DEVELOPMENT OF ANOMALY DETECTION USING VISION SYSTEM

AHMAD HAFIY BIN HARIZAN

BACHELOR OF MECHATRONICS ENGINEERING
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2019

DEVELOPMENT OF ANOMALY DETECTION USING VISION SYSTEM

AHMAD HAFIY BIN HARIZAN

A report submitted in partial fulfillment of the requirements for the degree of Bachelor of Mechatronics Engineering

Faculty of Electrical Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2019

DECLARATION

I declare that this thesis entitled "Design and Development of Anomaly Detection Using Vision System" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature :

Name : AHMAD HAFIY BIN HARIZAN

Date :

APPROVAL

I hereby declare that I have checked this report entitled "Design and Development of Anomaly Detection Using Vision System" and in my opinion, this thesis complies the partial fulfillment for awarding the award of the degree of Bachelor Of Mechatronics Engineering With Honours.

Signature : ....

Supervisor Name : ....

Date : ....

DEDICATIONS

To my beloved mother and father

ACKNOWLEDGEMENTS

First praise is to Allah, the Almighty, on whom ultimately, we depend for sustenance and guidance. Second, my sincere appreciation goes to my supervisor En Mohd Bazli Bin Bahar for his support throughout this project whose guidance, careful reading and constructive comments was valuable. I express my sincerest appreciation for his assistance in every way throughout this semester which undoubtedly without them, this project may not able to come to its completion.

I would also like to express my sincere gratitude to all of my colleagues who never stopped motivating and sharing their knowledge since the very beginning of my study. Special thanks to University Teknikal Malaysia Melaka (UTeM) and everyone that their names have not been mentioned who may have contributed to the completion of this project.

Finally, I'm forever indebted to my beloved parents and family who have always given me never-ending support and encouragement.

ABSTRACT

As crime rate greatly increases over the years along with technological advances on vision systems, surveillance systems are being installed in almost every possible location such as banks, offices, and residential areas. With continuous recording of videos, humans are facing difficulty to monitor them manually for the detection of anomalous activity. In attention to this problem, the proposed work designs a semi-supervised anomaly detection technique in stationary backgrounds to reduce the need of human intervention in detecting anomalies. Segmentation, detection of objects and noise removal in foreground are the main phases of the proposed work. Identification of important target features such as centroid and the bounding box assists in object detection and localization of objects in frames. The performance evaluation of the simulation results based on localization error and rate of accuracy validated that the proposed work could deal with various anomaly activity in a static background scene and successfully detect anomaly without much human intervention.

ABSTRAK

Kadar jenayah kian meningkat seiring dengan kemajuan teknologi pada sistem penglihatan, sistem pengawasan dipasang di hampir setiap lokasi yang mempunyai kebarangkalian jenayah yang tinggi seperti bank, pejabat, dan kawasan kediaman. Dengan rakaman video berterusan, manusia menghadapi kesukaran untuk memantau mereka secara manual untuk mengesan aktiviti anomali. Dalam perhatian terhadap masalah ini, kerja yang dicadangkan ini merangka teknik pengesanan anomali pengawasan separa dalam latar belakang pegun untuk mengurangkan keperluan intervensi manusia dalam mengesan anomali. Segmentasi, pengesanan objek dan penyingkiran hingar di latar depan adalah fasa utama kerja yang dicadangkan. Pengenalpastian ciri sasaran penting seperti centroid dan kotak pengikat membantu dalam pengesanan objek dan penyetempatan objek dalam bingkai. Penilaian prestasi keputusan simulasi telah membuktikan bahawa kerja yang dicadangkan dapat menangani berbagai aktivitas anomali dalam adegan latar belakang statik dan berjaya mengesan anomali tanpa banyak intervensi manusia.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

m	-	Meter
cm	-	Centimeter
MM	-	Mathematical Morphology
GMM	-	Gaussian Mixture Model
FD	-	Frame Differencing
FP	-	False Positive
FN	-	False Negative
TP	-	True Positive
TN	-	True Negative

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

Over the past decade, the rapid growth and exceptional improvements in real-time video analytics is evident. The detection and localization of an anomaly is the primary goal of video analytics in pursuit of recognizing suspicious activities and potential threats by means of fewer or zero human intervention. Anomaly detection using vision system is a prominent research area that includes the identification and classification of human activities into ordinary (normal), irregular (anomalous) activities. The primary task is to locate uncommon events in videos using a manual, semi-automatic or fully automatic recognition system.

Anomaly detection systems can be generally categorized into two main groups: unsupervised anomaly detection which implies a fully automatic recognition system and supervised anomaly detection for manual or semi-automatic system. The goal of this project is to design and develop a semi-automatic system which involves less human intervention compared to manual systems where real-time monitoring by humans is required. On the other hand, automatic and intelligent anomaly detection systems with machine learning capabilities does not require human intervention in decision making.

Anomaly detection has practical applications in many real-life scenarios. Pimentel et al. [1] classify these scenarios in six main domains: image and video processing, medical diagnostics, electronic IT security, industrial damage detection and monitoring, text mining, and sensor networks. This project focuses on the field of image and video processing as the other domains are beyond the scope of this work. Anomaly detection is carried out in the context of image processing and video surveillance using data collected through vision systems that monitor the behavior and activities of target objects [2]. Without the lack of ability to automatically track and analyses in real-time, a human operator had to either properly monitor an abundance of footage data in real time to detect any anomalies or events, or the footage can only be reviewed after the occurrence of an anomalous event.

Anomalous event detection can be done in two ways: firstly, by training the system with normal and anomalous events, and then utilizing prior data information to distinguish future events. Secondly, the dominant set property according to which the dominant behavior (higher occurring frequency and less attention seeking behavior) of the object is regarded as normal behavior while the less dominant behavior is recognized as unusual behavior or an anomaly. Having said that, this project focuses on the latter technique. The target for detecting anomalies is often linked to human behavior in most applications. However, it is difficult to determine what should be regarded as an anomaly when it comes to human behavior. This is because anomalies are subjective concepts defined by humans and varies according to the situation.

Since video cameras generate image data, it is natural for an automatic video surveillance system to be broken down into several image processing functional blocks such as foreground objection, motion detection and tracking, and blob analysis. The aim of this project is the design of moving object detection framework around these blocks which are implemented using Computer Vision techniques and algorithms. In order to tackle image-related issues such as variation of brightness or color information in images and changes in illumination, image processing that usually involves the segmentation of the foreground and the extraction of features is studied. Then, target activity analysis allows preventative acts or alerts when a specific event is detected.

1.2 Motivation

Due to increased security concerns, there is now an increasing demand for automated monitoring systems in the span of the past decade. Technological advances and reduced costs have led to the accelerated deployment of both public and private surveillance cameras. The monitoring task is traditionally carried out by human operators to inspect video feeds from cameras thoroughly. It has been shown, however, that even dedicated personnel receive reduced visual attention after long periods of observation on monitoring. This prohibits their ability to detect and react to possible real-time threats [3], transforming current surveillance systems into mere recording devices used only for analyzing of footage after an event occurrence [4]. For these reasons, in the past decade, real-time event recognition and anomaly detection in using vision system has undeniably become an interesting research topic. Automatic anomaly detection algorithms can help human operators to recognize suspicious events or potential threats helping them to respond appropriately when necessary.

In this context, the recognition of human activity has been widely studied in the literature. Most approaches in this field have definitive designation of certain events and their application is therefore limited to the detection of these events, usually in controlled scenarios. An example of event detection is the infringement of prohibited areas such as ATM area break-ins after working hours. Based on market research by ATMIA and reported in the 2017 annual global fraud and security survey, the percentage of respondents reporting in ATM crime increased from 42% in 2016 to 54% in 2017 [5]. More recently, the focus has been increased on detecting anomalies without explicit modelling. However, occurrence of events of interest in video surveillance scenarios are sparse and difficult to predict making it very difficult to train a system to cover all possible cases of anomalous events. The underlying assumption that anomalous events are characterized by their low occurrence frequency compared to normal events is common to these techniques.

1.3 Problem Statement

Traditional video surveillance systems relying on human operations are unproductive and inefficient because the number monitoring devices exceeds the monitoring capacity of human operators. Anomaly detection in many vision system frameworks requires significant human intervention making it time consuming and not scalable to high volumes of video footage [6]. In general, abnormal events seldom occur in comparison with normal activities. As a result, humans are indeed incapable to continue to analyze the ever-increasing volume of security recordings and hinder the efficiency of the security system due to fatigue and lack of observation or concentration. Subsequently, a huge portion of video is simply stored without review. With the development of anomaly detection system, management of manual labor and storage resource is less demanding as administrators can analyze specific time frames of activity occurrence.

Another challenge of the vague anomaly definition is due to the diversity of abnormal events which contributes to a high false positive decision making. In most constrained environments, abnormalities are well-defined for example the event of any movement in an ATM area after service hours is considered as anomaly. Regardless, anomaly objects in most scenarios are undefined. For example, any objects except for moving cars on a highway can be treated as anomaly. An algorithm for anomaly detection therefore encounters difficulties where it has little information to predict an event until it literally occurs. Consequently, it is a very difficult to develop a good and accurate anomaly detector to detect unknown anomalous objects with very little information on the target of interest.

1.4 Objectives

The purpose of this research is to:

1. Design and develop a supervised anomaly detection framework on static backgrounds using vision system based on foreground detection and classifier principles.

2. Evaluate the accuracy of the proposed framework in detecting objects.

3. Analyze the performance and reliability in detecting moving objects.

1.5 Scope and Limitation

This study is based on the machine vision and image processing. The anomaly detection algorithm is done based on rule-based approach with predefined problem domain rules set as classifiers.

The proposed working system is based on interfaced program and simulation using MATLAB R2019a and V-REP PRO EDU V3.5.0 rev4. The system works with a 1280x720 X/Y resolution vision sensor with 1.00e-01m and 1.10e+01m near and far clipping planes respectively. The performance of the proposed model is evaluated based on the localization error of detected objects on static backgrounds and recognize classified activities

In particular, the surveillance system can monitor both static and moving objects in the environment and is limited to the classifying between predefined normal background and foreground. Furthermore, the evaluation of the prototype was dependent on simulated scenes in the software V-REP PRO EDU V3.5.0 rev4, which could have influenced the results that were obtained.

With anomalies being a rare occurrence, obtaining enough samples to explicitly categorize anomalous events and behavior is difficult. Another vital limitation is to develop a method that can automatically analyze the behavior of multiple moving objects in videos becomes complex especially when the objects overlaps each other. Several algorithms have been developed to track and distinguish individual targets and distinguish their action and activities in the pursuit of detecting anomaly behaviors. However, a majority of the designed algorithm are high in complexity and require an abundance of data thus is difficult to deliver successful results in a short time constraint. Another limitation entails the case when sometimes the vision of a camera in real life is not focused or blurred in the presence of smudges. Consequently, the detection of objects or foreground is hindered with no clear vision of subjects.

This analysis is expected to focus on developing the ideal foreground segmentation and anomaly detection framework using vision system to extend the ability of semi-automatic foreground detection within static backgrounds. Blob analysis which compasses the measurement, size, centroid and bounding box becomes the main parameters in this study. Based on these parameters, simulated scenes are used to study the system's performance in detecting anomalies.

CHAPTER 2

LITERATURE REVIEW

There are numerous research papers about detecting anomalies using vision system. Anomaly detection using vision system is an active research area thus plenty of previous works had been done relating this area. Past works occurred in this area can be generally classified into two categories, supervised methods where they have a training phase and unsupervised methods which they don’t have such an unequivocal training phase. While several methods of detecting anomalies using vision system are highlighted in this chapter, the discussion here focuses more on pursuing supervised video event detection. This chapter firstly starts with an introduction followed by the description and related work of anomaly detection. Then, previous works and background of object segmentation methods are discussed. Finally, previous methods on noise removal and theoretical background of Mathematical Morphology is outlined before summarizing the chapter.

2.1 Anomaly Detection

Anomaly detection is described in [7] as the assignment of discovering substances in a framework that don't fit into a normal example. The term is often used as a synonym for novelty or outlier detection. As Pimentel et al. points out in [1], the term comes from different application regions, and for each of them there is no collaboratively accepted definition. In this problem class, the positive class is called "normal" while the negative class is designated as "anomaly".

There is extensive research in computer vision and multimedia analyzing object behaviors and movements from videos. A variety of framework for anomaly detection are available including frameworks that requires no human intervention. Different researchers applied different methods and techniques to suit different applications or events. Most of the work has focused on extracting useful information including behavior patterns and situations for surveillance analysis through activity recognition and abnormal behavior detection. Some methods focus on the classification of the most common behavior patterns in each scene such as linear and radial patterns.

In a visual system, a normal example or pattern recognition is utilized to decrease and sum up the information data before sorting and classifying the information into normal or abnormal data [7]. To find defects in normal, rehashing and uniform movement of objects, an algorithm for the calculation of the anomaly detection can be implemented. A simple method in solving anomaly detection in vision system is by utilizing high speed cameras and high specifications video capturing hardware which are used in most modern visual based systems to capture frame data and then analyzing them. As stated in [7] this strategy requires costly equipment as well as a broad measure of handling power.

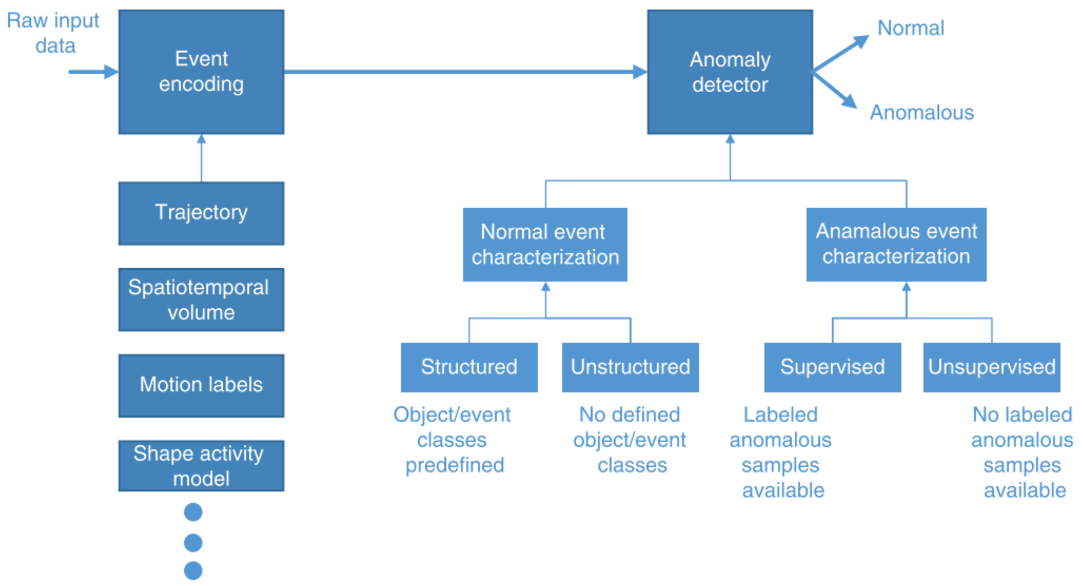

Figure 2.1 Flowchart of anomaly detection in videos [6].

2.1.1 Anomaly Detection Techniques

Unsupervised detection of anomalies refers to the scenario in which there are not enough labels of anomalous samples to precisely categorize an anomalous event class while Supervised anomaly detection refers to the setting where anomalous samples are labelled into known classes [6]. Most of the previous research into anomaly detection focuses on the understanding of behavior through manual system training. Inspired by cognitive science, previous work by Dee and Hogg [1] has demonstrated that by implementing a rule-based approach, unusual trajectories can also be identified which scales the extent of the behaviors a given target could be considered as goal-directed behavior.

A supervised learning framework that utilized the novel incremental one-class learning algorithm as key-component for modelling the distribution of normal motion trajectories occurring in a scene was proposed in [8] where the approval of an administrator is required before any behavior pattern that appears novel with respect to the model is incorporated. This algorithm is described by the control points of cubic spline curves and trained on the trajectories of motion which allows the gradual building of normal behavior model as new examples are added. Moreover, the authors in [8] suggests that for filtering the data needed to create a useful model of normal behavior, only a very low cost in terms of manual classification effort is necessary. This novel supervised or can be considered as semi-supervised learning framework incorporated to any existing anomaly detection algorithm capable of accumulative unsupervised learning.

Hu et al. suggested an approach in [9] where common trajectories are patterned with a more complex multi-levelled grouping technique. Even so, past techniques are commonly constrained to inadequate groups or are intended for offline or non-real-time implementations [10] thus leaning to the approach of utilizing online techniques to determine and track the trajectory-level behaviors of target objects with the combination of Bayesian learning to non-linear motion of models. The trajectory of each targets in a given a video footage is acquired using real-time tracking methods that can distinguish between activities of each target objects and the obstacles. Bayesian inference technique is then implied to calculate the trajectory behavior feature for each agent namely for the detection of anomaly in terms of object motion and behaviors. This approach does not involve offline training and can be used for interactive monitoring regardless of being indoors or outdoors.

Event occurrence in a video always spans numerous counts of frames thus handling video processing frame by frame in video event detection is insignificant [11]. A method had been proposed in [12] to utilize the "hard-to-describe" but "easy-to-verify" aspect of anomaly events without having to create explicit models of normal events. This "hard-to-describe" but "easy-to-verify" property of anomaly events suggests an intuitive two-step solution for their detection. With this method, each event occurrence can be compared with all other events being observed to determine how many similar events exist. When many similar events occurring in a large data set, the event can be decided as normal. An irregular event is considered if there are no occurrence of similar events: although the event is unknown, it is different from the others. Thus, detecting anomalies in a large data set does not require modeling normal events, but rather the ability to differentiate between the occurrence of events and measure their similarity.

To summarize, there are various reliable sophisticated methods available for the detection of anomaly activity. Most of them combines machine learning capabilities such as Bayesian learning method with image processing techniques resulting in advanced anomaly detection algorithm. However, these methods are high in complexity thus demanding higher computational cost that may cause delays in time-constrained projects. That being said, most of the supervised anomaly detection techniques implements manual training or labelling of samples are better suited considering the limitations present.

2.2 Object Segmentation

In accordance with the approach to detect anomaly using vision system, it is necessary to employ a method for object detection. Object segmentation is the first stage of object recognition, where the anomaly objects are segmented from the background image. Image segmentation can be described more precisely as the process of labeling each pixel in an image so that pixels with the same label share certain features.

The process of object segmentation is done frame by frame in the video sequence to extract the target object [13]. The object segmentation can be categorized into two types of segmentation depending on the mobility of cameras or setup of hardware which are the static segmentation and mobile segmentation as represented in Figure 2. This paper focuses on approaches to detect anomaly using static vision systems thus static camera segmentation applied.

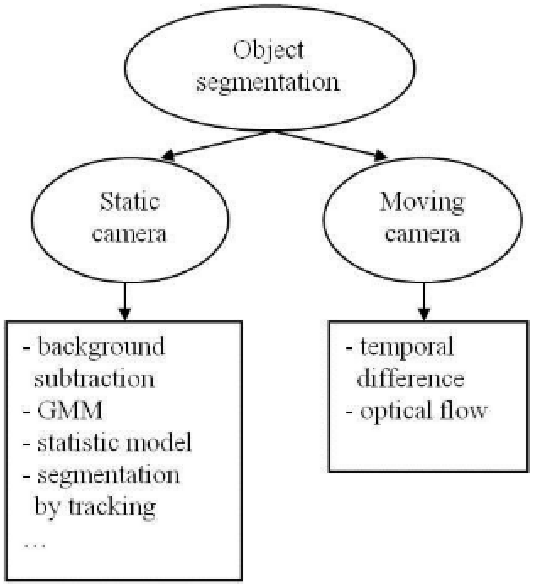

Figure 2.2 Division of object segmentation categories [13].

For segmentation in stationary camera scenes, the camera is in a fixed specific position and angle. Naturally, a background model is built in advance since the background never moves or changes. This allows the segmentation of foreground from the image of the background model. Therefore, the point of view or frame of reference of the background scene and object are fixed.

2.2.1 Background Subtraction

Due to it being simple and effective, background subtraction [12,13,14] are the most prevalent technique for stationary camera video segmentation. In this method, the image of a stationary background model is originally established without any foreground object. Then, to obtain the foreground objects, the present image in the video sequence being analyzed is then removed from the background image.

Previous work [17] applied frame differencing method. Frame differencing is the simplest form of background subtraction. This is done by comparing the pixels values from current frame with the previous image set as the background [17]. This method depends heavily on predefined threshold value which determines whether a particular pixel is foreground or background.

The advantage of this method is its highly adaptive background model and less computational load. In fact, the background model is based on the previously stored frame thus it can adapt to changes in the background at 1/fps (one frame period) [18]. However, if the objects remain motionless for more than one frame period, it will be considered as part of the background. Besides that, objects that are moving consistently are likely to be detected as background.

However, this background subtraction method has a high sensitivity towards unstable shifting illuminations. In order to handle this issue especially on dynamic or waving backgrounds, a method was implied in [15]. The method studies about sequential changes in the background of the scene in terms of the image vector distribution. Occasionally, in consequence of its simplicity and efficiency some pixels tend to be incorrectly classified.

2.2.2 Gaussian Mixture Model (GMM)

Previous work had proposed several other methods with higher complexity in terms of background modelling from an accumulated batch of background images. This approach is developed to a high degree of complexity making it the most sophisticated technique used for background subtraction [9,10,14]. A single Gaussian likewise a mixture of Gaussians can be used to create the model of the background to accommodate different background scenarios making it adaptable to multi-modal environments. Generally, GMM is learned by an iterative method called the expectation-maximization (EM) algorithm. This method determines the maximal probability or maximum a posteriori (MAP) estimates of parameters in statistical unobserved latent variable dependent models.

A particular pixel with higher probability of a pixel value in GMM is more likely to be designated to the background [13]. The probability considering a current pixel value X_t is equated in [6] as follows:

$$\boldsymbol{P(X_t) = \sum_{k=1}^{k} w_{k,t} * \eta(X_t\,, \mu_{k,t}, \Sigma_{k,t})} \tag{2.1}$$

where $w_{k,t}$ is the weight or relative importance of the kth Gaussian distribution with mean $\mu_{k,t}$ and covariance $\sum_{k,t}$.

Therefore, a pixel of the image is classified to be part of the foreground object for the image sequence if the probability of the pixel value is lower than the predetermined threshold. The top B modes are marked as background modes, and B is derived in [6] as:

$$\boldsymbol{B = argmin_b\left(\sum_{k=1}^{b} w_k > T\right)} \tag{2.2}$$

where T is a measure of the minimum portion of the data that should be accounted for by the background.

The GMMs are constructed in [17] across a variety of colors and textures. The k-means clustering algorithm was applied by Permuter et al. in [17] to reduce the high calculation of the EM algorithm. While the probability value achieved by this is slightly lower than that achieved by the EM algorithm, the difference in performance is insignificant with the same amount of data involved in the background model training.

Sometimes the dimensions of the features need to be lessened to avoid the frequently encountered problems of singular covariance when training data is insufficient. In [18] the Gauss mixture vector quantization (GMVQ) classifier was developed to reduce this problem when the coefficients of a linear image transformation are feature vector elements. The Lloyd algorithm is implied to train the (GMVQ)-based classifier by repeatedly appointing each training vector to the Gauss mixture component, minimizing the QDA distortion follow by the updating of the Gaussian mixture parameters.

In short, GMM is very fast and scales well to large amounts of data plus it can be altered effectively to outline a more complex background. Although the high computation costs associated with EM training impose a tradeoff, no tuning of any hyper-parameters are needed, only the threshold. However, the viewpoint of the object and the background are fixed and illumination changes are neglected. Thus, background subtraction is adequate due to its simplicity and efficiency.

2.2.3 Statistical Model

In addition to GMM, methods are also proposed for more sophisticated background modelling. One common method is to create the background model as a statistical model that includes changes in intensity and chromaticity for each pixel [20].

In [21] the modeling of each pixel depends on parameters like brightness distortion, chromaticity distortion, brightness distortion variation and chromaticity distortion variation. Each pixel in the current image can be categorized into either the original background, the highlighted background, the moving foreground or the shadow based on the brightness and chromaticity distortion thresholds.

To be precise, if brightness and distortion of chromaticity are small, it is classified as the background pixels while if the chromaticity distortion is small but has less brightness it is considered as the shadow pixels. Meanwhile. if it possesses a high chromaticity distortion it is categorized as the foreground pixels.

In general, the statistical models are more efficient in the construction of the background model compared to GMM and can be used to segment shadows on top of the foreground objects. However, the complexity of obtaining the parameter values is a major drawback when considering the time-constraint and equipment limitations.

2.3 Noise Removal

This process of image enhancement can be done in the spatial domain or in the frequency domain. Low pass, high pass, band pass and clamp filters are designed to eliminate noise from the image or improve the image quality. If the filter or kernel size is small, filtering in the space domain can be carried out more effectively. If the size of the kernel or convolution mask is large, the space domain is time-consuming. In such a case, filtering in the frequency domain will be more effective. Since the convolution multiplies here, even with large kernel sizes the processing would be more efficient.

In some applications, the detection of the shape and structure of an image or a pattern within an image is needed. Morphological algorithms are more relevant for this type of image analysis. By means of dilation and erosion processes, a particular shape can be extracted and analysed. The specific pattern size can be enlarged by dilation process and compressed by erosion. Morphological filtering would be achieved by combining both dilation and erosion processes. The input image can be smoothed together with the noise removal by opening and closing operations. Other morphological algorithm applications include edge detection, feature detection, image segmentation, and object counting. Unlike spatial filtering using convolution masks, these operations are not linear.

2.3.1 Mathematical Morphology

Mathematical Morphology is defined by the authors in [22] as a theory and technique for the analysis and processing of image geometric structures which depends on set theory, lattice theory, topology, and random functions. Four basic operations applied are Erosion, Dilation, Opening and Closing. Operations in morphological image processing operates upon these fundamentals.

The basic idea in binary morphology is to test an image with a simple, predefined form and draw conclusions about how this form fits or misses the image. This simple controlling "sample" referred to as a structuring element is a binary image that albeit the existing of eight adjacent pixels version, it usually consists of four adjacent pixels. Figure 3 below shows both types of structuring element.

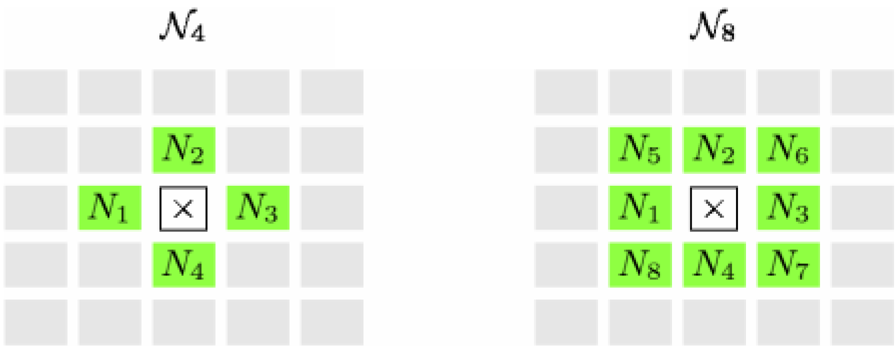

Figure 2.3 Models of structuring element [22].

They can be used for tasks such as extraction of features, noise filtering in an image, segmentation of images i.e. separating foreground objects from a background image and texture analysis operations [3].

2.3.1.1 Erosion

Erosion is a process in which the structuring element is placed on the image input and the center pixel is replaced by 1 if each pixel in a structuring element corresponds to each pixel in the image area. Objects present in a binary image is compressed and shrank in this operation. The limit of shrinking the object depends on the structuring element as a controller. If this process is repeated twice or three times, the area with an interesting pattern size is further reduced. Mathematically, where $\boldsymbol{A_{-b}}$ denotes the translation of A by $-b$ erosion is defined in [22] as:

$$\boldsymbol{A \ominus B} = \bigcap_{\boldsymbol{b \in B}} \boldsymbol{A}_{-b} \quad \textbf{(2.3)}$$

This shows that the erosion of image $\boldsymbol{A}$ by the structuring element $\boldsymbol{B}$ is the set of all structuring element locations that are not overlapping with the background of $\boldsymbol{A}$. An example of erosion operation using the structural element $\boldsymbol{H}$ is shown in figure 4 below.

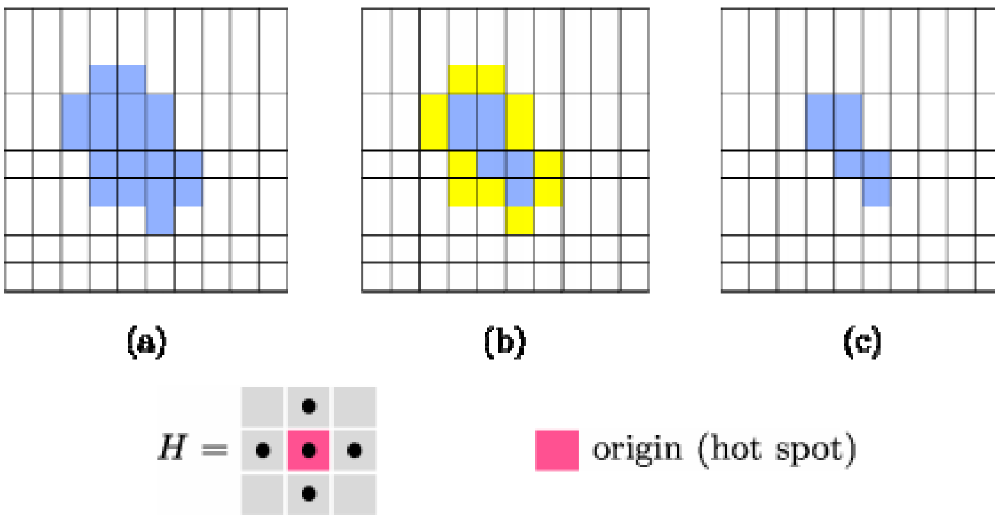

Figure 2.4 Erosion operation. (a) Original pixels, (b) Yellow pixels will be eliminated, (c) Resultant pixels [22].

2.3.1.2 Dilation

Dilation is a process where the structuring element is placed on the input image and the centre pixel is replaced by 1 if at least one pixel matches the structuring element. If dilation in all directions or horizontally and/or vertically is required, the structuring elements are appropriately selected. The size and design of the structuring element play an important role in the process of dilation. Objects of interest in a binary image are expanded through this operation. The size and shape of the structuring controller is vital as it becomes the boundary of the enlargement process. The dilation of image A by the structuring element B is defined in [22];

$$\boldsymbol{A \oplus B} = \bigcup_{\boldsymbol{b \in B}} \boldsymbol{A_b} \quad \textbf{(2.4)}$$

Thereby, the dilation of image $\boldsymbol{A}$ with the structuring element $\boldsymbol{B}$ is the set of all structuring element locations where the symmetry of $\boldsymbol{B}$ covers at least a fragment of image $\boldsymbol{A}$. An example of dilation operation using the structural element $\boldsymbol{H}$ is shown in figure 5 below.

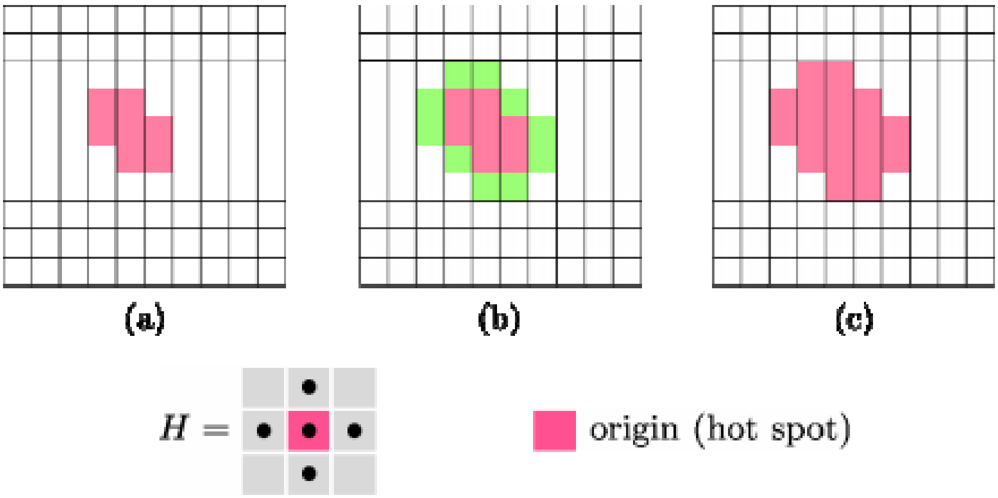

Figure 2.5 Operation of dilation. (a) Original pixels, (b) Green pixels will develop and merge, (c) Resultant pixels [22].

2.3.1.3 Opening

Morphological opening is the Union of all possible structuring element $\boldsymbol{B}$ locations where $\boldsymbol{B}$ fits completely within $\boldsymbol{A}$. The mathematical representation for morphological opening operation is given by;

$$\boldsymbol{A} \circ \boldsymbol{B} = (\boldsymbol{A} \ominus \boldsymbol{B}) \oplus \boldsymbol{B} \qquad \textbf{(2.5)}$$

This operation is simply the erosion of $\boldsymbol{A}$ by a structuring element $\boldsymbol{B}$ followed by an output dilation by the same structuring element as depicted in figure 6 below.

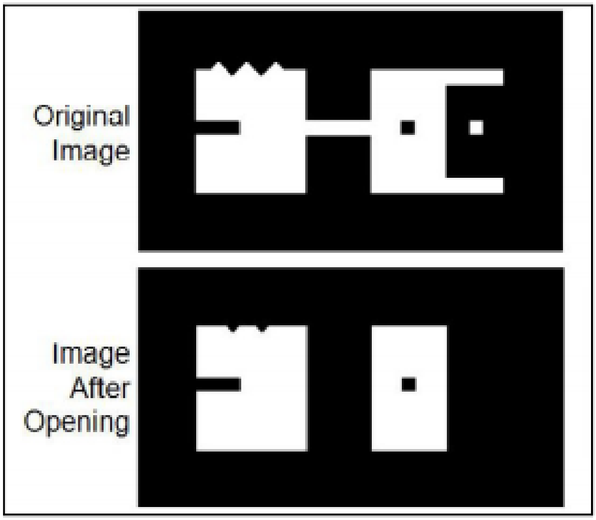

Figure 2.6 Example of morphological opening operation with a 20-pixel square as structuring element [9].

2.3.1.4 Closing

Morphological closing process is a process where in total, the union of all possible locations of structuring element $\boldsymbol{B}$, where $\boldsymbol{B}$ fits completely outside $\boldsymbol{A}$, is complementary and can be represented mathematically as follows:

$$\boldsymbol{A} \circ \boldsymbol{B} = (\boldsymbol{A} \oplus \boldsymbol{B}) \ominus \boldsymbol{B} \tag{2.6}$$

This operation is simply the dilation of $\boldsymbol{A}$ by a structuring element $\boldsymbol{B}$ followed by the erosion of the output by a similar structuring element. An example is shown in Figure 7 below.

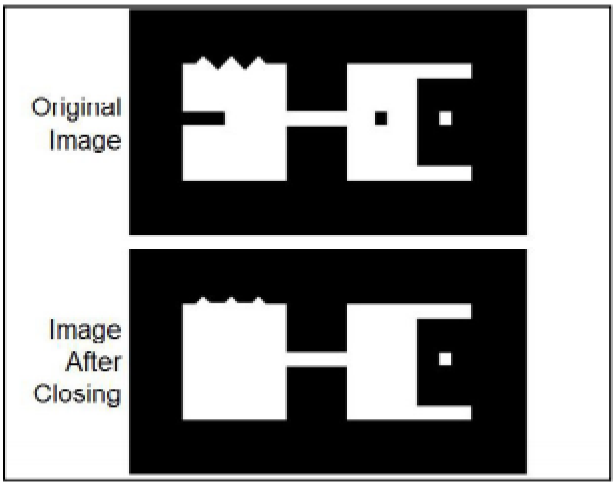

Figure 2.7 Example of morphological closing operation controlled by a 20-pixel square structuring element [9].

2.4 Summary

Conclusively, even with its high sensitivity, a simple background subtraction method is preferred compared to the more sophisticated GMM with its higher computation costs. Besides that, GMM can also be altered effectively to outline a more complex background plus only tuning of the threshold is enough to model backgrounds well. For noise removal, Mathematical Morphology is more relevant for the detection of the shape and structure of an image within an image which is one of the purposes of this work.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The development of anomaly detection systems is envisioned to make the monitoring system intelligent and smarter. The main objective is to detect and recognize suspicious or abnormal activities in videos and trigger warning every time any kind of anomaly occurs. This anomaly detection program simply compares the input background frame with the current frame.

The proposed work can recognize single or multiple activities in a single video. Background subtraction and Mathematical Morphology is applied in background modelling and foreground detection. In an event that any odd action happens, the system labels each detected anomaly and alerts users.

This chapter will discuss on the proposed methodology of this work. Detailed procedures of background subtraction in order to distinguish between background and foreground for object detection are described. In addition, the classifier principle which in this case is the predefined if-then rule based classifier for anomaly recognition will be explained in this chapter.

3.2 Methodology Flowchart

The flowchart below illustrates the flow of proposed anomaly detection framework which can be divided into three main components: pre-processing stage, feature extraction stage and recognition stage.

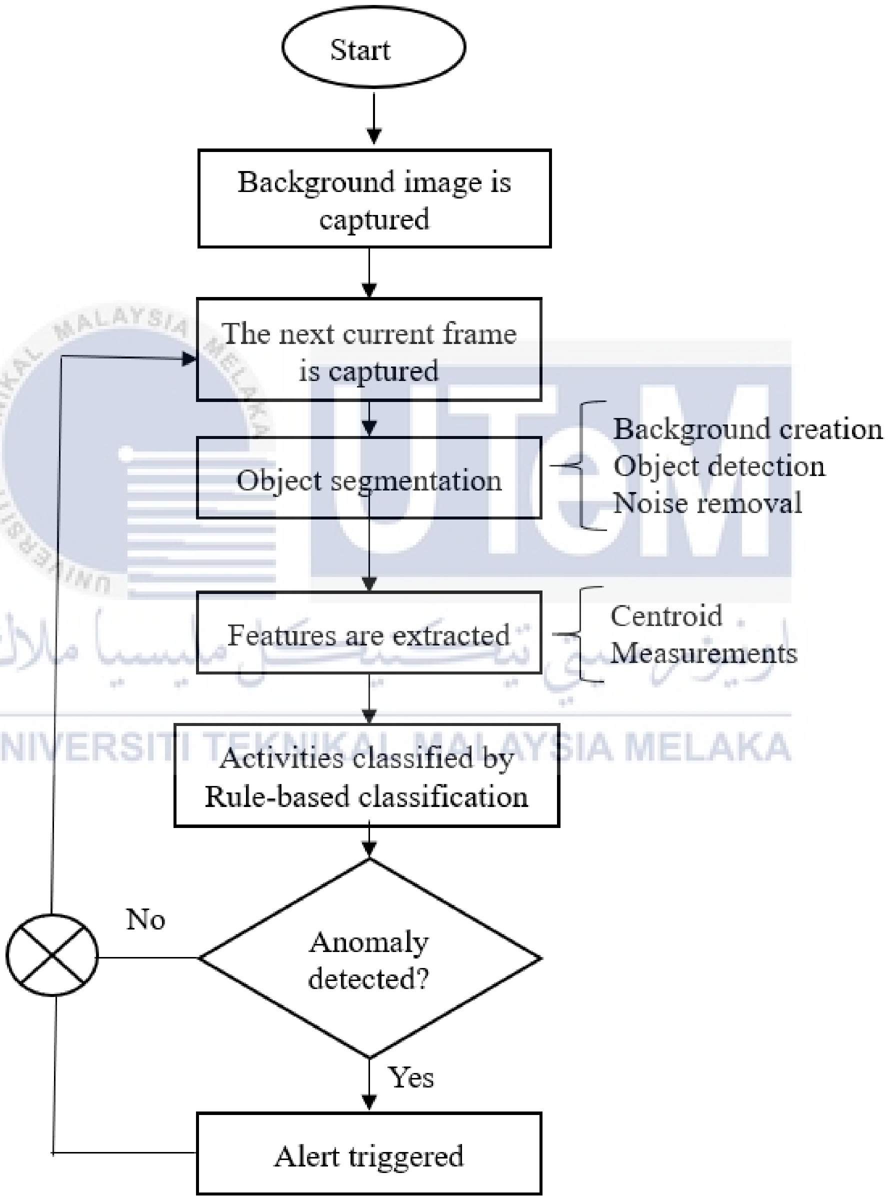

Figure 3.1 Flowchart of the proposed framework

3.3 Pre-processing Stage

The pre-processing stage consists of the extraction of input frames, modelling of background, detection of object and filtering of noise.

Initially, input video data is extracted, and the video is converted into frames. Next, background modelling or background subtraction is accomplished by frame differencing as described in [17]. Then blob analysis is performed for the detection of foreground together with noise removal or smoothing which is accomplished via Morphological Opening method.

3.3.1 Foreground Detection by Frame Differencing

Foreground detection is achieved using frame differencing on consecutive frames in the image acquisition loop, which identifies difference in pixel values - usually from moving objects - from a background frame. In this case, the background frame is the captured frame at the start of the simulation where simulation time, $t = 0$. In the proposed method, the frame of the background image is first extracted. Then, consecutive frames are extracted at one frame per second.

The method applied basically employs the image subtraction operator. Background frame and current frame are taken as input and the image subtraction operator basically subtracts their pixel values producing a third image output from the resultant pixels. The output image which is the resultant pixel difference will be compared with a threshold value.

This technique depends heavily on the defined threshold value. Threshold is a value to determine whether the given pixel is foreground or background. If the absolute difference between the two frames is greater than the threshold, the pixel will be assumed as part of the foreground as in (3.1). Otherwise, it is considered as background pixel.

$$|frame_t - frame_{t=0}| > T \tag{3.1}$$

Several key challenges are faced in developing a good foreground detection algorithm. First, it must be robust against changes in illumination. Second, it should avoid detecting other background objects such as moving leaves, rain, snow, and shadows cast by moving objects. At this point, one remaining issue is that the output image from the differenced frames are likely to be noisy hence some form of smoothing is therefore needed to overcome localization error and remove noise leaving behind our object of interest. In order to develop a high-performance anomaly detection algorithm, the image obtained is preprocessed to eliminate unwanted disturbances. This includes application of a thresh-holding operation and Mathematical Morphology operations.

3.3.2 Noise Removal using MM

To eliminate unwanted disturbances, the filtered images are preprocessed to avoid detecting non-stationary background objects such as shadows from moving objects. Noise removal and smoothing of image is done by Mathematical Morphology (MM) method. By performing the morphological opening operation, the smoothed image minimizes inaccuracies of object detection in the background

Boolean algebraic operation is applied to the mapping of selected regions in a digital image. Groups of neighboring grid centers (pixels) are viewed as sets, and Boolean algebra is applied to them. To apply such methods the digital image is converted into binary image where the points of interest sections are labelled as one and the rest labelled as zero.

Referring to (2.5), morphological opening operation is performed and controlled using a square shaped 3-pixel flat morphological structuring element. A flat structuring element is a binary valued neighborhood, either in 2-D or multidimensional. This structuring element is represented as an object called "strel" in MATLAB which is an essential part of morphological dilation and erosion operations. The strel function which is applicable to both binary and greyscale images creates a square structuring element whose width can be defined in pixels.

The morphological open operation here is done by performing an erosion followed by a dilation of the output. The same structuring element is used for both operations. The true pixels of the structuring element are included in the morphological computation and applied to (2.3) and (2.4) while the false pixels are excluded. The pixel of the image being processed is identified by the origin or the center pixel of the structuring element.

3.3.3 Experimental Setup

To execute the experiments, a scene is created for anomalous activity detection by using V-REP PRO EDU V3.5.0 rev4 software. By assuming that the lighting and illumination of a given area of scene is consistent and considered constant, the images are captured by a stationary vision sensor with a 1280x720 X/Y resolution with 1.00e-01m and 1.10e+01m near and far clipping planes respectively. Objects and motion are generated in the software with programmed movements in a manner which they will avoid much overlapping since this may affect the experimental results. Table 3.2 below shows the experimental procedures of creating the activity scene.

Table 3.1: Procedure to create dataset.

Step	Procedure
1	Static background scene is produced.
2	The background image is captured and stored.
3	Stationary foreground object is added to the scene.
4	The current frame of the updated scene is captured.
5	Steps 1-4 are repeated replacing the foreground with a moving object.
6	Stop.

The experiments are done on a system configuration with an AMD Ryzen 5 1600 processor and 16GBs of RAM. The program code is run and simulated in MATLAB R2019a with in synchronous mode with V-REP PRO EDU V3.5.0 rev4. The procedure for video pre-processing stage is listed in Table 3.3 below.

Table 3.2: Experimental procedure for video pre-processing.

Step	Procedure
1	The current image frame is extracted.
2	Background subtraction is performed by differencing background frame and current frame.
3	The resultant image is converted to grayscale.
4	Foreground is filtered using MM.
5	Clean foreground is obtained and displayed.
6	Steps 1-5 are repeated for the rest of the captured frames.
7	Stop.

3.4 Feature Extraction Stage

The extraction of features is a very important step in any anomaly recognition techniques for the accurate localization of various activities. This is because connected components of objects such as pixel size or intensity can be extracted and compared to differentiate between objects of interests. After pre-processing, multiple features are extracted from the consecutive frames. Parameters such as measurement, size, centroid and bounding box are considered as the main features for classifying normal and anomalous activities. Once image properties and features of objects are extracted, objects with connected regions of properties are tracked. Practically, tracking of those quantities that are likely to provide strong indicators of anomalous events are requisite for anomaly detection.

Another considered feature is the centroid or the center of gravity. The shape centroid is the arithmetic mean or the average of all points in the shape. Simply put, regarding image processing and computer vision, the centroid is the weighted average of all pixels in the shape. Consider a shape consisting of n distinct points $x_1 \dots x_n$ and $y_1 \dots y_n$, then the centroid in x-direction, x_c and in y-direction, y_c is denoted in (3.1) and (3.2) respectively. n is the number of pixels in the blob and x_i are the x coordinates of the n pixels whereas y_i are the y coordinates of the n pixels. The centroid is then plotted over the detected object or blob indicating their corresponding pixel location.

$$x_c = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{3.2}$$

$$y_c = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{3.3}$$

Additionally, a blob's bounding box is calculated. The blob's boundary is the minimum space in the rectangle containing the blob. It can be defined by running through all pixels that belong to a blob and finding the minimum and maximum x and y values of the four pixels. Once all four values have been obtained, the bounding box width is given as $x_{max} - x_{min}$ and the height as $y_{max} - y_{min}$. The calculated bounding box is considered as Region of Interest (ROI).

On top of that, the height of the object is also measured from the bounding box to get a special calibration factor. The height of the object in real world units is compared to the computed height from the bounding box in terms of pixel values. The calibration factor is used for the conversion of pixels to unit length and is computed by;

$$\boldsymbol{Calibration\ Factor} = \frac{\boldsymbol{Unit\ height\ (cm)}}{\boldsymbol{Number\ of\ pixels\ (pixels)}} \tag{3.4}$$

Conversion from pixels to unit length in vertical and horizontal directions is accomplished by multiplying the number of pixels with the calibration factor in that direction [27].

3.4.1 Experimental Setup

The program code is run and simulated in MATLAB R2019a on a system configuration with an AMD Ryzen 5 1600 processor and 16GBs of RAM. The procedure for feature extraction stage is as listed in Table 3.4.

Table 3.3: Experimental procedure for feature extraction.

Step	Procedure
1	Image region properties is measured.
2	Properties of connected region of the filtered image is computed.
3	Centroid is plot over the filtered image.
4	Detected objects are highlighted with red rectangle.

Step	Procedure
5	The rest of the image frames are processed.
6	Stop.

3.5 Recognition Stage

In the proposed method, a rule-based classification was used to recognize and categorize activities as normal or anomalous. At this stage, activity classification is based on pre-defined set of rules. Any kind of training samples either labelled or unlabeled samples are not needed for this method. Meanwhile, some external knowledge or rules of the region of interest are needed to create a model. The biggest limitation of this approach is that it relies heavily on the predefined set of rules by the user.

In this method, a set of rules or pre-determined threshold is defined initially. It includes if-then rules for the decision making on whether the recognized activity is abnormal. This work focuses on anomaly detection within static confined backgrounds thus any difference in current frame compared to the background frame is recognized as an anomalous activity. A warning message is triggered to alert users in the presence of an activity and the detected anomaly activity is bounded by a red colored bounding box.

In order to test the result of the anomaly detection framework, data of multiple activity occurrences is required thus simulation of different types of scenes and activity is done in V-REP software. Several scenes of moving and static objects in the software is used to simulate the scene and activities. The reliability of the approach is then tested based on localization error. The accuracy to estimate positions of detected objects is measured by comparing the exact locations of objects in real world units with the obtained locations in the program.

3.5.1 Experimental Setup

Table 3.5 lists the procedure for testing of the recognition stage. The program code is run and simulated in MATLAB R2019a on a system configuration with an AMD Ryzen 5 1600 processor and 16GBs of RAM.

Table 3.4: Experimental procedure for feature extraction.

Step	**Procedure**
1	Number of objects bounded by bounding box is counted.
2	If number of objects is more than or equal to one, “Anomaly Detected” message is displayed. Else “Normal” message is displayed.
3	The video is processed until the end.
4	Stop.

3.6 Summary

As a summary, methodology of the proposed work can be divided into three key stages: pre-processing stage, feature extraction stage and recognition stage. Starting with the pre-processing stage, objects are detected via frame differencing and noise removal is done by using mathematical morphology. Next, different features such as centroid and dimensions of the objects are analyzed and extracted during the feature extraction stage. Finally, rule-based classification approach is implemented during the recognition stage to recognize an anomaly thus triggering an alert.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter discusses all achieved results together with the analysis from the experiments and techniques done on the anomaly detection framework. Different types of graphs and diagrams are used to distinguish the results obtained for each experiment. Each of these results are interpreted and represented with precise values and appropriate comparison.

4.2 Anomaly Detection

This subsection represents the experimentations done to validate the anomaly detection framework. Three different scenes were simulated in V-REP Pro Edu and the program is tested via synchronous simulation with MATLAB R2019a. The scenes are then captured using a vision sensor with a resolution of 1280 x 720 pixels. The first experiment is done with only one moving object before adding another moving object and testing the program with a stationary object added to the scene.

The results obtained are for the clean segmentation of foreground, detection of objects and correct alert triggers. Similar structuring element which is a square structuring element with width of 3 pixels is used in all the experiments. Figure 4.1 and Figure 4.2 shows the results achieved for detecting anomalies of single and multiple activities respectively.

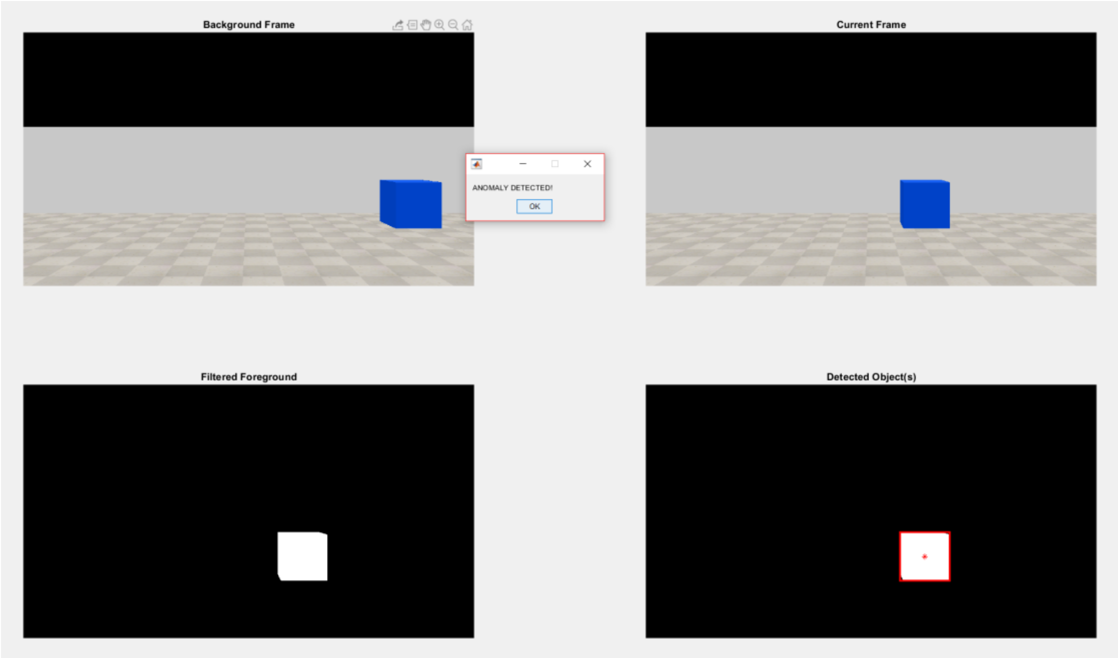

Figure 4.1 Result showing the anomaly detected within a single activity.

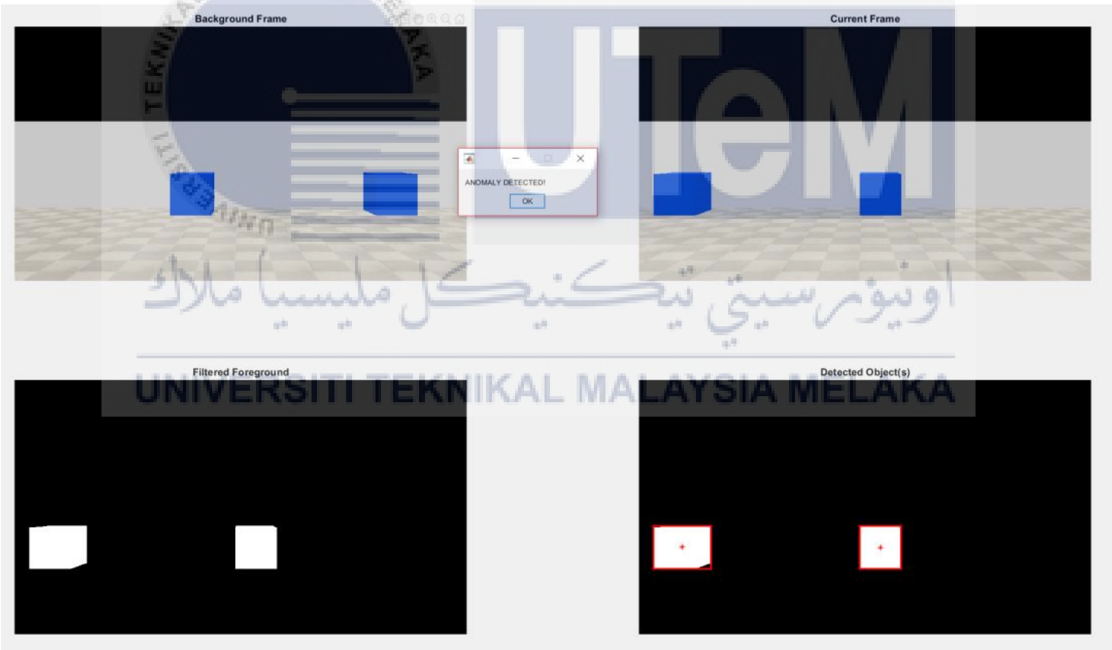

Figure 4.2 Result showing the anomaly detected within multiple activities.

The top left panel is the captured background frame of the scene where simulation time,t is 0. The background is static with minimum illumination changes. The top right panel is the next frame of the scene captured at t=1. The blue boxes in the scene are moved at a constant velocity of 0.3 m/s. Consecutive frames of captured at 1 frame per second. The bottom left panel is the filtered foreground or clean foreground obtained after performing noise removal by Morphological Opening Method. Unwanted noise or small objects are removed leaving only objects of interest. Finally, the detected object is shown in the bottom right panel where it is bounded with a red rectangle box and the centroid is plot. Thus, when the number of detected objects is equal to or more than 1, the alert is triggered warning the user of the occurring anomalies.

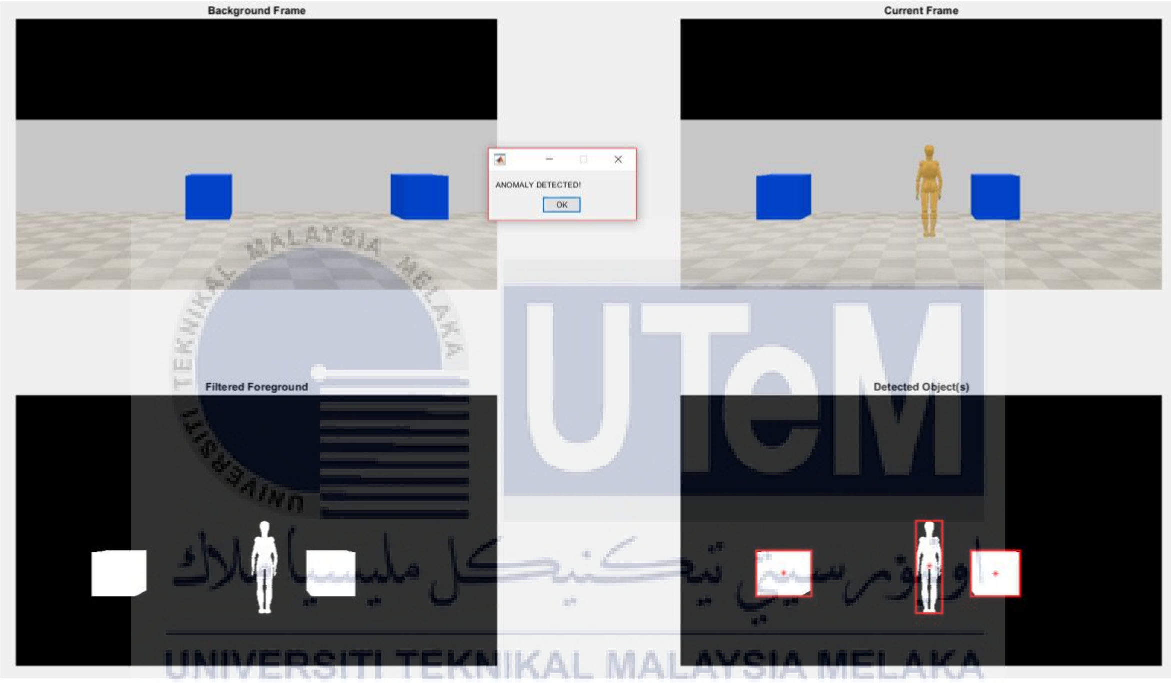

Figure 4.3 Result showing the stationary object is detected as anomaly.

Figure 4.3 above shows even with multiple activities occurring, the system is able to detect occurring anomalies as long as there is difference in pixel intensity. However, if pixels of similar intensity values overlap with each other, it will affect its accuracy in detecting objects as shown below.

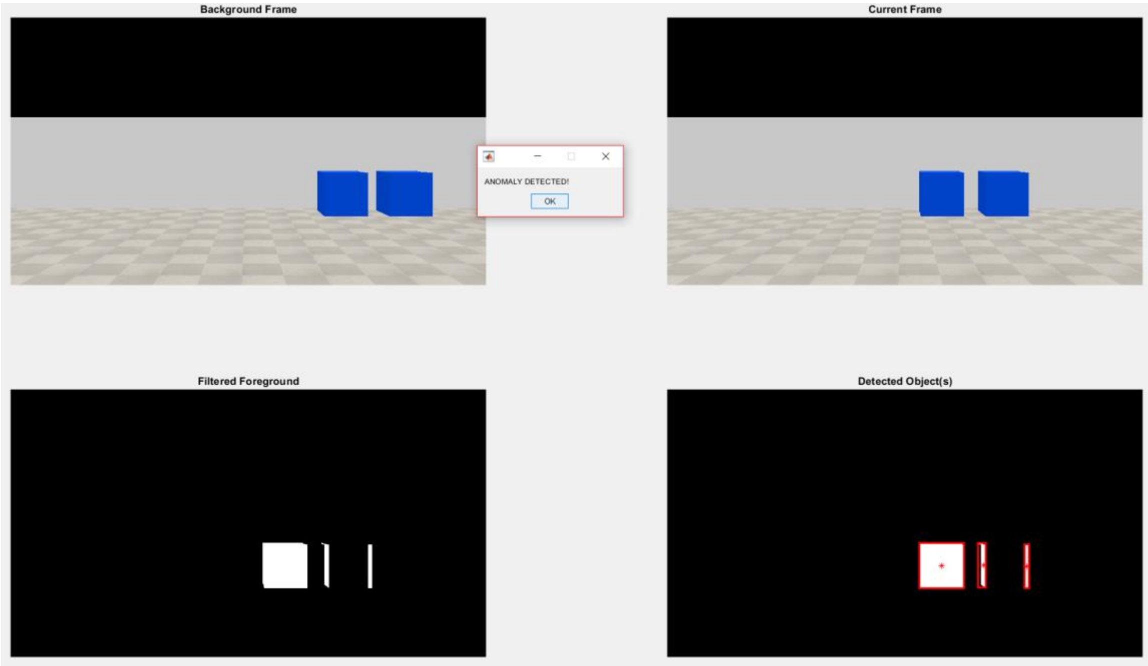

Figure 4.4 Result obtained when objects with similar pixel intensity overlaps.

In Figure 4.4, the blue boxes are moved at a slower speed of 0.1m/s. Therefore, at the first frame where t=1, one of the boxes had only moved a bit. This causes the change of pixel values for only a part of the box thus the output detects a false positive (FP) and false negative (FN) anomaly. The occurrence of FP and FN affects and reduces the accuracy of the model. This shows that the system is sensitive to illumination changes. Nevertheless, based on previous results, the proposed method is proved capable of detecting foreground objects and anomalies considering the project's scope and limitations.

4.3 Accuracy of the Approach

The experiment is done to measure the accuracy of the proposed framework. To measure the accuracy of the approach, the obtained foreground of the current frame is compared to the filtered foreground after performing noise removal. The accuracy of the model is determined by the ratio of correctly recognized anomaly over the total occurrence of activity or in terms of false positive (FP), false negative (FN), true positive (TP) and true negative (TN) as denoted in (4.1).

$$\% \, \boldsymbol{Accuracy} = \frac{(\boldsymbol{TN} + \boldsymbol{TP})}{(\boldsymbol{TN} + \boldsymbol{TP}) + (\boldsymbol{FN} + \boldsymbol{FP})} \times \mathbf{100} \tag{4.1}$$

Here, true positives (TP) refer to the positive objects or anomalies that were correctly labeled by the classifier while true negatives (TN) are the negative anomalies that were correctly labeled by the classifier. False positives (FP) are the negative anomalies that were incorrectly labeled as positive and False negatives (FN) are the positive anomalies that were mislabeled as negative. The tests were simulated in MATLAB for three different type of scenes produced in VREP. Results for the experiments 1,2 and 3 are shown in Figure 4.5, Figure 4.6 and Figure 4.7 respectively.

Figure 4.5 Foregrounds obtained before and after noise removal for a single moving anomaly.

The left panel is the foreground obtained for the current frame before performing noise removal and the right panel is the filtered foreground after noise removal is done. All the tests were done using the same square shaped 3-pixel flat morphological structuring element for the noise removal using Morphological Opening Method. Table 4.1, Table 4.2 and Table 4.3 shows the number of TP, TN, FP and FN obtained for each test.

Table 4.1: Results for test 1.

Test 1	**TP**	**TN**	**FP**	**FN**
Before	1	0	2	0
After	1	0	0	0

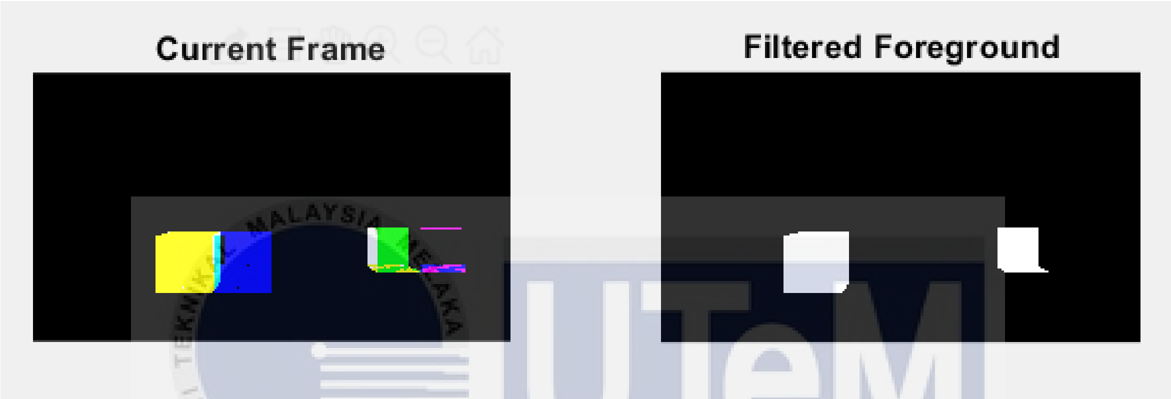

Figure 4.6 Foregrounds obtained of test 2 before and after noise removal for a multiple moving anomaly.

Table 4.2: Results for test 2.

Test 2	**TP**	**TN**	**FP**	**FN**
Before	2	0	2	0
After	2	0	0	0

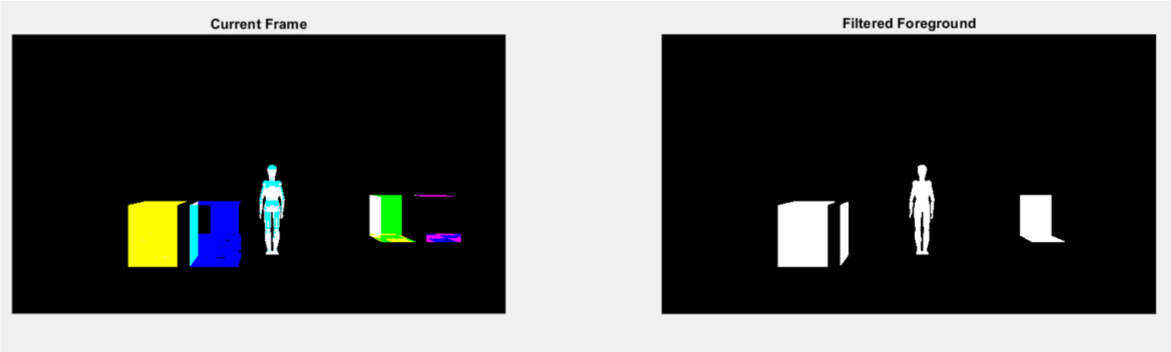

Figure 4.7 Foregrounds obtained of test 3 before and after noise removal for both static and moving anomalies.

Table 4.3: Results for test 3.

Test 3	TP	TN	FP	FN
Before	3	0	3	0
After	3	0	0	0

Table 4.4: Accuracy of the approach.

Test	%Accuracy Before	%Accuracy After	%Accuracy Increase
1	33.33	100	66.67
2	40	100	60
3	50	75	25

Table 4.4 shows the measured accuracy of the approach before and after performing noise removal using Morphological Opening method and the percentage of increase in accuracy. The accuracy of the approach significantly increases by performing noise removal for a clean segmentation of the foreground. The execution of Morphological Opening removes unwanted blobs to distinguish detected objects leaving only objects of interest in the obtained clean foreground.

4.4 Reliability of the Approach

The reliability of the system is evaluated based on the act of localization. Localization is the estimation of the true location of an object in space and is characterized by a certain amount of inherent uncertainty and operational bias that results in estimation errors.

This part represents the significance of parameters such as the motion velocity and their effects on the system. This is because parameters play a significant role in affecting the performance of the algorithm for various problems. The results of the parameter tuning are then compared thoroughly.

4.4.1 Velocity

To study the performance of the system, the velocity of the moving object is tuned and the output is evaluated based on the localization error. To validate that, experimentation was done with the object's pixel intensity, shape and size set to be constant. All values are obtained at the third frame and illumination changes are neglected.

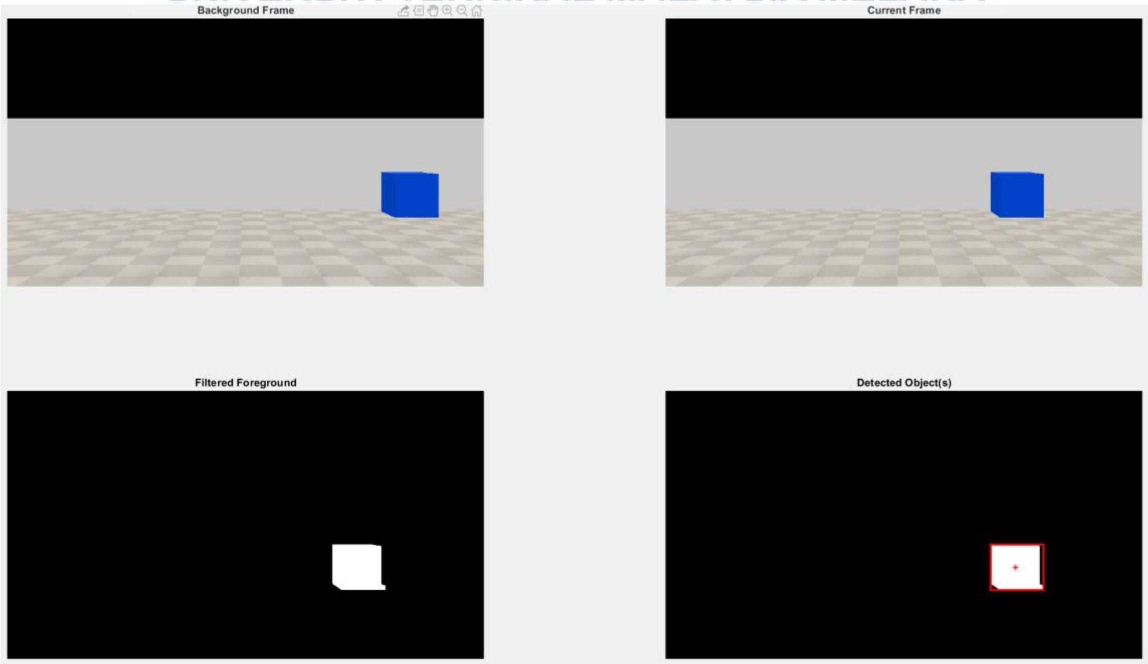

Figure 4.8 Ouput result when velocity = 0.1m/s.

Figure 4.9 Ouput result when velocity = 0.9m/s.

Nine different values were tested and the results on the effect of object velocity is shown in Figure 4.4. Error values are measured by comparing the location obtained in MATLAB to the real-world location of the object in V-REP. To get the location in MATLAB, conversion from pixels to unit length in vertical and horizontal directions is accomplished by multiplying the number of pixels with the calibration factor as described in (3.3) in that direction.

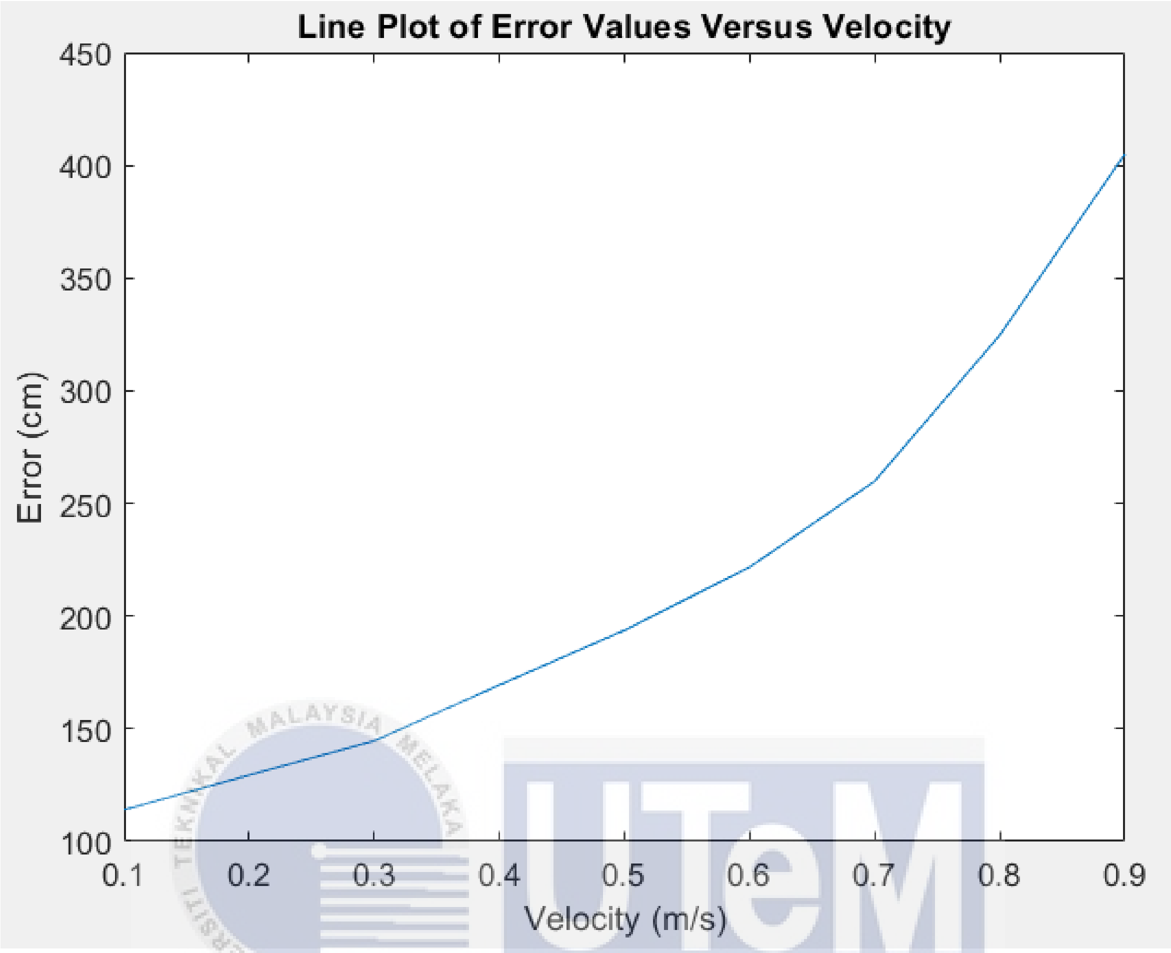

Figure 4.10 Results of object velocity against error.

From these results, it is evident that values of error will become larger as velocity of the object gets faster. However, the program is still able to detect objects and extract features correctly. This indicates that it can detect and track fast motions effectively with object velocity having less effect on the accuracy of the proposed work.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In this paper, the focus is on the problem of the development of anomalous activity detection framework. The proposed work includes segmentation of foreground, detection of objects, extraction of features and recognition of predominant behavior. The segmentation of foreground and detection of moving object achieved by frame differencing and Mathematical Morphology. Blob analysis is done to extract different features like centroid and bounding box for each object. The connected components of detected foreground objects such as object size in pixels are compared to differentiate between objects of interest. Detection of anomaly is achieved by applying if-then rule-based approach which contributes to determining dominant and less dominant behavior. The approach is proved reliable and tested based on localization error by tuning object velocity while the accuracy of the model is based on rate of false positive and false negative occurrences by the program.

As a conclusion, the development of anomaly detection framework using vision system is completed in this project. From the results, it is evident that the approach proved reliable to handle one or more stationary and moving objects on a stationary background. Besides that, the system is capable of detecting fast motions without much localization error. However, the proposed framework is proved sensitive to pixel intensity and illumination changes and the synchronization between MATLAB and V-REP are slow.

5.2 Future Works

Future improvisation may include the implementation of adaptive thresholding for a more accurate adaptive background modeling. On top of that, the proposed work can be improved for the application on dynamic backgrounds and live video streams. Another consideration is the adaptation of machine learning for the detection and tracking of multiple overlapping objects and abnormal behavior analysis.

REFERENCES

[1] M. A. F. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "A review of novelty detection," *Signal Processing*, vol. 99, pp. 215–249, 2014.

[2] B. J. B. Angela A. Sodemann, Matthew P. Ross, "Meteorological report from Jerusalem for year 1885," *A Rev. Anom. Detect. Autom. Surveill.*, vol. 26, no. 2, pp. 144–146, 2012.

[3] A. Hampapur *et al.*, "Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking," *IEEE Signal Process. Mag.*, vol. 22, no. 2, pp. 38–51, 2005.

[4] C. S. Regazzoni, A. Cavallaro, Y. Wu, J. Konrad, and A. Hampapur, "Video analytics for surveillance: Theory and practice," *IEEE Signal Process. Mag.*, vol. 27, no. 5, pp. 16–17, 2010.

[5] "ATMIA's Market Research Library." .

[6] R. P. Loce, R. Bala, and M. Trivedi, "Computer Vision and Imaging in Intelligent Transportation Systems," 2017.

[7] S. Bremer, R. Fachgebiet, N. Systemtheorie, T. Universit, T. Martin, and B. Prof, "Anomaly Detection with event-based Vision : Detection of broken Cogwheels in a running System," 2013.

[8] R. R. Sillito and R. B. Fisher, "Semi-supervised Learning for Anomalous Trajectory Detection," *Procedings Br. Mach. Vis. Conf. 2008*, no. January 2008, pp. 103.1-103.10, 2008.

[9] D. F. Pava, "Object Detection In Low Resolution Video Sequences," no. April, p. ProQuest, 2009.

[10] A. Bera, S. Kim, and D. Manocha, "Realtime Anomaly Detection Using Trajectory-Level Crowd Behavior Learning," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, pp. 1289–1296, 2016.

[11] E. Varghese, J. Mulerikkal, and A. Mathew, "Video Anomaly Detection in Confined Areas," *Procedia Comput. Sci.*, vol. 115, pp. 448–459, 2017.

[12] Hua Zhong, Jianbo Shi, and M. Visontai, "Detecting unusual activity in video," *Proc. 2004 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, 2004. CVPR 2004.*, vol. 2, pp. 819–826.

[13] S.-R. Ke, H. Thuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi, *A Review on*

Video-Based Human Activity Recognition, vol. 2, no. 2. 2013.

[14] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, “P finder: real-time tracking of the human body,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, 1997.

[15] M. Seki, H. Fujiwara, and K. Sumi, “A robust background subtraction method for changing background,” *Proc. IEEE Work. Appl. Comput. Vis.*, vol. 2000-Janua, pp. 207–213, 2000.

[16] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, “Detecting moving objects, ghosts, and shadows in video streams,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1337–1342, 2003.

[17] D. J. Shah, D. Estrin, S. Ahmadian, and M. Rahimi, “Motion Based Bird Sensing Using Frame Differencing and Gaussian Mixture,” 2007.

[18] C. M. Sabliov, D. Boldor, K. M. Keener, and B. E. Farkas, “Image processing method to determine surface area and volume of axi-symmetric agricultural products processing method to determine surface area and volume of axi-symmetric,” vol. 2912, 2007.

[19] H. Permuter, J. Francos, and I. Jermyn, “A study of Gaussian mixture models of color and texture features for image classification and segmentation,” *Pattern Recognit.*, vol. 39, no. 4, pp. 695–706, 2006.

[20] K. M. Ozonat and R. M. Gray, “Gauss mixture image classification for the linear image transforms,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, vol. V, no. April 2005, 2005.

[21] T. Yu, C. Zhang, M. Cohen, Y. Rui, and Y. Wu, “Monocular video foreground/background segmentation by tracking spatial-color Gaussian mixture models,” *2007 IEEE Work. Motion Video Comput. WMVC 2007*, no. March, 2007.

[22] T. Horprasert, D. Harwood, and L. S. Davis, “A Robust Background Subtraction and Shadow Detection,” *4th Asian Conf. Comput. Vis.*, no. May, pp. 983–988, 2000.

[23] T. Horprasert, D. Harwood, and L. S. Davis, “A statistical approach for real-time robust background subtraction and shadow detection,” *Ieee Iccv*, vol. 99, no. May 2014, pp. 1–19, 1999.

[24] R. Augusto and C. Jimenez, “Event detection in surveillance video,” no. May, 2010.

APPENDICES

APPENDIX A GANTT CHART

Project Activities of FYP 2	Week													
	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
Meeting with supervisor														
Further improvise data														
Recording raw trials data														
Preparing raw trials data														
Data collection														
Data analysis														
Final report preparation														
Presentation of FYP 2														
Submission of report														

APPENDIX B MATLAB CODING

```
clc; % Clear command window.
clearvars; % Get rid of variables from prior run of this m-file.
fprintf('Running testing.m...\n'); % Message sent to command window.
disp('Program started');
vrep=remApi('remoteApi');
vrep.simxFinish(-1);
clientID=vrep.simxStart('127.0.0.1',19997,true,true,5000,5);

if (clientID>-1)
           disp('Connected to remote API server');
    pause(0.1);
    vrep.simxSynchronous(clientID,true);
    vrep.simxStartSimulation(clientID,vrep.simx_opmode_blocking);

           vrep.simxFinish(-1);

    else
    disp('Failed connecting to remote API server');
end

    clientID=vrep.simxStart('127.0.0.1',19997,true,true,5000,5);

    if (clientID>-1)

[err,camhandle]=vrep.simxGetObjectHandle(clientID,'Vision_sensor',vrep.simx_opmode_oneshot_wa
it);

[errorCode,resolution,img]=vrep.simxGetVisionSensorImage2(clientID,camhandle,0,vrep.simx_opmo
de_oneshot_wait);
            subplot(2,2,1), imshow(img); title('Background Frame');

            t0 = clock;
            while etime(clock, t0) < 30
            pause(1);

[errorCode2,resolution,img2]=vrep.simxGetVisionSensorImage2(clientID,camhandle,0,vrep.simx_op
mode_oneshot_wait);
            % Image difference
            difference_image = double(img) - double(img2);
            subplot(2,2,2), imshow(img2); title('Current Frame');

            % Detect Objects in an Initial Video Frame
            grayImage = rgb2gray(difference_image)
            se = strel('square', 3);
            filteredForeground = imopen(grayImage, se);
            subplot(2,2,3), imshow(filteredForeground); title('Filtered Foreground');

            % find bounding boxes of each connected component
            labeledImage = bwlabel(filteredForeground);

            info = regionprops(labeledImage, 'centroid');;

                for k = 1 : length(info)
                    centroids = cat(1, info.Centroid);
                    subplot(2,2,4), imshow(filteredForeground);
                    hold(imgca,'on');
```

```
                    plot(imgca,centroids(:,1), centroids(:,2), 'r*'); title('Detected
Object(s)');
                    hold(imgca,'off');
                end

            info = regionprops(labeledImage,'Boundingbox') ;

            % BB will be having four values, first two points corresponds to (x,y)
coordinates of left bottom corner of the rectangle,
            % third is width and fourth is height of the rectangle.
            % The code is already automates, it will count the number of boxes and plot.
            hold on
                for k = 1 : length(info)
                    BB = info(k).BoundingBox;
                    rectangle('Position',
[BB(1),BB(2),BB(3),BB(4)],'EdgeColor','r','LineWidth',2) ;

                    beep; msgbox('ANOMALY DETECTED!')
                end

            end
                vrep.simxStopSimulation(clientID,vrep.simx_opmode_blocking);
        end
    vrep.delete(); % call the destructor!

    disp('Program ended');
```

APPENDIX C V-REP CODING

```
simRemoteApi.start(19999,1300,false,true)

path=sim.getObjectHandle('Path')
box=sim.getObjectHandle('Cuboid') -- make sure the cuboid is static!
pathLength=sim.getPathLength(path)
posOnPath=0 -- varies between 0 (start) and pathLength (end)
v=0.3 -- velocity of the movement on the path (1cm/s)

deltaTime=0
deltaTimeLeft=sim.wait(deltaTime,simulationTime)

while true do

        l=posOnPath/pathLength
        if (l>pathLength) then
            l=pathLength
        end

        position=sim.getPositionOnPath(path,l)
        orientation=sim.getOrientationOnPath(path,l)
        position[3]=sim.getObjectPosition(box,-1)[3]
        sim.setObjectPosition(box,-1,position)
        sim.setObjectOrientation(box,-1,orientation)

        posOnPath=posOnPath+v*sim.getSimulationTimeStep()

        sim.switchThread() -- do not waste time in the same simulation step!

end
```