GMM. Based on the detected foreground, further processing can be carried on.

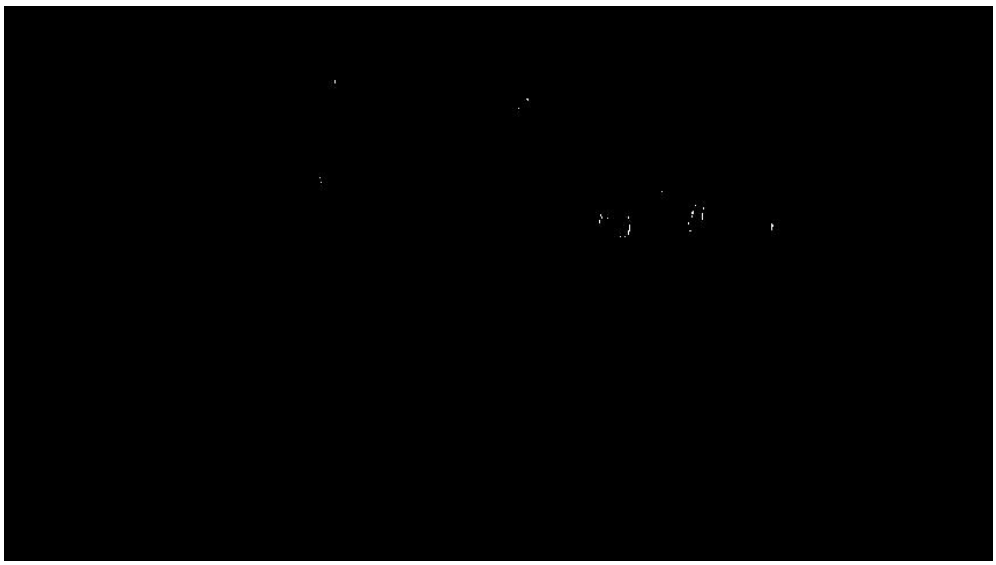

Fig. 4.6 The Detected Foreground by Frame Differencing

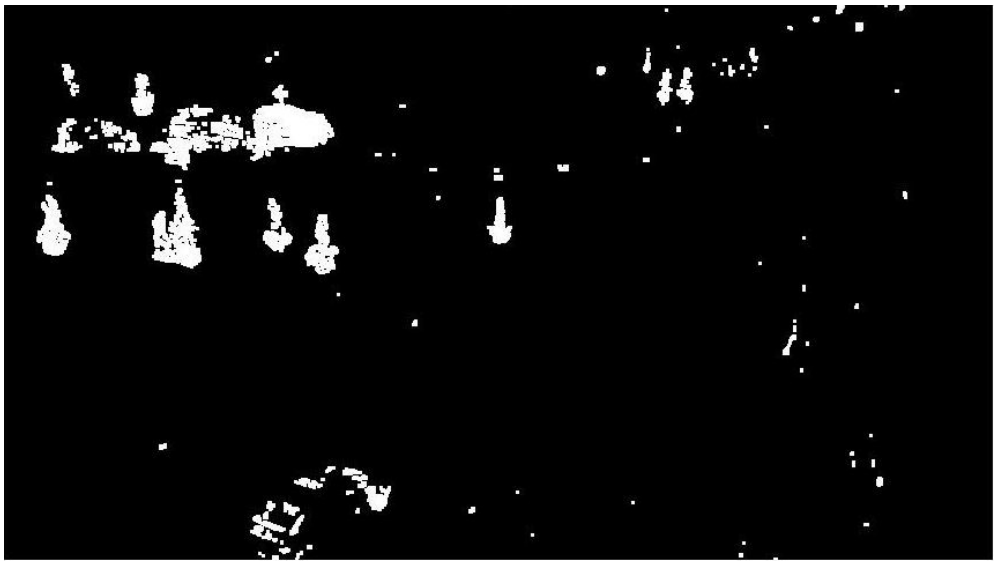

Fig. 4.7 The Detected Foreground by GMM

After the detection of moving objects, a viewable mark is required to identify the moving objects frame by frame. A bounding box or Region of Interest (ROI) is selected to identify the moving objects. There is a built-in function called 'vision.BlobAnalysis' provided in the Computer Vision System Toolbox in Matlab to produce the bounding box for the detected targets. Figure 4.8 shows the moving objects marked with bounding boxes.

Fig. 4.8 The Detected Moving Objects Marked with Bounding Boxes

4.1.3 Setting the Monitoring Region

In order to achieve the goal of event detection, a monitoring area must be created to trigger the alarm. The triggering condition can be set as "If a car and a pedestrian appear in the detecting area at the same time." When the pre-defined triggering condition has been satisfied, the alarm will be triggered. There are two advantages for the setting of the monitoring region. Firstly, the area allows the program to focus on the information within a specific area and reduce the amount of data for processing. Secondly, any other information such as flying birds or shaking trees will be removed from the processing area.

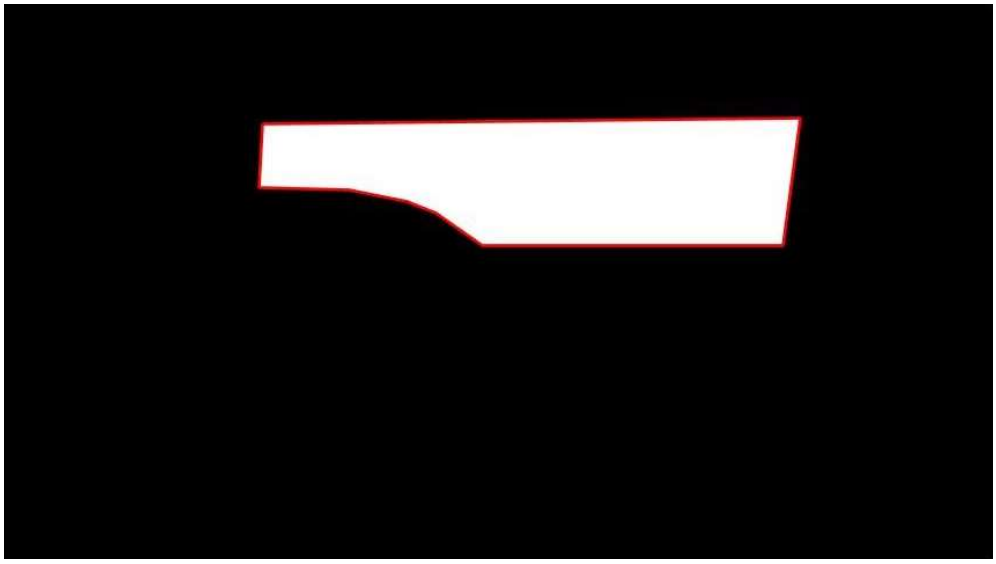

Fig. 4.9 The Mask for Detecting Area Setting

As shown in Figure 4.9, the detected region for the experiment in this project was set by the mask or sillouette. Actually, the mask is a filter to remove all the information in the pre-defined area. The mask frame in Figure 4.9 was superimposed to all of the binary frames so there is nothing outside of the detecting area. Figure 4.10 shows a frame after the setting of the detecting area.

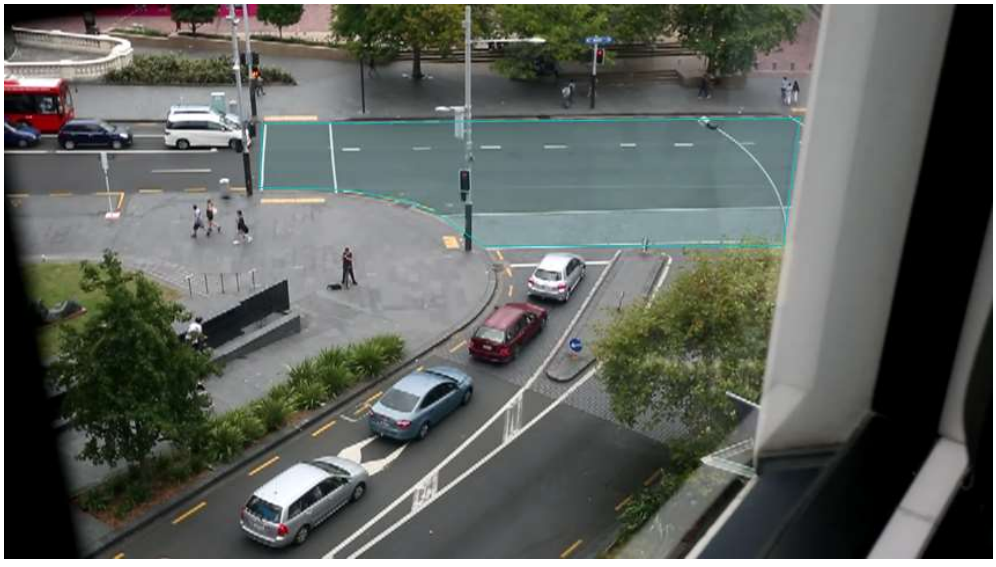

Fig. 4.10 The Detected Region or the Region of Interest (ROI)

4.1.4 Moving Objects Localization

The coordinates of a moving object represent the location of the object in the surveillance video. Depending on the coordinates, the detected moving object can be extracted from the surveillance video based on its location, then it will be able to trigger an alarm. For the details of the coordinates, on one hand, the moving objects are moving in a two-dimensional space, so the coordinates of a detected moving object are expressed as a pair of numbers: one number of these two represents the position in the direction of the X-axis, and the other one is the position in the direction of the Y-axis. On the other hand, the moving objects in a two-dimensional space occupy a certain area; each pixel contained in these areas has coordinates in the same coordinate system.

Fig. 4.11 The Centroids of Moving Objects

In order to determine the location of a moving object in the surveillance scene, the centroid of the moving objects was selected to represent the location of a moving object. The command 'CentroidOutputPort' was applied to the centroids obtaining in the Matlab in this project. Furthermore, the centroids were utilized for the object's segmentation in subsequent steps. Figure 4.11

shows the marked centroid points of the moving objects.

4.1.5 Multiple Objects Tracking

Moving objects detection based on the analysis of morphology is able to provide the spatial relationship between the environment and the object itself in a single frame (Li et al., 2013). However, a surveillance video has the information of temporal and spatial. In the view of the human visual system, people see moving objects moving across their field of view; people understand an object in different frames is exactly the same object (Yang, Pan, & Li, 2005). However, the moving objects in different frames are separable, independent and different from the point of view of the computer (Kim & Hwang, 2002). In other words, the computer vision system only extracts the current location of the moving objects in a current frame rather that a complete motion trajectory of the moving objects (Pfoser, Jensen, & Theodoridis, 2000). In addition, the detected moving objects have the chance to hide or to be overlapped by other static objects like buildings, trees and so on (Saunier & Sayed, 2006). Hence, techniques of multiple objects tracking have to be applied in order to ensure the coherence of the motion information (Li, Winfield, & Parkhurst, 2005). The multiple objects tracking method used in this project is Kalman filtering. Kalman filtering is a computer vision based algorithm which is provided in the Computer Vision System Toolbox of Matlab (Li et al., 2010). Kalman filtering is able to predict the location of a moving object in the next frame (Simon, 2010) according to the motion information extracted from the previous frames, and the predicted new location will be updated when the location of the moving object has been received (Lipton, Fujiyoshi, & Patil, 1998).

During implementation of multiple objects tracking in Matlab, five functions have been created to achieve the goal of multiple objects tracking. Firstly, the

function named ‘Initialize Tracks’ will be able to create an array for the storage of detected tracks. The initializing information for each track includes an integer ID for each track, the coordinates of the centroid and bounding box and the present interval and so on. According to the initialization of trackers, it will be able to construct a structure to manage each tracker. When a moving object has been detected, the function named ‘Assign Detection to Tracks’ is able to assign the detected moving object to the array that has been initialized by the ‘Initialize Tracks’ function. Each detected object will be assigned an ID, and the coordinates of the centroid and bounding box, the present interval and other related information will be recorded as a new track. Once a detected moving object has been assigned to the trackers, the system will be able to predict the new location of the detected moving object in the next frame using Kalman filtering (Peterfreund, 1999). Kalman filtering is able to predict the location of moving objects in the video based on their current location (Stauffer & Grimson, 2000). The function named ‘Predict New Location of Tracks’ has been created in Matlab for the purpose of location prediction using Kalman filtering and it is also known as the predicted phase of Kalman filtering. Another phase exists in Kalman filtering called update phase. Because the predicted result of Kalman filtering is slightly different with the real location of the moving object, when the moving object appears in the next frame, the real location will be able to extract it and it should be updated correctly. The function named ‘Update Assigned Tracks’ has been created for updating. Finally, if a moving object has disappeared or is moving out of the monitoring area or the field of view, the information of the object’s track must be deleted. The function named ‘Delete Lost Tracks’ has been created for this purpose. Each track has recorded information of the present interval for each detected moving object, and if the objects disappear for too many frames, the object will be considered as a lost track and the information of this track will be deleted from the array of the trackers. Figure 4.12 shows detected moving

objects applied in the tracking algorithm. Each detected moving object has been assigned to different trackers and each tracker uses a different color to draw the bounding box to distinguish the different tracks and labelled IDs.

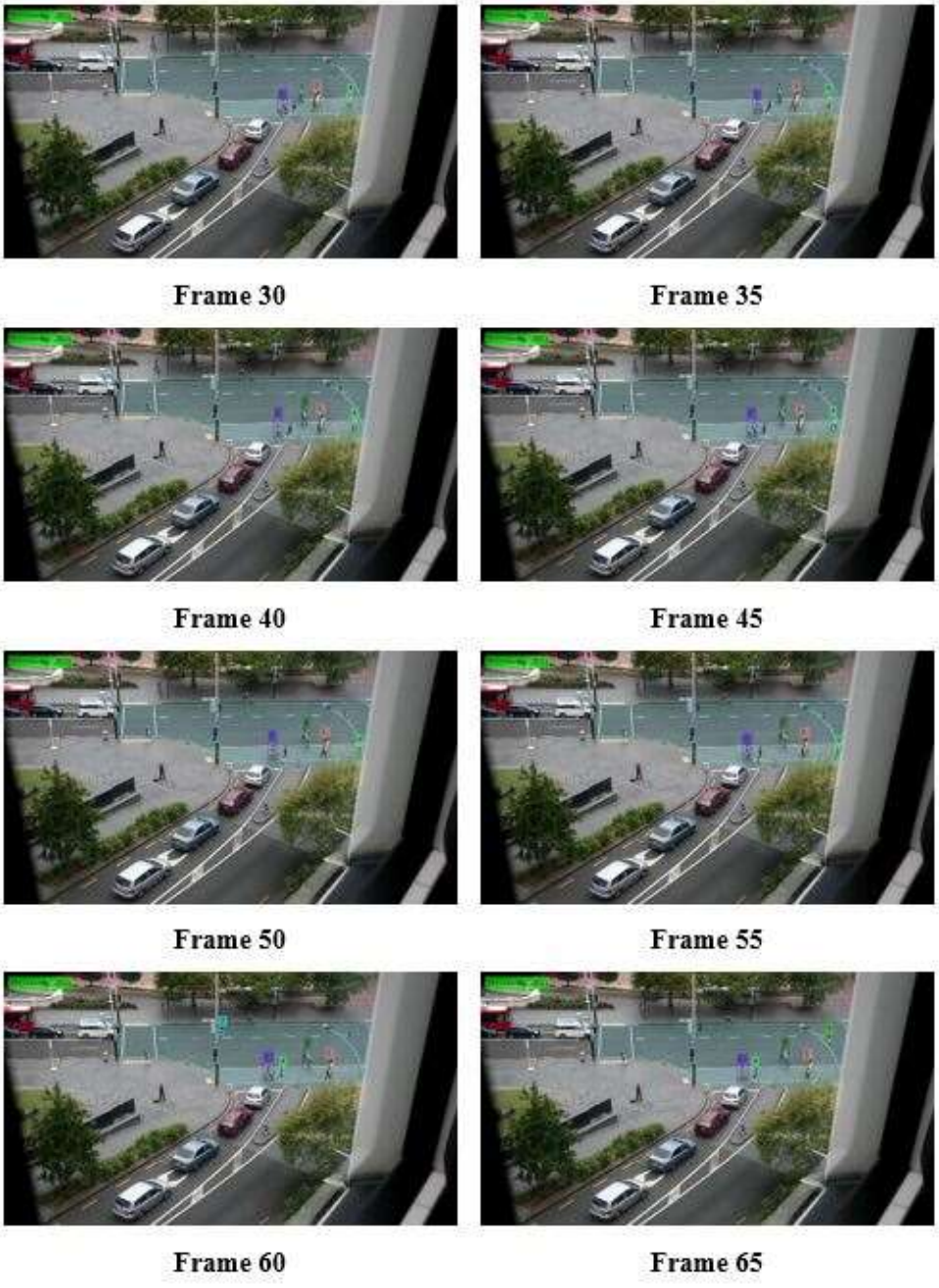

Fig. 4.12 Detected Moving Objects Applied Tracking

4.2 Abnormal Event Detection

Although the moving objects in the surveillance video have been detected and tracked based on a set of video processing methods and techniques, the result of the detection and tracking contains various types of objects such as pedestrians, vehicles and so on. Artificial Neural Network (ANN) is a powerful and popular classification tool in the field of computer technology, and are also widely utilized for data analysis, clustering, pattern recognition and so on. ANNs provide a systematic and clear overview of the features, structure and the working process (Zelnik-Manor & Irani, 2006)). In this section, the configuration, training, testing and implementation of ANNs used in the project of this thesis will be described.

4.2.1 Dataset Description

According to the description of the ANN, it requires a training process. Hence, a suitable dataset is required to complete the training of the ANN. There are a plenty of human video datasets which have been utilized for human behavior recognition. Moreover, the human body can be seen as specific video frames with two-dimensional motion. In this project, the datasets from INRIA will be utilized for pedestrian detection. In the positive sample, there are 614 images and 2416 pedestrians. The pedestrians in this dataset are all standing and view angles are various including front, back and from various sides. The negative samples of the INRIA person dataset include 1218 images, and these images are non-pedestrian. The negative samples will be input into the neural network for training the neural network.

4.2.2 Feature Extraction

The INRIA person dataset provides a vast quantity of positive and negative

samples for the training process of the ANN. But the samples collected in the INRIA person dataset are saved as image format. The problem is that the image data cannot be input into the ANNs directly for training and testing. Therefore, the abstract information must be extracted from the images. The abstract information extracted from the images contains representative data of the images known as a "feature". In this project, the HOG feature and LBP feature have been extracted from the images collected in the INRIA person dataset to compare and test the recognition performance in the following sections.

HOG (Histogram of Oriented Gradients) is a kind of classification feature that usually utilized for the recognition of human body. The working process of HOG is to detect the appearance and shape of a local object using light intensity gradient or edge distribution. Before the extraction of HOG feature, we have to segment the image into a number of small connection regions which called 'cells'. Each of cell generates a histogram of oriented gradients, or cell edge direction of one pixel, the collection of these histograms will be able to express the HOG feature of an image. The length of the HOG feature is affected by the segmentation of 'cell'. With the increasing number of cells, there are more feature will be extracted from the image. But it is not necessary to extract the HOG feature for each cell because of the processing speed. Therefore, we further segmented the image to a number of 'blocks' and extracted the feature for one cell in each block as the sample of this block. According to the segmentation of blocks, the length of the extracted HOG feature will be shorter that the feature extracted for each cell to reducing the data amount of processing.

LBP (Local Binary Pattern) is an operator to describe the local texture feature of an image. The advantage of this operator is the characteristic of rotational invariance and gray scale invariance. Therefore, LBP is a simple and effective

algorithm for feature extraction. LBP describes a relationship between a pixel and the others around it. The extraction of LBP depends on the grayscale of the image. For one target pixel, there are at more than 8 pixels around it, if the gray scale level of surrounding pixels is greater than the grayscale level of the pixel in the middle, they will be marked by 1. On the contrary, they will be marked by 0. The eight-bit binary number will be the LBP feature of the pixel in the central. For the LBP extraction in practice, the original image will be divided into a number of cells with a fixed size usually using the size of 16 × 16. After that, we applied this method to obtain the eight-bit binary number and then calculate the histogram for each cell and applied normalization to the histograms. Finally, we combine all the histogram of cells to get a feature vector of the whole image. According to study, because of the cell size for sampling used in LBP is much bigger that the size of cells used in HOG, the length of extracted LBP feature will be shorter than the length of HOG feature, the processing speed will be much faster that HOG feature.

4.2.3 Artificial Neural Network Design

Artificial Neural Networks (ANNs) have been applied to classification for pedestrian recognition in this project. The details of the ANN have been described in the previous section. Hence, the implementation of ANNs that will be described in this section include the organization of data, training process, parameter setting and so on.

The training data is a core element to establish a functional ANN. Through feature extraction, a set of the feature vectors is obtained for both positive and negative samples, but the data still cannot be input into the ANN classifier for training directly because the classifier cannot distinguish whether a feature vector was extracted from a positive sample or a negative sample. Therefore, before the start of the training process, the data needs to be organized through

the process of labelling.

Labelling (Tagging) is a process of adding the symbols ‘0’ and ‘1’ to the training data. Through the tagging process, the computer will be able to understand whether a feature vector is extracted from a positive sample or a negative sample.

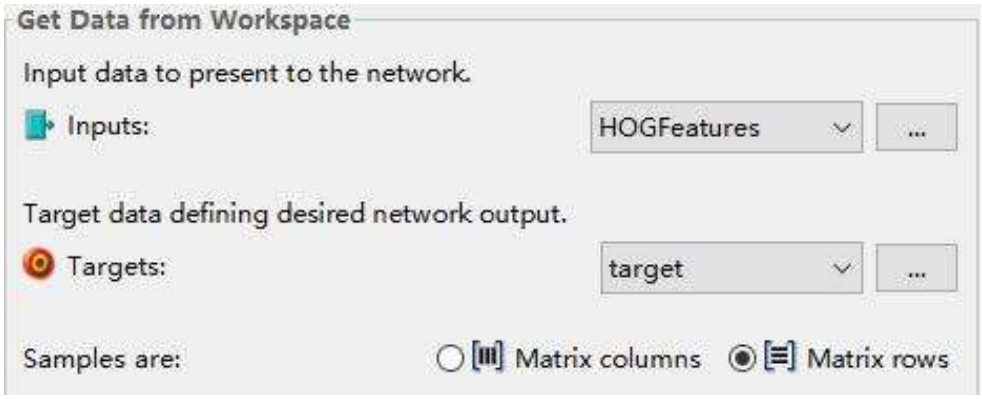

Fig. 4.13 The Data Required for ANN Training

As shown in Figure 4.13, there are two datasets required for the training of the ANN in Matlab. The ‘Input’ dataset is required to input the set of feature vectors extracted from the images described in the last section. The ‘Targets” require another dataset to identify the type of feature vector as the ‘tag’. These two datasets are prepared during the feature extraction and stored in the workspace in the Matlab. The ‘HOGFeatures’, contains the extracted HOG feature vectors, ‘0’ representing the negative samples and ‘1’ standing for the positive samples.

Once the organization of the training datasets are ready, they will be divided into three classes randomly: training set, testing set and validation set, as the default setting shows in Figure 4.14. In Matlab, 70% of the total number of samples will be grouped into the ‘training set’ and utilized for training the

ANN. 15% of the total number of samples will be grouped into the ‘testing set’ and utilized to measure the performance of the ANN during and after training. The testing set will not be able to affect the training and performance of the ANN as it is utilized for testing only. The remaining 15% will be grouped in the ‘validation set’ to verify the quality of ANN training.

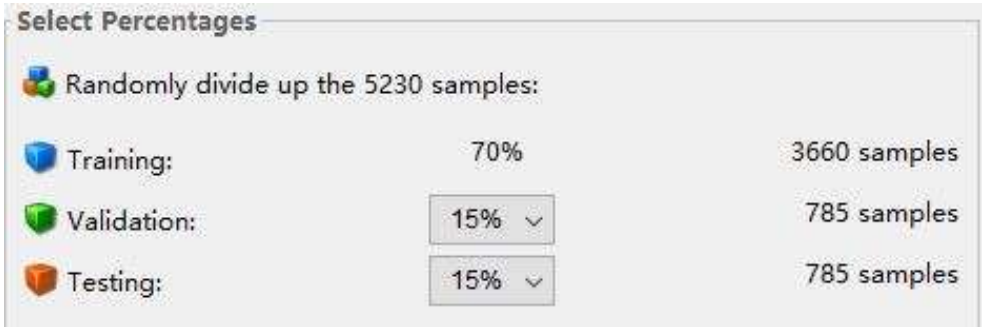

Fig. 4.14 The Default Setting of Data Grouping in Matlab

For the structure of the ANN classifers, a feedforward ANN will be trained through supervised learning. The ANN is composed of an input layer, an output layer and a hidden layer. Many neurons exist in the hidden layer. The default number of neurons in the hidden layer is 10. The number of neurons will be retained as the default setting without change and the neurons in the hidden layer will be self-organized. Figure 4.15 shows the structure of the trained ANN.

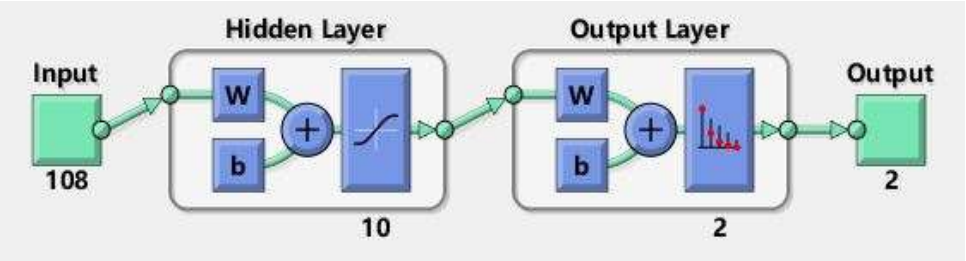

Fig. 4.15 The Structure of the ANN in the Project

4.2.4 Target Extraction

After the training process, the trained ANN will be saved as a '.mat' file for recognition. Because the system should be able to detect multiple moving objects in the video, the ANN can only deal with one object recognition at a time, so the targets in a frame should segmented and inputted to the ANN one-by-one. The length of the features extracted from the target images should be exactly as same as the training features, otherwise the ANN classifier will not work due to the nonmatching length of feature vectors. Figures 4.16 and 4.17 show the target images from the video frames with the size of 70 × 134.

Fig. 4.16 Target Images with Pedestrians

Fig. 4.17 Target Images without Pedestrians

For each of the target images, the feature vectors such as HOG or LBP are also extracted using the same method of feature extraction from the training images. The features extracted from the segmented images will be sent to the trained neural network for classification. The result will be sent back to the ANN classifier and continually update the 'type' for each track as shown in Table 4.1.

Table 4.1 The Recognition of Segmented Images

Type	**Display**	**Description**
Pedestrian	Pedestrian	Recognized as pedestrian
Car	Car	Recognized as non-pedestrian
Others	Identifying…	Unrecognized or recognized

4.3 Alarm Making

The final step of this thesis project is alarm making for abnormal events. Based on the results obtained from the video processing and the ANN, the system will be able to detect abnormal events to trigger an alarm. In this project, there are two scenarios which have been designed to test the alarm making function of the system. The details of the alarm making including the scenario, triggering condition and the alarming approaches will be described in this section.

The testing scenario of an alarm making system is an intersection of urban streets, with pedestrians and cars passing through the intersection from four directions. Under normal circumstances, cars and pedestrians have their own time window to pass through the intersection which means that the pedestrians and cars are not allowed to appear in the center of the intersection at the same time. Abnormal events will occur such as traffic violations. Under abnormal circumstances, cars and pedestrians will appear in the center of the intersection at the same time. Figure 4.18 shows a flowchart for decision making of alarming. The decision making is based on a Decision Tree. According to the flow chart, the moving objects in the video will be firstly detected and then the type of each object will be distinguished. If a car and a pedestrian appear at the same time in the detecting area, the computer will generate an alarm to notify that something happened in an abnormal way.

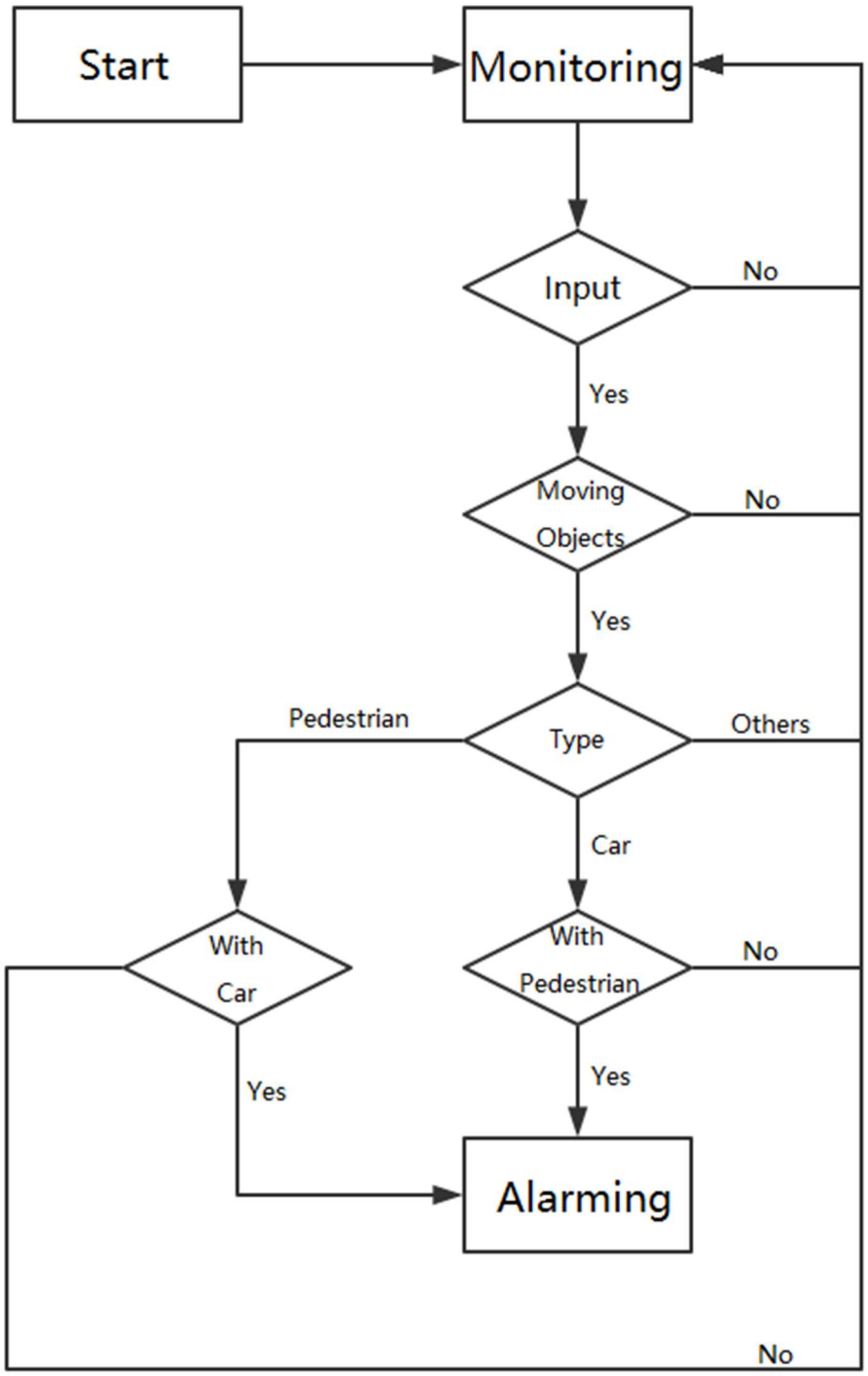

Fig. 4.18 Decision Making for Alarming

Our surveillance camera is placed in a car and captures the video view in front of the car. Under normal circumstances, pedestrians are not allowed to walk in

front of the vehicles or too close to the vehicles otherwise it will be considered dangerous or harmful to personal safety. Therefore, an abnormal event will be detected by the computer if pedestrians are walking in front of the vehicles or too close to the vehicles. Taking this abnormal event into consideration, the computer should report an alarm to the driver. The triggering of an alarm for an onboard view scenario is slightly different from the street view scenario. There is one more step to calculate the probability of collision in light of the current speed of the car and the distance between the pedestrian and the camera. In an actual case, the current speed would be provided by GPS, but in this project, the current speed will be simulated by manually inputting. For the distance between the moving object and the camera, because the video was captured by a monocular camera, it is not possible to provide depth information as it would be if a dual camera was applied. Therefore, the pixel distance between the centroid of the moving object and a fixed point at the bottom of the video will be adopted as the distance between the moving object and the camera. The calculation of collision probability is defined as:

$$CP = ((V_t / D_f) / 1.5) \times 100\% \quad (4.2)$$

where CP is the collision probability, V_t is the current speed of the vehicle and D_f is the distance between the centroid of the moving object and the camera in the current frame. According to the collision probability calculation, the system will be able to produce an alarm to indicate the severity of the event.

There are two approaches utilized in this project for alarm making: one approach is the screen display, the is through a loud speaker. When an abnormal event has been detected, the computer will be able to pop up an alert message on the screen to notify the driver that an abnormal event has happened. The color of the alert message is yellow and flashing to make the message

more visible.

Sometimes, drivers are not focusing on the screen before the message disappears. In order to avoid missing the warning message as much as possible, the use of a 'beep' sound is the best way to alert the driver. Different frequency sounds will be produced to indicate the severity of the detected events. For example, according to the collision probability, the events can be grouped into levels such as harmless, attention, warning, harmful, particularly dangerous and so on. For different levels of severity of the detected events, the system will be able to produce distinct beep sounds as shown in Table 4.2.

Table 4.2 The Different Level of Warning

Severity Level	**Warning Frequency**	**Description**
Level 5	0 times per second	Harmless
Level 4	2 times per second	Attention
Level 3	5 times per second	Warning
Level 2	10 times per second	Harmful
Level 1	20 times per second	Particularly Dangerous

Chapter 5

Experimental Results and Discussions

The testing data and results of the system developed in the experiment have in-depth evaluation and reflection evident in this chapter. Specifically, the features HOG and LBP for abnormal event detection are adopted. The solutions applied to achieve the research objectives are also discussed including limitations, implications and recommendations.

5.1 Experimental Environment

The approaches related to the implementation of the project have been introduced in previous chapters. The results of the experiment will be described and analyzed in this section. The experiments were conducted on a laptop equipped with an Intel Core i7-3720QM CPU and quad-core and eight threads which allows multitasking in 2.60 GHz at default setting and turbo boost to 3.60 GHz. The experimental platform is Matlab 2015b.

5.2 Video Processing Module

As the infrastructure of this experiment, the video processing module directly affects the performance of the whole system. The video processing module is mainly responsible for video pre-processing, moving objects detection, multi objects tracking and so on. The results of the video processing module will be described.

5.2.1 Moving Objects Detection

In the project, the GMM was applied to the system for the purpose of moving object detection. In order to test the performance of moving object detection, a part of the video containing 550 frames was selected as shown in Figure 5.1. In this part of the video, there are six cars, and each of them were driven through the monitoring area. The first step was to count the total number of frames for each car to turn up in the monitoring area and label them as the ground truth. Then, the program was run to detect the cars and count the total number of frames for the correctly detected results of each car. The correctly detected results mean all the effective detection, while any lost detection or redundant detection will not be taken into account.

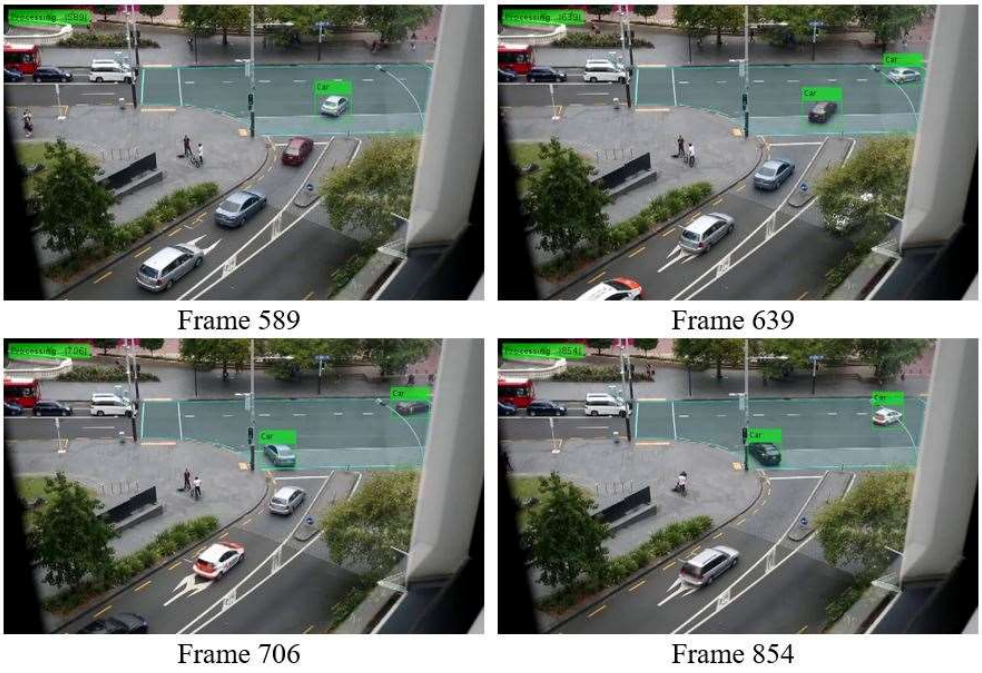

Fig. 5.1 The Results of Moving Object Detection

Table 5.1 The Precision of Moving Objects Detection

	Car 1	**Car 2**	**Car 3**	**Car 4**	**Car 5**	**Car 6**
Color	Silver	Red	Blue	Silver	White	Black
Display Frame	530-660	595-726	670-794	725-832	776-882	820-940
Ground Truth	130	131	124	107	106	120
Detected Frames	128	112	87	95	98	87
Precision	98.5%	85.5%	70.2%	88.8%	92.5%	72.5%

The results are recorded in Table 5.1. According to the results shown in the table, the best precision for moving objects detection in this experiment was

98.5%, and the worst was 70.2%. The overall result of moving objects detection in this experiment was 84.7%. Figure 5.1 shows some of the frames of the video as examples of detected moving objects.

5.2.2 Multiple Objects Tracking

For the tracking of moving objects, Kalman filtering was applied to predict the new location of the moving objects and assigned the detected moving objects to the tracker (Tao, Sawhney, & Kumar, 2002). The tracked moving objects were allocated a tracked ID to identify the tracks of the moving objects. In order to test the performance of the multiple objects tracking, the same video file as with the moving objects detection testing was selected as the testing sample.

There are 550 frames in the video clips, and there were six cars moving through the detecting area. As for the moving objects detection, the number of frames for each car appearing in the detecting area was recorded as the ground truth. After the tracking method was applied, each car was tracked and identified with different colors of the bounding box. The number of frames for correct tracking was also recorded. Any incorrect tracking such as lost tracking or ID changing was taken into account. Figure 5.2 shows the frames of the video as a sample of the tracked moving objects.

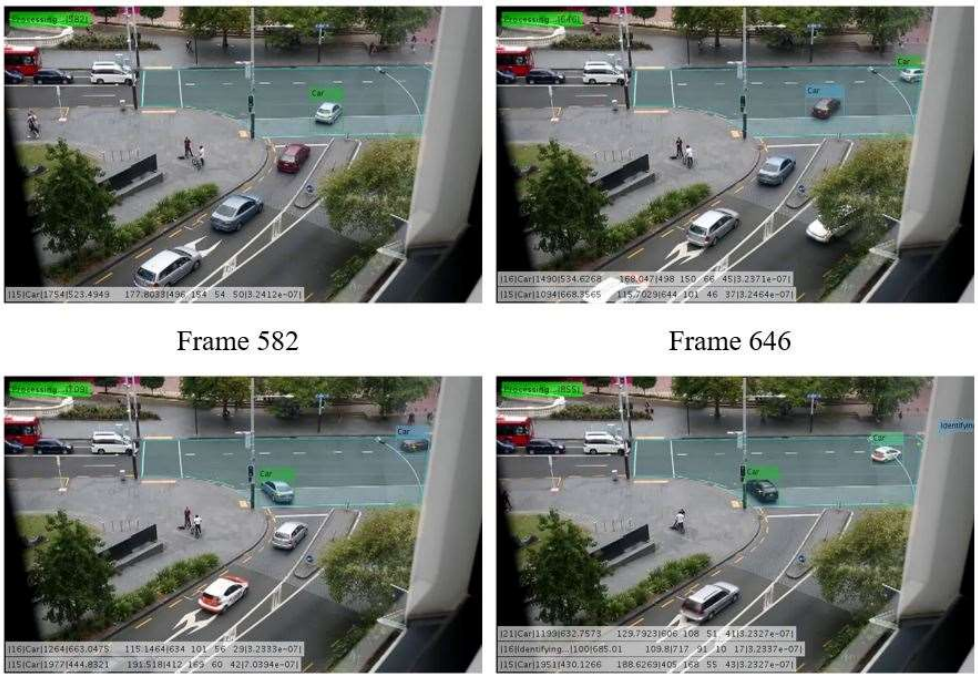

Fig. 5.2 The Results of Multiple Object Tracking

Table 5.2 The Precision of Multiple Objects Tracking

	Car 1	**Car 2**	**Car 3**	**Car 4**	**Car 5**	**Car 6**
Color	Silver	Red	Blue	Silver	White	Black
Display Frame	530-660	595-726	670-794	725-832	776-882	820-940
Ground Truth	130	131	124	107	106	120
Track ID	15	16	15	19	21	15
Tracked Frames	128	114	89	98	97	87
Precision	98.5%	87.0%	71.8%	91.6%	91.5%	72.5%

According to the experimental results for multiple objects tracking shown in Table 5.2, the best precision for multiple objects tracking in this experiment was 98.5% and the worst was 71.8%. The overall result of multiple objects tracking was 85.5%.

5.3 Pedestrian Recognition Module

5.3.1 Features and Classifiers

Two features, HOG and LBP, were utilized for pedestrian recognition. The ANNs were applied as the classification tool in this project. This section will describe the results of pedestrian recognition, and the performance of various classifiers including KNN (K-Nearest Neighbor), DT (Decision Tree) and MLP (Multilayer Perceptron) will be compared. The testing sample used the INRIA person dataset. Figure 5.3 shows the dataset for the experiment. (http://pascal.inrialpes.fr/data/human/).

Fig. 5.3 INRIA Dataset

The INRIA dataset contains the positive and negative samples. In the positive samples, there are 614 images and 2416 pedestrians. The pedestrians in the images of this dataset are all standing; view angles are various including front, back, and from various sides. The feature extraction will be applied to the testing sample to extract the features for both HOG and LBP. Figures 5.4 and 5.5 show the classification results for HOG and LBP. The recognition result is shown in the top-left corner of the images; the label 'P' represents that the object is recognized as a pedestrian and the label 'B' refers to non-pedestrian objects. The accuracy of HOG and LBP in different classifiers will be shown later.

Fig. 5.4 The Results of Pedestrian Recognition by HOG

Fig. 5.5 The Results of Pedestrian Recognition by LBP

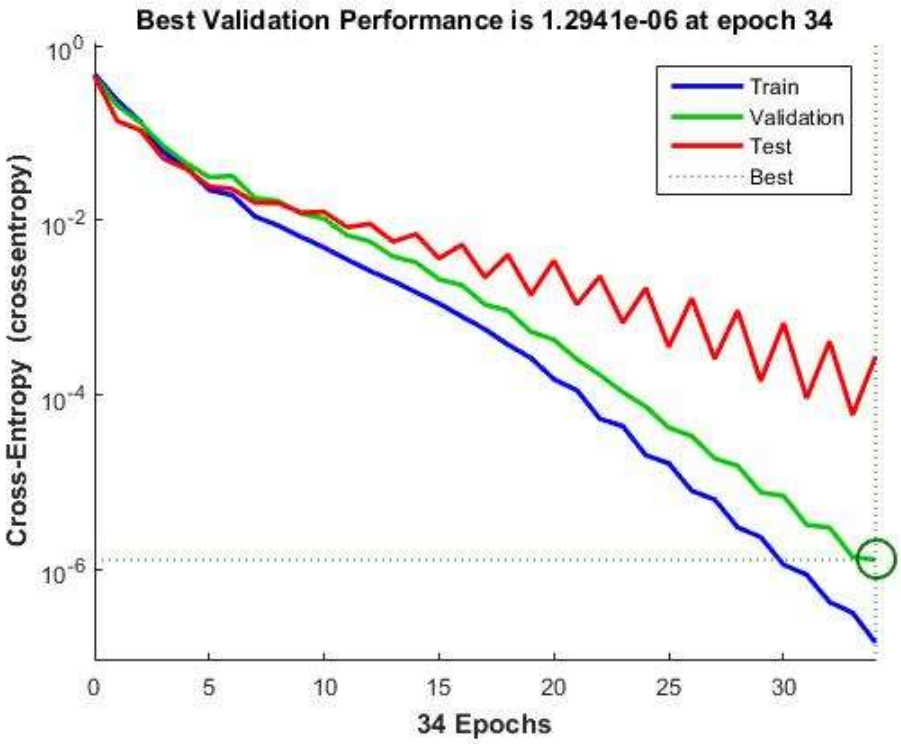

Fig. 5.6 The Performance with Less Training Features Using HOG

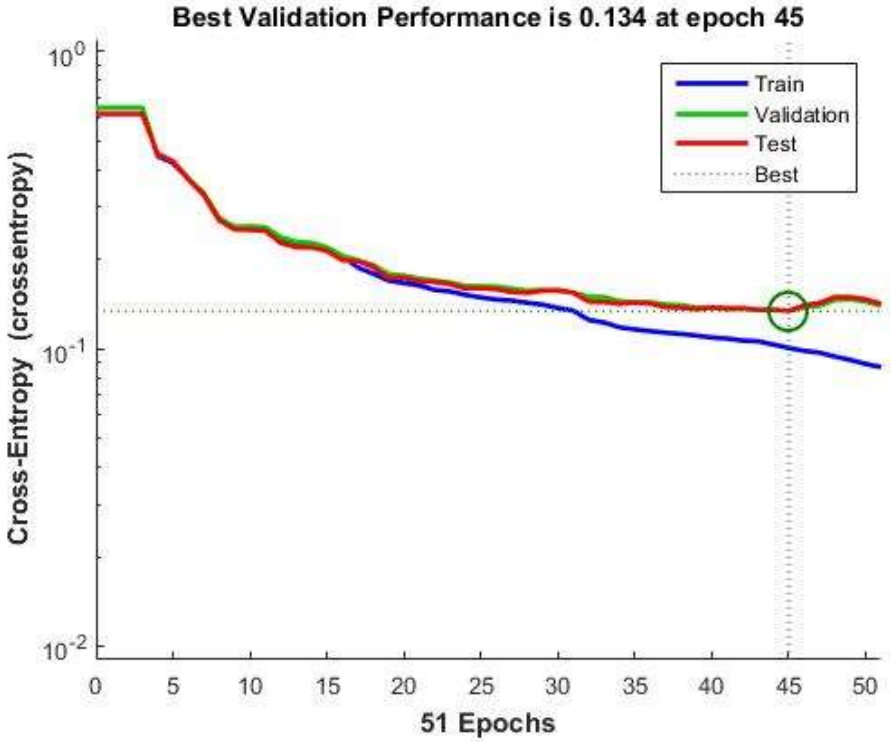

Fig. 5.7 The Performance After Increasing Training Features Using HOG

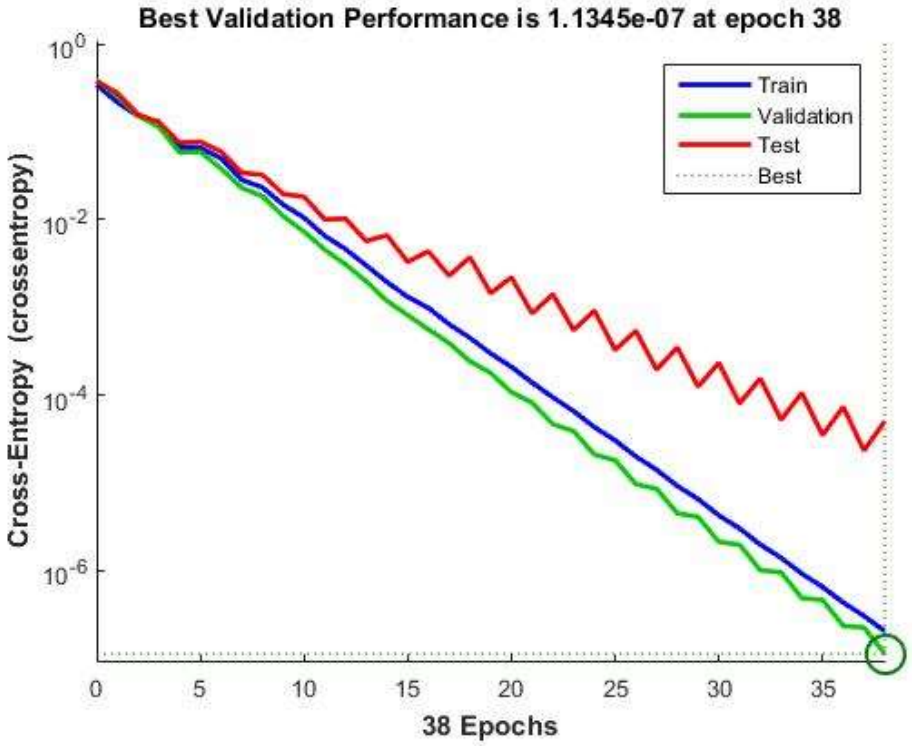

Fig. 5.8 The Performance with Less Training Features Using LBP.

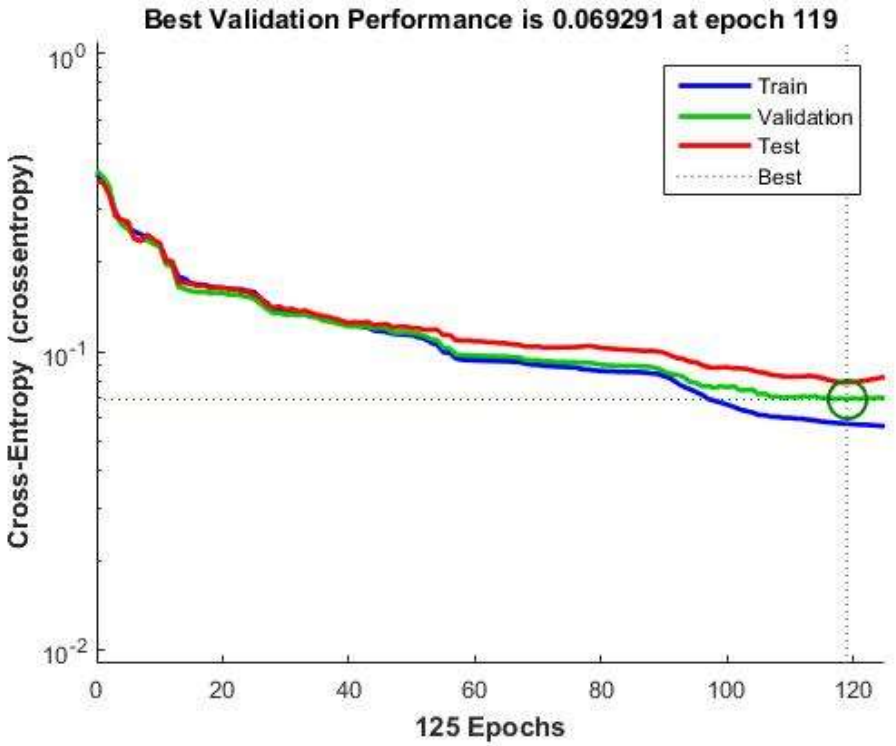

Fig. 5.9 The Performance After Increasing Training Features Using LBP

Table 5.3 The Accuracy for Various Features and Classifiers

	HOG		LBP	
Classifier	**Pedestrian**	**Non-pedestrian**	**Pedestrian**	**Non-pedestrian**
KNN	94.2%	94.7%	94.9%	94.1%
DT	87.9%	87.7%	86.4%	86.8%
MLP	95.6%	94.4%	93.9%	93.0%
ANN	97.3%	97.5%	90.7%	93.1%

Table 5.3 shows the comparisons of accuracy between various features and classifiers. According to the comparison between the results obtained from different features and classifiers, the HOG feature provided a better accuracy for pedestrian recognition when the ANN utilized as the classification tool. The overall accuracy of the ANN trained by the HOG feature for pedestrian recognition is up to 97.3%. Also, the ROC curve shows in Figures 5.6 to 5.9,

for both HOG features and LBP features, with the increasing number of samples input to the ANN for training, the performance of the ANN for the classification task increased. Figure 5.10 shows the correct recognition result using the ANN and HOG feature for the external images rather that the sample image collected in the INRIA person dataset.

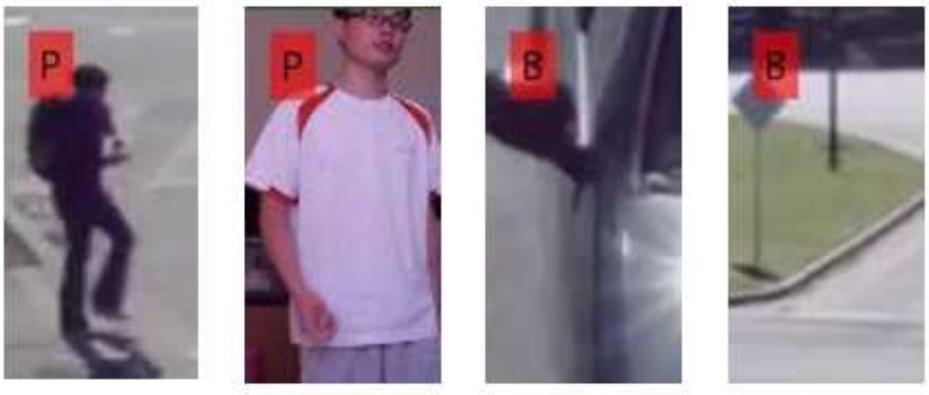

Fig. 5.10 The Recognition Results for External Samples

5.3.2 Region of Interest in Object Detection

The moving object had to be segmented from the video frames and the HOG feature extracted to be sent to the ANN for the recognition task. In order to test the performance of moving objects segmentation, a part of the video with 350 frames was selected as the sample video. Then, the moving objects segmentation program was run to segment the moving objects. The size of the segmented images was normalized to 70 × 134. There are 2000 images were randomly selected from the segmented images. The number of correct segmented images and incorrect segmented images were recorded as the experimental data. There are two criteria to measure the correct segmentation: the object should appear around the centre of the segmented image, and the segmented image should contain more than 90% of the object image. Any images that do not fit these two criteria were considered a failure. According

to Table 5.4, the precision of the correct objects segmentation is 82.2%.

Table 5.4 The Result of Objects Segmentation

	Correct Segmentation	**Incorrect Segmentation**
Result	1643	357
Total Sample	2000	
Precision	82.2%	17.8%

Fig. 5.11 Correct Segmented Images

5.4 Alarm Making Module

The final step is to make alarms when the triggering condition is reached. The results of the alarm making depend on processing results from previous results. In order to test the performance of the alarm making module, a sample video containing 331 frames was selected. The system will be able to recognize the pedestrian in the video based on the previous steps and calculate the collision probability based on pre-defined speed for alarm making. The system can generate different levels of alarm as the explanation in the previous section. For different levels of alarm, the frequency of beep sound will be different. With the increase of alarming levels, the frequency of beep sound will be more rapid. Figure 5.12 shows the testing sample of alarm making.

Fig. 5.12 The Results of Alarm Making

As shown in Table 5.5, the results for different speeds have been recorded. The alarming frames are the number of frames that the alarm should be generated.

The false alarm frames recorded the frames with unnecessary alarms or alarming using the beep sound of other levels. The overall false alarm rate of the system developed for this project was 13.8%.

Table 5.5 The Results of Alarm Making

Setting Speed (km/h)	0	20	40	60	80	100	120
Total Frames	331	331	331	331	331	331	331
Alarming Frames	0	106	108	105	109	109	112
False Alarm (Frames)	0	13	10	22	18	29	12
False Alarm Rate	0%	12.3%	9.3%	20.9%	16.5%	26.6%	10.7%

5.5 Discussions

The results of this experiment have been described and demonstrated in this chapter. Each of the three sections is related to the different modules developed in this experiment. The results related to the video processing module are discussed in Section 5.2. Section 5.2 has two sub-sections: Section 5.2.1 described the moving objects detection and Section 5.2.2 described multiple objects tracking. For the moving objects detection, GMM has been applied to background modelling. Based on the GMM, the system will be able to extract the moving objects from the video frames. Compared with frame differencing, GMM provided a better performance for moving objects, the overall result of GMM based moving objects detection being up to 84.7%. However, the pre-condition of the application of GMM is that the camera must be stable without

motion. The results of multiple objects tracking by using Kalman filtering were shown in Section 5.2.2. The overall result of multiple objects tracking was 85.5%. Section 5.3 is about the application of the ANNs for pedestrian recognition. Section 5.3.1 compared the results obtained by different feature selection and classifiers. The experiment proved that the ANN is a more advanced classifier based on machine learning. The performance of the ANN is also higher than other classifiers such as KNN, MLP and DT. In addition, the HOG feature has been proven more suitable for the pedestrian recognition task than the LBP feature although LBP is much faster than the HOG in processing speed. The overall accuracy of pedestrian recognition using the ANN trained by the HOG feature is up to 97.3%. The results of moving objects segmentation are recorded in Section 5.3.2. The moving objects were segmented from the video frames for feature extraction and recognition. There are 2000 segmented images were selected as samples, and the correct objects detection rate was 82.2%. Finally, the alarm making module was described in Section 5.4. The probability was calculated in this module, and the false alarm rate of the system developed for this project was 13.8%.

Chapter 6
Conclusion and Future Work

In this thesis, a computer vision based alarm making approach has been developed, tested and discussed. A clear and substantial articulation of the significance including limitations and implications of the experiment will be described in this chapter. It will also provide recommendations for the improvement of the current system in the future.

6.1 Conclusion

Following the requirement of improving the safety of the traffic environment and reducing the human labor, a computer vision based alarm making approach has been developed for traffic environments. In the first chapter of this thesis, the important and significant role of video-based surveillance system in our daily lives was introduced. Also, the computer vision techniques related to the development of the project in this thesis were briefly described. The ultimate goal of this thesis is to investigate the techniques of computer vision and develop a computer vision based alarm making approach for traffic environments. There are basically four modules employed in the system developed in this project: moving objects detection module, multiple objects tracking module, pedestrian recognition module and alarm making module.

There are five parts to the design and implementation in this project. Firstly, in the video pre-processing part, the means to provide better video frames for subsequent processes was introduced. In order to remove the interference from shaking trees and flying birds etc., a mask was applied to the frames in the video pre-processing part. In the moving objects detection part, GMM has been applied to background modelling. Based on the background model created by GMM, the foreground such as vehicles, pedestrians and other moving objects will be detected in the video frames. Once the moving object had been detected, the position of the detected moving objects was assigned to trackers. Kalman filtering provided the function of moving object tracking. The Kalman filtering consists of two phases: predict phase and updating phase. Based on these two phases, the system can predict the new position of the detected moving objects and continually update the real position of the objects. The centroid of the moving object was utilized for the localization of moving objects, and then the ROI was applied to segment objects from video frames.

For the segmented images, the HOG feature and LBP feature was extracted and input to a trained ANN for recognition. Combined with the alarm triggering rules, the abnormal event was detected and the alarms triggered to notify the users. The alarming module adopts a different frequency of beep sound and eye-catching display for the alarm approach.

The experimental results and discussions were present in the Chapter 5. In this chapter, the testing results for each module were recorded and discussed. Furthermore, according to the analyzing and comparison between different methods and algorithms, the ANN and region-based method were combined and applied in the system for pedestrian recognition. The results show this combination gave a better performance.

In conclusion, there is a new schema presented in this thesis for the implementation of a computer vision based alarm making system. The system is able to detect moving objects and track the trajectory of the moving objects. For the detected moving objects, the trained artificial neural network is utilized to detect the abnormal events and make alarm based on the recognition result.

The contributions of this thesis are: firstly, the shortcomings of the traditional surveillance and alarm systems have been studied. Secondly, the computer vision based techniques have been utilized to alarm making. Thirdly, it has innovatively deployed artificial neural networks for abnormal events detection and improved the accuracy of alarming to reduce false alarms. The overall result for the false alarm rate of the system developed in this project was 13.8% which is lower that the mainstream 15.27% which without ANN implemented and also helpful for the management of traffic environments.

6.2 Limitations and Future Work

After a long period of developing and testing, the whole computer vision based alarm making has been implemented. However, there is still space for advancement. Due to the constraints of techniques and time, there are still limitations existing in the developed system which should be improved in the future:

(1) The sample videos for the experiments are captured by a normal camera or a smartphone built-in camera. Therefore, the resolution and contrast are unable to compete with professional surveillance cameras like CCD and so on.

(2) A mask was applied for interference removing, but the shortcoming is that the mask covers only limited regions under monitoring. If abnormal events happened in the area out of the mask area, they would not be detected.

(3) The alarming approach is based on GMM. Therefore, the system can only cope with the video captured by a stationary camera.

(4) The training dataset of ANNs is limited. The amount of training datasets will be affected by the performance of the ANN.

In future, our suggestions for the improvement of current system are:

(1) The GMM should be replaced by other foreground detection algorithms with a moving camera capturing surveillance videos.

(2) The training dataset utilized for the training of the ANN could be expanded. In order to improve the performance of the ANN, there are a large number of training samples that could be provided for the training process of ANNs.

References

Artikis, A., Baber, C., Bizarro, P., & Canudas-De-Wit, C. (2014). *Scalable proactive event-driven decision making. IEEE Technology & Society Magazine*, 33(3), 35-41.

Alm, H., & Osvalder, A. L. (2012). *The alarm system and a possible way forward*. Work, 41(Supplement 1), 2840-2844.

Bandini, S., Bogni, D., & Manzoni, S. (2002). *Alarm correlation in traffic monitoring and control systems: A knowledge-based approach*. InECAI (Vol. 2002, pp. 638-642).

Beauchemin, S. S., & Barron, J. L. (1995). *The computation of optical flow*. ACM computing surveys (CSUR), 27(3), 433-466.

Baumgartner, B., Rodel, K., & Knoll, A. (2012). *A data mining approach to reduce the false alarm rate of patient monitors*. Conference: International Conference of the IEEE Engineering in Medicine & Biology Society IEEE Engineering in Medicine & Biology Society Conference (Vol.2012, pp.5935). Conf Proc IEEE Eng Med Biol Soc.

Cong, Y., Yuan, J., & Liu, J. (2011). *Sparse reconstruction cost for abnormal event detection*. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on (pp. 3449-3456). IEEE.

Chien, S. Y., Ma, S. Y., & Chen, L. G. (2002). *Efficient moving object segmentation algorithm using background registration technique*. IEEE Transactions on Circuits and Systems for Video

Technology, 12(7), 577-586.

Ding, D., Cooper, A., Pasquina, F. (2011). *Sensor Technology for Smart Homes*. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/21531517/

Dreiseitl, S., & Ohno-Machado, L. (2002). *Logistic regression and artificial neural network classification models: a methodology review*. Journal of biomedical informatics, 35(5), 352-359.

Gomez, J., & Dasgupta, D. (2002). *Evolving fuzzy classifiers for intrusion detection*. In Proceedings of the 2002 IEEE Workshop on Information Assurance (Vol. 6, No. 3, pp. 321-323). New York: IEEE Computer Press.

Huang, L., Chen, T., Wang, Y., & Yuan, H. (2015). *Congestion detection of pedestrians using the velocity entropy: A case study of Love Parade 2010 disaster*. Physica A: Statistical Mechanics and its Applications, 440, 200-209.

Huanxiong, X., Shuwen, S., Yiwu, Y., Wenzeng, G., Shanshan, Z., & Jie, W. (2011). *Design and Realization of Fire Alarm by Determining Probability Based on Multi-Sensor Integrated*. Computer Measurement & Control, 2, 042.

Huicong, T. (2002). *Methods to reduce the false alarm*. Retrieved from http://wenku.baidu.com/linkurl=ocsYZA9X5BHj3vPGd63xwBpwRgOzvccIXakKGGZRl5P1Od_9jz54YhbI7UKaeFdOsiilIi_ZqxE1CVzJUxjn9i9jurTGCdbmbZTwRALS

Hu, J., & Yi, Y. (2016). *A two-level intelligent alarm management framework for process safety*. Safety science, 82, 432-444.

Hampapur, A., Brown, L., Connell, J., Ekin, A., Haas, N., Lu, M., ... & Pankanti, S. (2005). *Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking*. IEEE Signal Processing Magazine,22(2), 38-51.

Horn, B. K., & Schunck, B. G. (1981). *Determining optical flow*. Artificial intelligence, 17(1-3), 185-203.

He, X., & Xu, S. (2010). *Process neural networks: Theory and applications*. Springer Science & Business Media.

Hayashi, Y., Setiono, R., & Azcarraga, A. (2016). *Neural network training and rule extraction with augmented discretized input*. Neurocomputing.

Hinton, G. E., Osindero, S. & Teh, Y. (2006). *A fast learning algorithm for deep belief nets*. Neural Computation, 18, pp 1527-1554.

Jiang, F., Wu, Y., & Katsaggelos, A. K. (2007). *Abnormal event detection from surveillance video by dynamic hierarchical clustering*. In 2007 IEEE International Conference on Image Processing (Vol. 5, pp. V-145). IEEE.

Jain, A. K., Mao, J., & Mohiuddin, K. M. (1996). *Artificial neural networks: A tutorial*. IEEE computer, 29(3), 31-44.

Jin, W., Li, Z. J., Wei, L. S., & Zhen, H. (2000). *The improvements of BP neural*

network learning algorithm. In Signal Processing Proceedings, 2000. WCCC-ICSP 2000. 5th International Conference on (Vol. 3, pp. 1647-1649). IEEE.

Kohavi, R. (1996). *Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid*. In KDD (Vol. 96, pp. 202-207).

Kruegle, H. (2011). *CCTV Surveillance.: Elsevier Science*. Retrieved from http://ebookcentral.proquest.com.ezproxy.aut.ac.nz/lib/aut/detail.action?docID=284021.

Kwon, J., & Lee, K. M. (2012). *A unified framework for event summarization and rare event detection*. In CVPR (pp. 1266-1273).

KallenberT, L. (2000). *Markov decision processes*. Retrieved from http://webee.technion.ac.il/~adam/MDP/kallenberg.pdf

Kim, C., & Hwang, J. N. (2002). *Fast and automatic video object segmentation and tracking for content-based applications*. IEEE transactions on circuits and systems for video technology, 12(2), 122-129.

Koller, D., Danilidis, K., & Nagel, H. H. (1993). *Model-based object tracking in monocular image sequences of road traffic scenes*. International Journal of Computer Vision, 10(3), 257-281.

Krumm, J., Harris, S., Meyers, B., Brumitt, B., Hale, M., & Shafer, S. (2000). *Multi-camera multi-person tracking for EasyLiving*. IEEE International Workshop on Visual Surveillance (Vol.39, pp.3). IEEE

Computer Society.

King, S. Y. (1988). *Parallel architectures for artificial neural nets*. InSystolic Arrays, 1988., Proceedings of the International Conference on (pp. 163-174). IEEE.

Lee, L., Romano, R., & Stein, G. (2000). *Introduction to the special section on video surveillance*. IEEE Transactions on pattern analysis and machine intelligence, 8, 740-745.

Luckham, D. (2002). *The power of events* (Vol. 204). Reading: Addison-Wesley.

Lomi, V., Tonetto, D., & Vangelista, L. (2003). *False alarm probability-based estimation of multipath channel length*. IEEE transactions on communications, 51(9), 1432-1434.

Liang, G. (2011). *The research of agrithim to reduce the false alarm*. Retrieved from http://www.docin.com/p-771627585.html

Lu, N., Wang, J., Wu, Q. H., & Yang, L. (2008). *An improved motion detection method for real-time surveillance*. IAENG International Journal of Computer Science, 35(1), 1-10.

Liu, Y., Ai, H., & Xu, G. Y. (2001). *Moving object detection and tracking based on background subtraction*. In Multispectral Image Processing and Pattern Recognition (pp. 62-66). International Society for Optics and Photonics.

Li, L., Huang, W., Gu, I. Y., & Tian, Q. (2003). *Foreground object detection from videos containing complex background*. In Proceedings of the eleventh ACM international conference on Multimedia (pp. 2-10). ACM.

Lipton, A. J., Fujiyoshi, H., & Patil, R. S. (1998). *Moving target classification and tracking from real-time video*. In Applications of Computer Vision, 1998. WACV'98. Proceedings., Fourth IEEE Workshop on (pp. 8-14). IEEE.

Li, X., Hu, W., Shen, C., Zhang, Z., Dick, A., & Hengel, A. V. D. (2013). *A survey of appearance models in visual object tracking*. ACM transactions on Intelligent Systems and Technology (TIST), 4(4), 58.

Li, D., Winfield, D., & Parkhurst, D. J. (2005). *Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches*. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops (pp. 79-79). IEEE.

Li, X., Wang, K., Wang, W., & Li, Y. (2010). *A multiple object tracking method using Kalman filter*. In Information and Automation (ICIA), 2010 IEEE International Conference on (pp. 1862-1866). IEEE.

Mittal, A., & Paragios, N. (2004). *Motion-based background subtraction using adaptive kernel density estimation*. In Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on (Vol. 2, pp. II-302). IEEE.

Mark, M. (2005). *The analysis of mathematical modelling*. Retrieved from http://d.wanfangdata.com.cn/Periodical_hqsgkj201415174.aspx

Møller, F. (1993). *A scaled conjugate gradient algorithm for fast supervised learning*. Retrieved from http://ojs.statsbiblioteket.dk/index.php/daimipb/article/download/6570/5693

Marcus, B. (2015). *Neural Network solution*. Retrieved from http://www.neuralnetworksolutions.com/nn/components1.php

Murata, N., Yoshizawa, S., & Amari, S. I. (1994). *Network information criterion-determining the number of hidden units for an artificial neural network model*. IEEE Transactions on Neural Networks, 5(6), 865-872.

Mitchell, T. M. (1999). *Machine learning and data mining*. Communications of the ACM, 42(11), 30-36.

Pustejovsky, J., Castano, J. M., Ingria, R., Sauri, R., Gaizauskas, R. J., Setzer, A., ... & Radev, D. R. (2003). *TimeML: Robust specification of event and temporal expressions in text*. New directions in question answering, 3, 28-34.

Piccardi, M. (2004). *Background subtraction techniques: a review*. In Systems, man and cybernetics, 2004 IEEE international conference on (Vol. 4, pp. 3099-3104). IEEE.

Pilet, J., Strecha, C., & Fua, P. (2008). *Making background subtraction robust*

to sudden illumination changes. In European conference on computer vision (pp. 567-580). Springer Berlin Heidelberg.

Pfoser, D., Jensen, C. S., & Theodoridis, Y. (2000). *Novel approaches to the indexing of moving object trajectories*. In Proceedings of VLDB (pp. 395-406).

Peterfreund, N. (1999). *Robust tracking of position and velocity with Kalman snakes*. IEEE transactions on pattern analysis and machine intelligence,21(6), 564-569.

Radinsky, K., Horvitz, E. (2013). *Mining the web to predict future events*. Proceedings of the sixth ACM international conference on Web search and data mining. ACM, 2013: 255-264.

Rock, I. (1986). *The description and analysis of object and event perception*. Retrieved from http://psycnet.apa.org/psycinfo/1986-98619-010

Romer, K. (2006). *Distributed mining of spatio-temporal event patterns in sensor networks*. EAWMS/DCOSS, 103-116.

Ray, G. (2013). *Reduce the false alarm rate alarm system*. Retrieved from http://cps.cpd.com.cn/n367536/c16619957/content.html

Steenweg, R., Whittington, J., Hebblewhite, M., Forshner, A., Johnston, B., & Petersen, D., et al. (2016). *Camera-based occupancy monitoring at large scales: power to detect trends in grizzly bears across the canadian rockies*. Biological Conservation, 201, 192-200.

Snoek, C. G., & Worring, M. (2005). *Multimedia event-based video indexing using time intervals*. IEEE Transactions on Multimedia, 7(4), 638-647.

Sampson, R. (2011). *False Burglar Alarms 2nd Edition*. Retrieved from http://www.cops.usdoj.gov/files/ric/Publications/e0307265.pdf

Shaikh, S. H., Saeed, K., & Chaki, N. (2014). *Moving Object Detection Using Background Subtraction*. In Moving Object Detection Using Background Subtraction (pp. 15-23). Springer International Publishing.

Stauffer, C., & Grimson, W. E. L. (2000). *Learning patterns of activity using real-time tracking*. IEEE Transactions on pattern analysis and machine intelligence, 22(8), 747-757.

Saunier, N., & Sayed, T. (2006). *A feature-based tracking algorithm for vehicles in intersections*. In The 3rd Canadian Conference on Computer and Robot Vision (CRV'06) (pp. 59-59). IEEE.

Simon, D. (2010). *Kalman filtering with state constraints: a survey of linear and nonlinear algorithms*. IET Control Theory & Applications, 4(8), 1303-1318.

Shotton, D. M., Rodriguez, A., Guil, N., & Trelles, O. (2000). *Object tracking and event recognition in biological microscopy videos*. In Pattern Recognition, 2000. Proceedings. 15th International Conference on (Vol. 4, pp. 226-229). IEEE.

Tseng, B. L., Lin, C. Y., & Smith, J. R. (2002). *Real-time video surveillance for traffic monitoring using virtual line analysis*. In Multimedia and Expo, 2002. ICME'02. Proceedings. 2002 IEEE International Conference on (Vol. 2, pp. 541-544). IEEE.

Tibor, B., Mark, H., Michel, K., Jan, T. (2011). *An Ambient Agent Model for Monitoring and Analysing Dynamics of Complex Human Behavior*. Retrieved from http://content.iospress.com/articles/journal-of-ambient-intelligence-and-smart-environments/ais117

Tao, H., Sawhney, H. S., & Kumar, R. (2002). *Object tracking with bayesian estimation of dynamic layer representations*. IEEE transactions on pattern analysis and machine intelligence, 24(1), 75-89.

William, C. (1992). *Complete and Unabridged 8th Edition*. Retrieved from http://www.thefreedictionary.com/alarm

Weber, L. (1985). *Alarm Systems and Theft Protection*. Retrieved from http://www.strausssecurity.com/residential/security-systems/home-burglary/

Withington, D. J. (1999). *Localisable alarms*. Human factors in auditory warnings, 33-40.

Wang, Y., Doherty, J. F., & Van Dyck, R. E. (2000). *Moving object tracking in video*. In Applied Imagery Pattern Recognition Workshop, 2000. Proceedings. 29th (pp. 95-101). IEEE.

Wang, S. C. (2003). *Artificial neural network*. In Interdisciplinary Computing

in Java Programming (pp. 81-100). Springer US.

Xie, L., Sundaram, H., & Campbell, M. (2008). *Event mining in multimedia streams*. Proceedings of the IEEE, 96(4), 623-647.

Xiaoli, G. (2009). *Researching on Controlling Model of IDS Alarm*. Retrieved from http://ieeexplore.ieee.org.ezproxy.aut.ac.nz/xpls/icp.jsp?arnumber=5362522

Yan, W. Q. (2016). *Introduction to Intelligent Surveillance*. Springer.

Yan, W., & Weir, J. (2011). *Visual Event Computing*. Bookboon.

Yin, Y., Liu, Q., & Mao, S. (2015). *Global Anomaly Crowd Behavior Detection Using Crowd Behavior Feature Vector*. International Journal of Smart Home, 9(12), 149-160.

Yang, T., Pan, Q., Li, J., & Li, S. Z. (2005). *Real-time multiple objects tracking with occlusion handling in dynamic scenes*. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)(Vol. 1, pp. 970-975). IEEE.

Yegnanarayana, B. (2009). *Artificial neural networks*. PHI Learning Pvt. Ltd..

Zhang, Z. X., Li, J. Z., & Li, N. N. (2003). *Detection of moving object using a fusion method based on segmentation of optical flow field and edge extracted by canny's operator*. Acta Electronica Sinica, 31(9), 1299-1302.

Zelnik-Manor, L., & Irani, M. (2006). *Statistical analysis of dynamic actions*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(9), 1530-1535.

Zhou, D., & Zhang, H. (2005). *Modified GMM background modelling and optical flow for detection of moving objects*. In 2005 IEEE International Conference on Systems, Man and Cybernetics (Vol. 3, pp. 2224-2229). IEEE.

Zhou, Z., Wu, D., & Zhu, Z. (2016). *Object tracking based on Kalman particle filter with LSSVR*. Optik-International Journal for Light and Electron Optics, 127(2), 613-619.

Zhu, X. (2011). *Semi-supervised learning*. In Encyclopedia of machine learning (pp. 892-897). Springer US.