

**FIGHTING VIDEO ANALYSIS EMPLOYING COMPUTER VISION
TECHNIQUE**

**BY
FOO WEN SHUN**

**A REPORT
SUBMITTED TO
Universiti Tunku Abdul Rahman
in partial fulfillment of the requirements
for the degree of
BACHELOR OF COMPUTER SCIENCE (HONOURS)
Faculty of Information and Communication Technology
(Kampar Campus)**

JAN 2022

REPORT STATUS DECLARATION FORM

Title: FIGHTING VIDEO ANALYSIS EMPLOYING COMPUTER VISION
TECHNIQUE

Academic Session: JAN 2022

I FOO WEN SHUN

declare that I allow this Final Year Project Report to be kept in

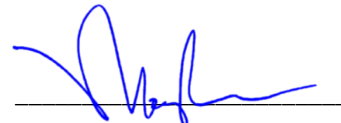
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,



(Author's signature)



(Supervisor's signature)

Address:

37, LEBUH RAPAT BARU 9,

TAMAN SONG CHOON, 31350,

IPOH, PERAK

Leung Kar Hang

Supervisor's name

Date: 20/4/2022

Date: 29 Apr 2022

Universiti Tunku Abdul Rahman			
Form Title : Sample of Submission Sheet for FYP/Dissertation/Thesis			
Form Number: FM-IAD-004	Rev No.: 0	Effective Date: 21 JUNE 2011	Page No.: 1 of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 20/4/2022

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that Foo Wen Shun (ID No: 18ACB03066) has completed this final year project entitled “ Fighting video analysis employing computer vision technique ” under the supervision of Prof. Leung Kar Hang (Supervisor) from the Department of Computer Science, Faculty of Information and Communication Technology.

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,



(*Foo Wen Shun*)

DECLARATION OF ORIGINALITY

I declare that this report entitled “**Fighting video analysis employing computer vision technique**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.



Signature : _____

Name : FOO WEN SHUN

Date : 20/04/2022

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to my supervisor, Prof. Leung Kar Hang, for providing me with the wonderful opportunity to work on an image processing project. Prof. Leung have given a lot of guidance for me to complete this project. Whenever I faced any problem, Prof. Leung always provide advice and assistance for me to overcome it. Once again, a million thanks to Prof. Leung.

Finally, I want to express my gratitude to my friends and family for their unwavering support and encouragement throughout the course.

ABSTRACT

This project is about analysing and classifying the fighting video from the UCF_Crimes dataset to explore solutions to detect fighting events employing computer vision technique. It is expected that different scenes need different approaches to solve the problems and innovative solution for each category should be implemented and tested. Anomaly detection is one of the most challenging tasks in computer vision due to ambiguous nature of the anomaly and the complex nature of human behaviours. Anomalous events rarely happen as compared to normal events which leads to the waste of labour and time. Motivation of this project is to detect a few categories of fighting events to timely signal such incidences as a warning and its innovation is to adapt the automatic anomaly detection and eliminate the use of manual anomaly detection. The field of study for this project includes computer vision, image processing, machine learning and deep learning. For the methodology of the project, the input video frames will first be split into training and testing data and undergo pre-processing steps such as conversion to grayscale to reduce noise and dilation process to increase the white region. Then, the important features between two consecutive frames of input videos are extracted using optical flow. The optical flow of the important features was calculated, and the tracks are drawn out using random colour lines. Next is the observation process to prove that the optical flow generated was meaningful and it was suitable for the project solution. The observation processes included are using YOLO to compare the human size body, recolour the optical flow based on the orientation value, draw the Delaunay triangle and Voronoi diagram, and generate the frequency histogram for the orientation value of optical flow. After observation, the standard deviations of orientation of optical flows on all the dataset videos was recorded and normalized. The normalized data was used to fit and train the SVM classification model. Last step is to perform classification to detect the fighting events on dataset videos. The trained model was also evaluated using confusion matrix, classification report, AUC-ROC curve, and the learning curve.

TABLE OF CONTENTS

TITLE PAGE	i
REPORT STATUS DECLARATION FORM	ii
FYP THESIS SUBMISSION FORM	iii
DECLARATION OF ORIGINALITY	iv
ACKNOWLEDGEMENTS	v
ABSTRACT	vi
TABLE OF CONTENTS	vii
LIST OF FIGURES	x
LIST OF TABLES	xii
LIST OF ABBREVIATIONS	xiii
CHAPTER 1 INTRODUCTION	1
1.1 Problem Statement and Motivation	1
1.2 Project Scope	1
1.3 Project Objectives	2
1.4 Impact, significance, and contribution	2
1.5 Background Information	3
CHAPTER 2 LITERATURE REVIEW	6
2.1 Non-Deep Learning Approaches	7
2.1.1 Motion-aware Feature with Temporal Augmented Network	7
2.1.2 MoSIFT Feature and Sparse Coding	8
2.2 Conventional Methods	10
2.2.1 STIP Spatio-temporal Descriptors	10
2.2.2 Histogram-of-Oriented-Rectangles (HOR)	11
2.3 Deep Learning Approaches	13
2.3.1 Fully Convolutional Neural Network (FCN)	13
2.3.2 Social-LSTM	14
	16

2.3.3 Survey of Vision-based Methods for Action Recognition	18
2.3.4 Survey of Deep Learning Based Approaches for Video Anomaly Detection	18
2.3.5 Overview of Research Methodologies in Deep Learning Based Anomaly Detection	20
2.3.6 Survey on Deep Learning Techniques for Video Anomaly Detection	21
2.3.7 Optical Flow on Violence Detection	22
2.4 Critical Remarks of previous works	
CHAPTER 3 PROPOSED METHOD/APPROACH	26
3.1 Design Specifications	26
3.1.1 Methodologies and General Work Procedure	26
3.1.2 Tools to use	28
3.1.3 System Performance Definition	28
3.1.4 Verification Plan	29
3.2 System Design/Overview	30
3.2.1 Flowchart	30
3.2.2 Descriptions of System Design	31
3.3 Implementation Issues and Challenges	32
CHAPTER 4 PROJECT IMPLEMENTATION	34
4.1 Dataset	34
4.2 Implementation Details	34
4.2.1 Pre-Processing	34

4.2.2	Optical Flow	35
4.2.3	YOLOv3	37
4.2.4	Observation using Orientation of Optical Flow	37
4.2.5	Delaunay Triangulation & Voronoi Diagram	39
4.2.6	Generate Frequency Histogram using Orientation of Optical Flow	40
4.3	Model Training	41
CHAPTER 5 EXPERIMENTAL RESULT		42
5.1	System Performance	42
5.2	Comparison of System Performance	42
5.3	Error Analysis & Future Work	46
5.4	Contributions	48
CHAPTER 6 CONCLUSION		50
6.1	Project Review, Discussions and Conclusions	50
6.2	Novelties	51
6.3	Contributions	52
6.4	Future Work	52
BIBLIOGRAPHY		54
WEEKLY LOG		56
POSTER		65
PLAGIARISM CHECK RESULT		66
FYP2 CHECKLIST		68

LIST OF FIGURES

Figure Number	Title	Page
Figure 2.1.1	Overall framework of motion-aware feature	7
Figure 2.1.2	Framework of MoSIFT feature and sparse coding approach	9
Figure 2.2.2	Process of Histogram-of-Oriented-Rectangles (HOR)	11
Figure 2.3.1	Schematic sketch of FCNN detection method	13
Figure 2.3.2	Overview of Social-LSTM method	15
Figure 2.3.3	A general process flow for generic action recognition system	17
Figure 2.3.5.1	Deep learning-based algorithms Versus traditional algorithms performance comparison	19
Figure 2.3.5.2	Deep learning-based anomaly detection algorithms successful applications	19
Figure 2.3.7	General Process of Violence Detection	21
Figure 3.1.1	Flowchart of General Work Procedure	25
Figure 3.2.1	Flowchart of System Design	29
Figure 4.2.1	Pre-Processing Steps	34
Figure 4.2.2.1	Optical Flow of Fighting Video	35
Figure 4.2.2.2	Running Information of Optical Flow	36
Figure 4.2.2.3	Optical Flow of Non-Fighting Video	36
Figure 4.2.3.1	YOLO3 of Fighting Video	37
Figure 4.2.4.1	Lookup Table for HSV to RGB conversion	38
Figure 4.2.4.2	Coloured Optical Flow of Normal Video	38
Figure 4.2.4.3	Coloured Optical Flow of Fighting Video (1)	38
Figure 4.2.4.4	Coloured Optical Flow of Fighting Video (2)	39
Figure 4.2.5.1	Delaunay Triangulation & Voronoi Diagram for Fighting Video	40
Figure 4.2.5.2	Delaunay Triangulation & Voronoi Diagram for Normal Video	40

Figure 4.2.6.1	Frequency Histogram for Fighting Video	41
Figure 4.2.6.2	Frequency Histogram for Normal Video	41
Figure 5.1.1	Training and Classification Process on Visual Studio 2019 using C++	42
Figure 5.1.2	Training and Classification Process on Jupyter Notebook using Python	42
Figure 5.1.3	Confusion Matrix Visualization	43
Figure 5.1.4	AUC-ROC Curve	45
Figure 5.1.5	Learning Curve	45
Figure 5.3.1	Fighting Events which Humans Involved are Far Away	47
Figure 5.3.2	Fighting Events which Obstacles Blocked the Sight	47

LIST OF TABLES

Table Number	Title	Page
Table 2.4	Comparison for different approaches	24
Table 3.1.2	Hardware Specification	27
Table 3.1.4.1	Verification Plan P1	28
Table 5.1	Classification Metrics Report	43

LIST OF ABBREVIATIONS

<i>AI</i>	Artificial Intelligence
<i>YOLO</i>	You Only Look Once
<i>CNN</i>	Convolutional Neural Network
<i>FCN</i>	Fully Convolutional Neural Network
<i>DAD</i>	Deep Anomaly Detection
<i>HOR</i>	Histogram-of-Oriented-Rectangles
<i>SVM</i>	Support Vector Machines
<i>LSTM</i>	Long Short-Term Memory
<i>MIL</i>	Multiple Instance Learning
<i>BoW</i>	Bag-of-Words
<i>KDE</i>	Kernel Density Estimation
<i>STIP</i>	Space-Time Interest Points
<i>HOG</i>	Histograms of Oriented Gradients
<i>HOF</i>	Histograms of Optical Flow
<i>ANN</i>	Artificial Neural Networks

CHAPTER 1

Introduction

In this chapter, we present the problem statement and motivation of the project, the project's scope and objectives, and the contributions to the field. The background information about the fields related to the project are also described at the last section in this chapter.

1.1 Problem Statement and Motivation

Surveillance videos can catch a wide range of realistic anomalous activities. Anomalous activities such as fighting, vandalism and other unusual activities should be identified automatically and in a timely manner. However, automatic anomaly detection is difficult due unclear nature of anomaly. Furthermore, given realistic circumstances, the same behaviour could be normal or abnormal depending on the circumstances. [1] Detecting anomaly events such as fighting events is a very important task in video surveillance. Anomaly event occurs infrequently compared to the normal activities which resulted in a waste of both labour and time. Thus, there is an acute need for such a software to detect a few categories of fighting events to timely signal such incidences as a warning.

1.2 Project Scope

A program that detects a few categories of fighting events to give early warning of such incidences will be delivered at the end of the project. The project title which is fighting video analysis employing computer vision technique is specific and clear since it is expected that different scenes require different approach to solve the problems. The different fighting scenes need to be group into different categories and innovative solution by employing computer vision technique should be implemented and tested for each category. Anomaly detection is a type of coarse video understanding that separates anomalies from common patterns. [1] Classification techniques can be used to categorise it into one of the specialised activities once an anomaly is identified.

1.3 Project Objectives

The project is about analysing fighting video employing computer vision technique. The fighting scenes from UCF_Crimes dataset will be used to analyse to explore solutions that detect fighting events. The objective of this project is to develop a software that detect a few categories of fighting events to give early warning. The software should send alerts when a fighting event happened and specify the time window in which the anomaly event occurs. This project focuses solely on analysing different fighting scenes to develop a solution and other anomalous activities such as vandalism and burglary will not be covered in this project.

The main objective of the project is to detect a few categories of fighting events and provide early warning of such incidences. This project's main objective can be broken down into several sub-goals. One of the sub-objectives of this project are to recognize the motion of people in different type of fighting video to detect different categories of fighting events. Another sub-objective of this project is to achieve real time processing in object detection in a high speed. Other than that, the solution also needs to achieve high accuracy while being able to run in real-time.

1.4 Impact, significance, and contribution

The innovation of this project is to adapt the automatic anomaly detection and eliminate the use of manual anomaly detection. Manual anomaly detection should be eliminated since it has many issues. One of them is there is an obvious shortcoming in the use of surveillance cameras, as well as an unworkable camera-to-human monitor ratio. [1] A manual anomaly detection system also necessitates the use of human operators and is labour-intensive, making it prone to mistakes and exhaustion.

Anomaly event detection in surveillance videos is a hot area in computer vision research. It's also been employed in a variety of security settings which includes fighting scenarios. The contribution of this project is developing intelligent computer vision algorithms for the detection of video anomalies automatically to alleviate the waste of labour and time and improve the community safety in our country.

1.5 Background information

Video surveillance systems have progressed with their automation grade since the 1960s, and three generations can be distinguished. From 1960 to 1980, the first generation consisted of analogue CCTVs with a modest level of automation. Digital monitor and computer vision processing were used in the second generation, which lasted from 1980 to 2000. The third generation since the year 2000 have uses semi-automatic video surveillance systems. Third generation's video surveillance systems have achieved a level of automation that allows them to recognise certain abnormal human behaviours and provide alerts. To establish the technique, these systems follow a common pattern which consists of multiple sequential processes, the most important of which are the detection, tracking, and behavioural analysis of the objects. [2] Despite all the advancements, a higher level of automation is hampered by several high-level concerns, such as low security system cooperation, highly dependent on intense human concentration to discover anomalies, etc. Anomaly detection in low-quality videos, cluttered background between the objects, and others are the low-level concerns that aren't properly addressed. [2]

One of the fields of this project is computer vision. Computer vision is one of the most powerful types of Artificial Intelligence (AI) that practically everyone has encounter without even realising it in daily lives. Computer vision makes computers understand images in the same ways that human do by mimicking the human vision system. It is a multidisciplinary area that can be categorised as a subfield of AI and machine learning.

To assess the properties of digital images and videos, computer vision collects and analyses data from it. Image identification and analysis are the computer vision's operations that can assist computers to interpret any digital images. The data gathered throughout the operation is then translated into a computer-readable format to help in decision-making. Computer vision's goal is to understand the content of digital images.

Image processing is a technique of transforming an image to a digital representation and then executing operations on it to improve it or extract important information. Image processing takes in an image as input and return the image-related features or parameter as output. Image processing is related to computer vision since some computer vision system's raw input require image processing technique for

enhancement or feature extraction. Image processing is used for image enhancement and restoration, image retrieval, etc.

Surveillance is defined as the monitoring of people's behaviour, activities to direct and protect them. Surveillance cameras are frequently being used in public locations such as roads, banks, and shopping malls to improve public safety. Detecting abnormal events such as fighting, vandalism, or other criminal behaviours is one of the most critical tasks in video surveillance. [1]

One of the most difficult challenges in computer vision is anomaly detection. Several ways for anomaly detection have been proposed throughout the years, ranging from statistical-based approaches to machine learning-based approaches. One of the methodologies utilised in the anomaly detection area is deep learning.

Deep learning is a subset of machine learning that attempts to mimic the human brain's ability for decision making. Artificial Neural Networks (ANNs), a computational architecture motivated by the operation of the human brain, are at the core of deep learning. A neural network is made up of many computational cells that individually do a simple task and communicate with one another to reach a decision. [3]

Traditional machine learning algorithms makes computers to learn from their prior experiences utilising 3 different learning approaches which are supervised, unsupervised and semi-supervised learning and these necessitate feature extraction which requires the assistance of a domain expert. Choosing the appropriate features for a specific situation is a difficult task. [4] However, deep learning solves the problem of feature extraction by automatically extracting relevant characteristics from raw input rather than requiring pre-selected characteristics. A deep learning model is made up of numerous processing layers that may learn multiple data features at different degrees of abstraction. The network can learn different features at different levels.

Deep learning is utilised in digital image processing to handle difficult challenges like image classification. Deep learning approaches such as Convolutional Neural Network (CNN) have pushed the boundaries of what is achievable by improving prediction performance using huge data and extensive computational resources. [3] Deep learning can achieve a level of precision that matches or perhaps surpasses that of humans.

CHAPTER 1 INTRODUCTION

Rapid advances in deep learning, as well as advancements in device capabilities including processing power, graphic card, storage capacity have boosted the performance of vision-based applications. [3] Deep learning has recognised as a promising method for accomplish impressive result in a variety of applications, including image and face recognition, diagnostic imaging in healthcare, autonomous driving, etc. [4]

CHAPTER 2

Literature Reviews

In this chapter, we reviewed papers for different approaches on anomaly detection. For video anomaly detection, there are several attempts to detect the anomalies using different approaches which can be divided to deep learning approaches, non-deep learning approaches and conventional method. Deep learning approaches organise the algorithms in layers to build an ANN that can learn and detect the anomalies on its own. Non-deep learning approaches use algorithms for interpreting data and learn from the data to detect the anomalies based on what it has learnt such as the machine learning methods. While conventional methods refer to the traditional and old way of recognizing the human action using the spatio-temporal descriptors and novel pose descriptor. These conventional methods provided simple and straightforward approaches for the video anomaly detection. For deep learning approaches, Sabokrou et.al. [5] proposed a Fully Convolutional Neural Network (FCN) to overcome the problem that Convolutional Neural Network (CNN) faced. Alahi et.al. [6] also proposed a deep learning method of Social-LSTM to predict human trajectory in the crowded spaces which was helpful to detect anomalies. Weinland et al. [7] conducted a survey of vision-based methods for the representation, segmentation, and recognition of actions. Kiran et al. [8] reviewed the state-of-the-art deep learning-based approaches for video anomaly detection and classified them according to the type of model and criteria of detection. Chalapathy and Chawla [9] presented an overview of research methodologies in deep learning-based anomaly detection, as well as an assessment of their effectiveness in diverse application areas. For non-deep learning approaches, Zhu & Newsam [10] proposed a temporal augmented network to learn a motion-aware feature for detecting anomalies in video. Xu et.al. [11] proposed a method that employs Motion SIFT algorithm and sparse coding to detect violence in the videos. Biswas and others [12] have conducted research studies that review optical flow methods for detecting violence in surveillance videos. Conventional methods like STIP proposed by Bermejo Nievas et.al [13] to detect violence in the videos. Another conventional method which is Histogram-of-Oriented-Rectangles (HOR) proposed by Ikizler and Duygulu [14] to recognize human actions.

2.1 Non-Deep Learning Approaches

2.1.1 Motion-aware Feature with Temporal Augmented Network

In 2020, Zhu & Newsam [10] proposed a temporal augmented network to learn a motion-aware feature instead of directly using optical flow for effective anomaly detection performance in video. They also used an attention block to include temporal context into the Multiple Instance Learning (MIL) ranking algorithm. The learned attention weights can aid in better distinguishing between abnormal and normal video segments. They claimed that their entire framework operates faster than real-time, allowing it to be directly applied to real-world issues. Figure 2.1.1 below is their overall framework.

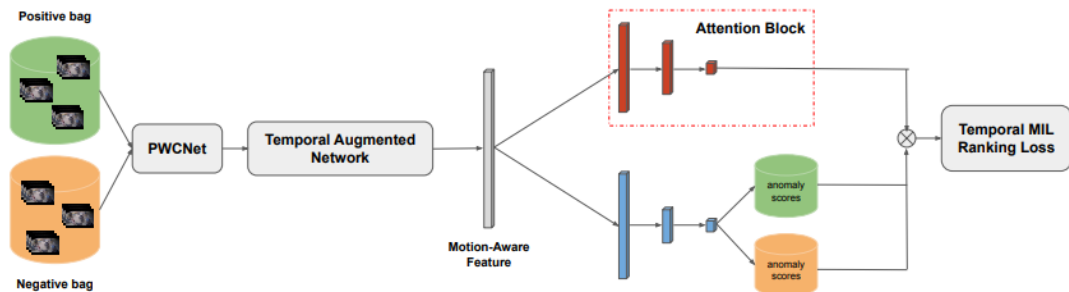


Figure 2.1.1: Overall framework of motion-aware feature [10]

For the input, a positive video with anomalies is referred as positive bag, while a regular video with no anomalies is referred as negative bag. It was assumed that at least one of the temporal segment instances has a positive bag anomaly, but none of the instances has a negative bag anomaly. The network they proposed is an autoencoder and they used optical flow as the autoencoder's input since it is the most extensively used motion representation which push the network to learn the complex motion pattern directly. To be specific, they calculate the optical flow between the adjacent frames using the network-based flow estimator PWCNet. Then, they encoded a compact representation that can be used to restore the input as accurately as possible to spot video anomalies.

For MIL Ranking Model, anomaly detection is formulated as an anomaly score regression issue, but it has some limitations to the ranking loss, so they introduced their temporal MIL Ranking Model by leveraging temporal context information and used an attention-based approach to collect the total anomaly score of the video. The total anomaly score of an anomaly video should be higher than normal video so they

compute the anomaly score video-wise rather than segment-wise. Within the network, the attention weights are learned from start to finish and the attention block has three fully connected layers. Finally, they used the sparsity constraints since anomalies are infrequent and the final loss function was made for their proposed method.

The dataset they used are the real-world surveillance videos from UCF Crime. Their motion-aware feature learned from the temporal augmented network and temporal MIL Ranking Model has achieved an impressive performance since it has a high anomaly detection AUC score and runs in a faster speed compared to the previous approach. However, their model still struggles in some well-known difficult conditions, such as fast motion, people clustering, etc.

2.1.2 MoSIFT Feature and Sparse Coding

Xu and others [11] proposed a method that employs Motion SIFT (MoSIFT) algorithm and sparse coding for violence detection. They stated that applying local spatio-temporal description to the query video was a frequent video description technique. The low-level description is then summed onto the high-level characteristic using the Bag-of-Words (BoW) paradigm. Traditional spatiotemporal descriptors such as HOG and HOF, on the other hand, were inadequately discriminative, and each feature vector in the BoW model was assigned to just one visual word, resulting in quantisation error. So, they proposed their method based on MoSIFT feature and sparse coding to solve this problem.

The MoSIFT method was implemented to extract a query video's low-level description by recognising distinctive local features based on their look and motion. To remove feature noise, the Kernel Density Estimation (KDE) method was utilised to choose features on the MoSIFT descriptor. The sparse coding-based system has been effectively employed for image and action categorization tasks, according to the researchers. The effective feature video feature is built using the sparse coding technique, which subsequently processes the selected MoSIFTs. According to the researchers, it produces a substantially smaller reconstruction error and a much more

discriminative video representation than the BoW model. The Figure 2.1.2 below is the framework of their proposed approach.

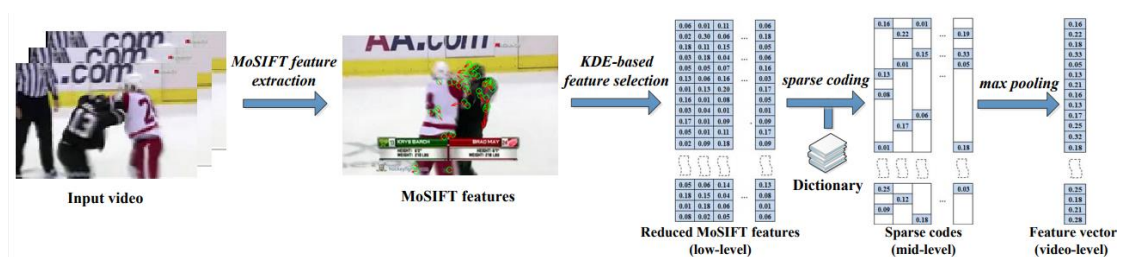


Figure 2.1.2: Framework of MoSIFT feature and sparse coding approach [11]

They started by detecting the interest locations in the input video and using the MoSIFT technique to extract the final MoSIFT feature, which is a multi-dimensional vector. The aggregated histogram of optical flow makes up half of the 256-dimensional vector, while the typical SIFT feature makes up the other half. They chose the most significant features from the original multi-dimensional MoSIFT descriptor using a feature selection method based on Kernel Density Estimation (KDE) because the multi-dimensional MoSIFT may contain some unnecessary features. The decreased low-level descriptors are converted into compacted mid-level features using sparse coding, resulting in a more discriminative description of human behaviour. The max pooling technique is applied to the query video's sparse code set to build an effective representation of the entire video to convert the mid-level features to video-level features. Finally, these feature vectors are used to train an SVM classifier.

The datasets they used for their experiment are hockey fight and crowd violence dataset. The BoW model is used in conjunction with HOG, HOF, and MoSIFT to provide the results on the datasets. They claimed that by combining the MoSIFT technique with the sparse coding framework, it outperforms state-of-the-art solutions in both datasets. Since their method has good precision and minimal quantization error, it demonstrates that their suggested video feature extraction strategy is effective. However, they only used two datasets in their studies, therefore the efficiency of their proposed strategy in additional datasets is uncertain and must be investigated further.

2.2 Conventional Methods

2.2.1 STIP Spatio-temporal Descriptors

In 2011, Bermejo Nieves et al. [13] evaluated the effectiveness of the action recognition methods which is known as STIP in the violence detection issue. Local image features, also known as interest points, are compact and abstract representations of visual patterns. Similarly, local spatio-temporal information can be used to create compact and descriptive motion representations. The spatio-temporal descriptors that they used is Space-Time Interest Points (STIP).

The detected interest sites in STIP exhibit a large amount of intensity fluctuation in space as well as non-constant movement in time. On a variety of spatial and temporal scales, these important locations can be located. The features vectors in the region of the detected important points are then extracted using HOG, HOF and a combination of HOG and HOF termed HNF. These features vector is resistant to changes in pattern and velocity, and they are utilised to accurately detect motion events. They then utilised the Bag-of-Words (BoW) technique, which involves representing every video frame as a histogram over a set of visual terms, resulting in a fixed-dimensional encoding that can be analysed by a classifier. Low-level STIP descriptors derived from k-means clustering of a set of data are frequently utilised to create the dictionary of visual words during the learning phase. The next stage, given a vocabulary, is to assign a numerical value to each descriptor extracted from a video and compare it to the nearest visual word, resulting in word occurrence histograms. The classification of the histograms is the final stage in this BoW technique. These histograms can be classified with a typical classifier like SVM.

They employed two datasets: a big video collection that consists of hockey games and a short video collection that consists of action movie clips. Using the same datasets, they compared the STIP and MoSIFT approaches. The MoSIFT outperforms the STIP in all cases, according to the results. The MoSIFT performs well, with the highest accuracy of 89.5 percent, whereas the STIP's highest accuracy was just 59.0 percent in all cases with varying amounts of language. Although the STIP approach was computationally less expensive than MoSIFT, the experiment findings reveal that the STIP strategy is unpromising due to its low precision. However, despite being more

computationally expensive than STIP, the MoSIFT representation demonstrates promise and successful performance for video violence identification.

2.2.2 Histogram-of-Oriented-Rectangles (HOR)

In 2009, Ikizler and Duygulu [14] proposed that the human actions can be easily represented by posture using HOR for representing and recognising human actions. The three fundamental factors that characterise an action are the body's stance, the speed of its motion, and the relative ordering of its positions. The relative importance of these factors is determined by the type of the acknowledged acts. The "bend" and "walk" actions, for example, can be distinguished by the body's posture. The pace of the body motion can distinguish the "jog" and "run" activities while the position itself may not be sufficient due to the similarities of both actions. The relative ordering of the poses distinguishes the "stand up" and "sit down" movements, which both include the same postures in reverse temporal orders. The human body is represented as a series of rectangular patches in their method.

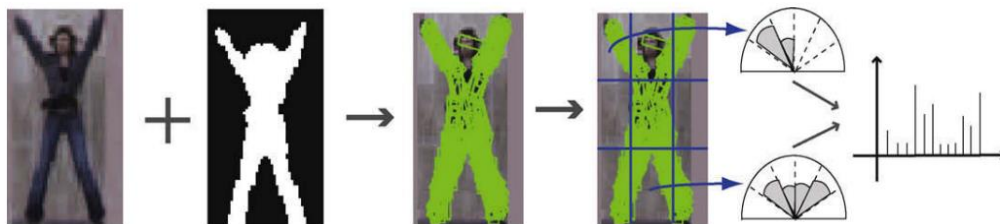


Figure 2.2.2: Process of Histogram-of-Oriented-Rectangles(HOR) [14]

Figure 2.2.2 above is the feature extraction stage of their approach. First, background subtraction is used to isolate the human figure in each picture. They looked for rectangular regions that could be candidates for limbs using these silhouettes. The silhouette's bounding box was then divided into an equal-sized grid. They proposed HOR as a simple pose descriptor after discovering the rectangular sections of the human body to determine the stance. Based on the orientations of the collected rectangular patches, they produced histograms. They combined the histograms from each subregion to create their feature vector.

When it comes to action recognition, there are times when looking at single poses is insufficient to distinguish between two actions. In many cases, a shape-based action description is insufficient, and temporal dynamics must be considered. HORs could be computed over segments of frames instead of single frames to incorporate temporal

components. They defined HOR over a window of frames, which makes it easier to distinguish activities that are identical in position but differ in pace.

After calculating stance descriptors for each frame, they conducted supervised action classification. They evaluated the performance of their posture descriptor in action classification situations in four distinct ways. The first method was nearest neighbour classification, which identified activities by matching each frame's characteristics. The second technique was to use global histogramming to integrate all of the spatial histograms of oriented rectangles in the series. A classifier-based technique employing Support Vector Machines was the third strategy (SVM). The fourth method was to apply Dynamic Time Warping to the HOR descriptor's spatial representation. In cases where the pose descriptor isn't strong enough on its own, they introduced a simple velocity descriptor before the classification step. They tested their approach on two widely used action datasets, the Weizmann and the KTH datasets, using various configurations and experiments, after performing supervised action classification. With a perfect accuracy rate of 100 percent on the Weizmann dataset and a high success rate of close to 90 percent on the KTH dataset, their technique yielded superior outcomes.

The strength of their proposed method is that it was able to provide robust recognition of human actions with a simple and compact representation when comparing to complex representations. Their method also produced an easy-to-use, quick-to-identify action system with high accuracy in even the most difficult situations. The reliance on silhouette extraction is one of their approach's shortcomings. Even with faulty silhouettes, their method produced high identification rates, indicating its resistance to noise. The solution to this problem is to increase the success rate of the silhouette extraction technique and decrease the quantity of faulty silhouettes.

2.3 Deep Learning Approaches

2.3.1 Fully Convolutional Neural Network (FCN)

In 2018, Sabokrou et al. [5] proposed a new method for anomaly detection which is a modified pre-trained convolutional neural network (CNN). The CNN are not created from ground up but was fine-tuned. CNNs have been shown to be successful in establishing adequate data processing algorithms for a range of applications, including object detection and activity recognition. However, there are some issues with using CNNs to detect anomalies in real-world videos, such as the Since CNN training is totally supervised, CNN approaches are too slow for patch-based methods, detecting anomalies in real-world videos is limited by the inability to train massive volume of data from non-existing classes of anomalies. To overcome these problems, they proposed a new fully convolutional network (FCN) based structure to extract different features of videos regions. This new technique combines an AlexNet model's pre-trained convolutional layers with an additional convolutional layer. AlexNet is a pre-trained image classification model that makes use of ImageNet as well as the MIT locations dataset. Hence, the extracted features will become discriminative enough for anomaly detection in video. The following Figure 2.3.1 is the workflow of their proposed method.

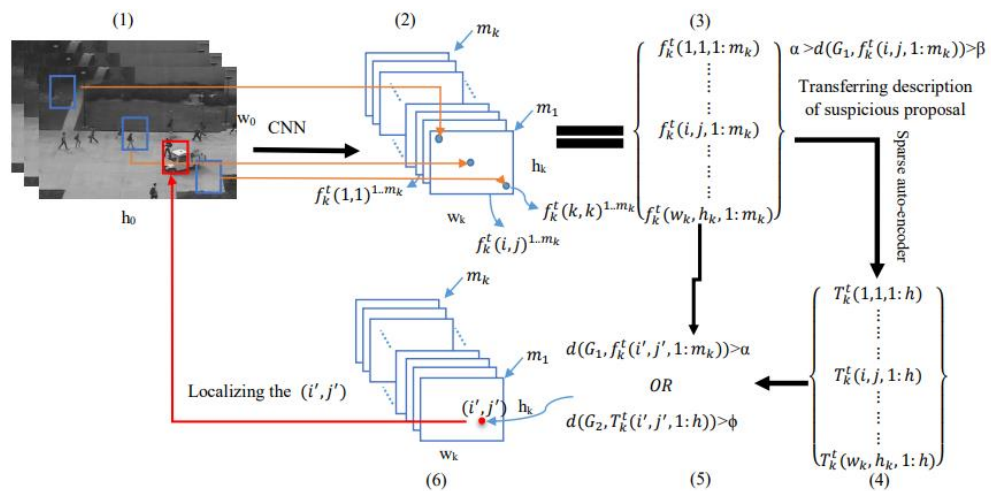


Figure 2.3.1: Schematic sketch of FCNN detection method [5]

First, input frames are sent to a pre-trained FCN. Then, the regional feature vectors are generated and verified using the Gaussian classifier. Patches that deviate significantly from the classifier as a normal reference model are classified as abnormal. A sparse auto-encoder is applied to suspicious regions that are fitted with low confidence and these regions are labelled on another Gaussian classifier. For anomaly detection, both Gaussian classifiers was fitted to all the normal regional features generated by the FCN and the regional features that are higher than certain threshold on their distance to the Gaussian classifiers are abnormal. The next convolutional layer contains all the suspicious regions, and it is trained on all normal regions created by the pre-trained FCN. Then, using a convolutional layer and a mean pooling layer, they performed localization to determine the location of descriptions that identify abnormality. In their proposed method, the Gaussian classifiers were implemented to identify the anomalous regions, while FCN was solely applied to extract the regional feature.

The datasets that they used are UCSD Ped2 and Subway. For experimental result of their proposed approach, they evaluated the performance of their approach using a variety of metrics such as ROCC, AUC, EER and performed run-time analysis to detect the speed of their method. The results reveal that their approach can detect anomalies at a speed of roughly 370 frames per second and is both fast and accurate.

2.3.2 Social-LSTM

In 2016, Alahi et al. [6] suggested a model called "Social" LSTM (Social-LSTM) that can predict the trajectories of all the individuals in a scenario by using common sense norms and cultural practices that humans use to move in public environments. The capacity to model these norms and utilize them to comprehend and forecast human mobility in complicated real-world contexts is immensely useful for a variety of applications such as the development of intelligent tracking systems in smart settings. Alahi et al. stated that Social-LSTM can learn common correlations between trajectories that occur at the same time automatically. Without the need for any additional inputs, this model uses existing human path datasets to learn common sense rules and cultural practices that human use.

As people move at varying speed, accelerate at varying rates, and walk in different postures, each person has a specific motion pattern. An ideal model is one that can

understand and learn such individual-specific motion properties from a limited set of baseline data corresponding to the individual. LSTM networks have been proven to learn and generalise the characteristics of discrete sequences like handwriting. As a result, they developed an LSTM-based solution for their own trajectory problem.

Figure 2.3.2 below is the overview of their Social-LSTM method. Each trajectory in a scenario was represented by a distinct LSTM network. A social pooling layer is then used to link the LSTMs together. This pooling layer allows spatially nearby LSTMs to share data. The Social pooling for one individual in the scene is shown at the bottom. The hidden states of all LSTMs within a given radius are pooled and utilized as input in the following time step.

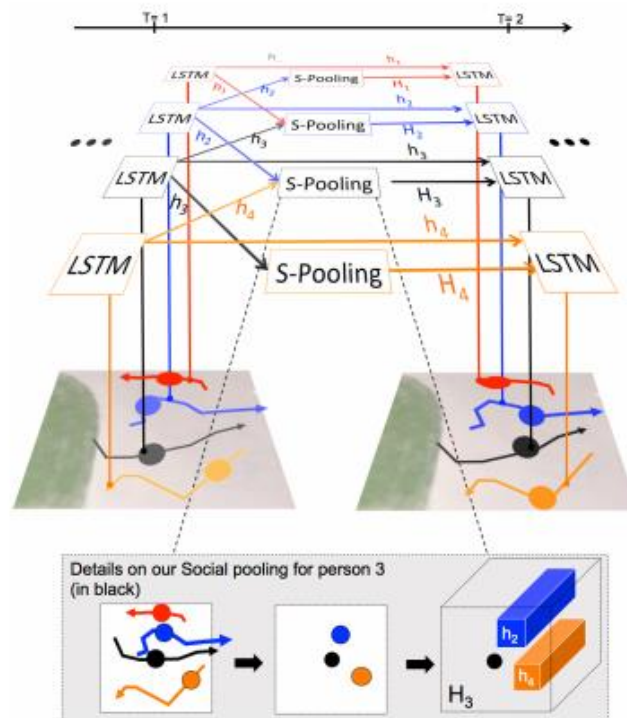


Figure 2.3.2: Overview of Social-LSTM method [6]

Individuals change their paths based on what their neighbours are doing. These neighbours are influenced by those in their near vicinity, and their behaviour may change over time. These time-varying motion-features were intended to be captured by hidden states of an LSTM. To collaboratively reason across numerous people, they shared the states amongst neighbouring LSTMS. Then they introduced "Social" pooling layers as shown in the Figure 2.3.2. At each time step, the LSTM cell gets pooled

hidden-state information from neighbouring LSTM cells. They employed the hidden state to forecast the distribution of the trajectory position for position estimation. To forecast the coordinates, they employed a bivariate Gaussian distribution approach. The parameters of the LSTM model are found by reducing log-Likelihood loss. The model was trained by reducing this loss across all trajectories in the training dataset. Any feature set from nearby trajectories can be pooled using the "Social" LSTM model. Their method learns to alter a route to avoid colliding with nearby objects. They put their strategy to the test on a variety of public datasets and discovered that in several circumstances, it outperformed state-of-the-art alternatives.

The strength of their method is that their Social-LSTM model has accurately predicted numerous non-linear behaviour's emerging from social interactions. Their method for learning interactions between people was more generic data-driven than previous models that were based on relative distances and regulations for specific scenarios. Their method has a limitation in that it only looks at human-to-human interactions and overlooks human-to-space interactions. The answer is to add the local static-scene image as a new input to the LSTM in their framework for describing human-space interaction. This could make it possible to simulate human-human and human-space interactions in the same environment.

2.3.3 Survey of Vision-based Methods for Action Recognition

In 2011, Weinland et al. [7] conducted a survey of vision-based approaches for action characterisation, separation, and identification. Action detection is a hot topic in computer vision research, and it has a wide range of uses, including video surveillance. Feature extraction which entails extracting discriminative pose and gesture signals from video that are unique to human behaviour is a critical task in action recognition. Action learning and classification are techniques for inferring statistical models from derived features to detect new feature data. Action segmentation is necessary to reduce sequences of motions into single action instances that are compatible with the collection of early training events used to create the algorithms.

The body parts involved, such as facial expressions, the chosen image features, such as optical flow, and the classifier model, such as nearest neighbours, can all be used to

identify vision-based approaches for capturing, sectioning, and identifying human movements.

The Figure 2.3.3 shown is a general process flow for generic action recognition system. [7]

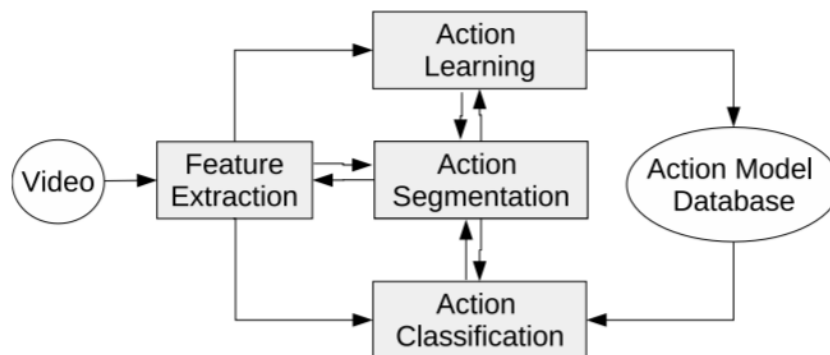


Figure 2.3.3: A general process flow for generic action recognition system [7]

Weiland et al. divided approaches into categories depending on how they collect the temporal and spatial pattern of actions, segment and recognise activities from a video stream, and deal with camera perspective variations. Weiland et al. also discovered many different ideas and picked 150 publications that show considerable progress in single-person, small-vocabulary, full-body movement recognition over the last decade. The methods that represent temporal and spatial data over local feature points have the advantage of avoiding the need to recognise people or body parts, which can be difficult in uncontrolled environments. The absence of publicly available and meaningful datasets is the method's flaw. Dealing with surveillance footage and video data from the Internet, on the other hand, can help solve the problem.

2.3.4 Survey of Deep Learning Based Approaches for Video Anomaly Detection

In 2018, Kiran et al. [8] examined state-of-the-art deep learning-based techniques for anomaly detection and categorised them by model type and detection conditions. Kiran et al. also conducted basic tests to better comprehend the various methodologies and to establish evaluation criteria for spatio-temporal anomaly detection. Based on the past knowledge utilised to generate the representations that define anomalies in this review work, Kiran et al. categorized the various unsupervised learning models for detecting anomalies in videos into three categories.

Rebuilding-based, time-dependant predictive, and procreative models are among them. Rebuilding-based models generate interpretations that lower the reconstruction error of normal-distribution training data. By viewing videos as a time series, time-dependent prediction models analyse the spatio-temporal relationship. During the training process, this model learned to reduce forecast error on a spatio-temporal order. With an emphasis on estimating the gap between sample and distributions, Procreative models seek to synthesise data from the training distribution with the least amount of reconstruction error.

Each of these approaches focuses on acquiring prior knowledge that may be utilised to build a model for the video anomaly detection problem. The methodologies used are limited by the difference in temporal scale of movement patterns among several surveillance recordings with similar background and foreground. The solution is to build a model that is temporal warping invariant and practical.

2.3.5 Overview of Research Methodologies in Deep Learning Based Anomaly Detection

In 2019, Chalapathy and Chawla [9] presented an overview of research methodologies in deep learning-based anomaly detection, as well as an assessment of their acceptance and effectiveness in diverse application areas. Chalapathy and Chawla have classified state-of-the-art research techniques into various categories based on the assumptions and methods used. For distinguishing between normal and abnormal behaviour in each category, Chalapathy and Chawla provide the basic anomaly detection technique, as

well as its variations and crucial assumptions. As the size of the data grows larger, deep learning surpasses conventional machine learning, as seen in Figure 2.3.5.1.

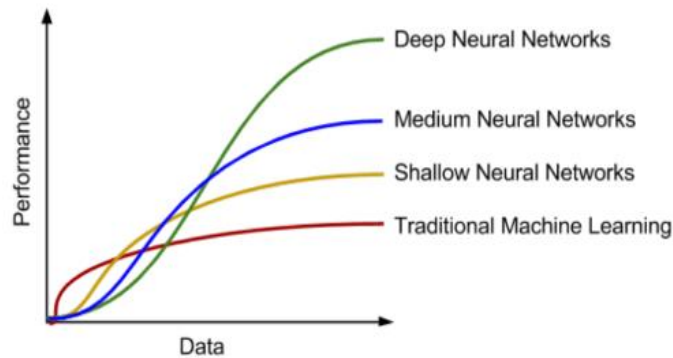


Figure 2.3.5.1: Deep learning-based algorithms Versus traditional algorithms performance comparison [9]

Anomaly detection techniques based on deep learning have increased in popularity over the years and have been used to a variety of tasks, as shown in Figure 2.3.5.2.

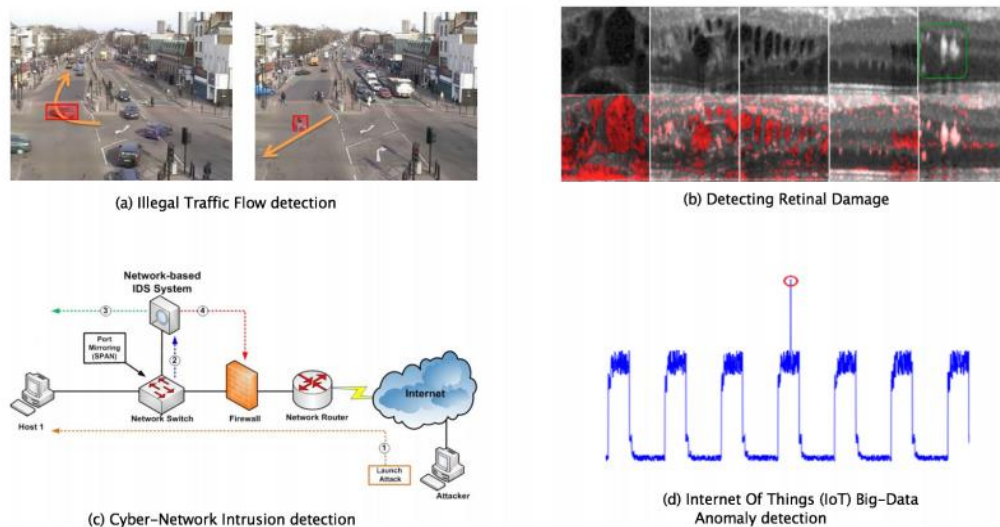


Figure 2.3.5.2: Deep learning-based anomaly detection algorithms successful applications. [9]

Every deep anomaly detection (DAD) method investigated in this study has its own set of benefits and drawbacks. Knowing which anomaly detection technique is optimal for a specific anomaly detection problem scenario is very important. In a setting with an equal number of labels for both normal and anomalous cases, classification-based supervised DAD approaches are superior choices. The supervised DAD approach's

computational complexity is a crucial concern, especially when the technique is utilised in a real-world environment. While classification-based, supervised, or semi-supervised techniques necessitate extensive training, testing is usually quick because to the use of a pre-trained model.

As label collection is a costly and time-consuming operation, unsupervised DAD approaches are extensively used. The models are less resilient when dealing with outliers because most unsupervised deep anomaly detection algorithms assume priors on the anomalous distribution. Hybrid models combine robust features from deep neural network hidden layers with the most successful standard anomaly detection techniques to create the models. Since it is unable to impact representational learning in the hidden layers, the hybrid model method is unsatisfactory. To better understand the benefits of this architecture in overcoming the limitations of current methods, more study and investigation is required.

2.3.6 Survey on Deep Learning Techniques for Video Anomaly Detection

In 2020, Suarez and Naval [15] conducted a survey on deep learning approaches for anomaly detection. This paper gave an overview of recent improvements in anomaly detection for videos, focusing on deep learning algorithms. It also covered the most regularly used datasets as well as the most used evaluation measures.

Regarding the final phase in recognizing anomalies, four types of current methodologies have been introduced: employing reconstruction error, forecasting future frames, classification, and scoring. These categories show the diversity of techniques as well as the difficulty of the task, which necessitates academics and practitioners to innovate and come up with effective alternatives.

End-to-end deep learning solutions necessitate a significant amount of data, which can be challenging for older datasets. However, some researchers have proposed large-scale datasets to help tackle this problem. One thing to keep in mind is that video data is difficult to label and collect, which is one of the reasons why, despite the high volume of data publicly available on video sharing platforms, there haven't been as many large-scale datasets released. The solution to this difficulty is to employ judiciously unsupervised or weakly-supervised methods.

2.3.7 Optical Flow on Violence Detection

In 2022, Biswas and others [12] have conducted research studies that examine various methodologies for detecting violence in surveillance videos. The detection of violence was carried out at several stages. The frames were first retrieved from the surveillance video input. Second, the objects were identified, and their features were extracted. After then, those features were classified, and the abnormality was discovered. The Figure 2.3.7 below was the general process of violence detection.

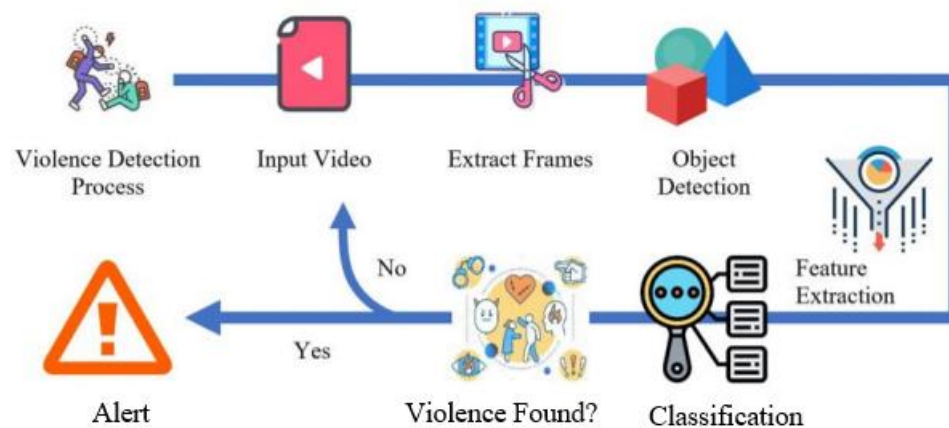


Figure 2.3.7: General Process of Violence Detection. [12]

Optical flow is one of the state-of-the-art violence detection techniques. Optical flow is a technique for calculating image's motion and tracking a single video feature. One of the research papers proposed a system for detecting violence in sensitive locations using machine learning and computer vision techniques. [12] The process begins with the frames being extracted according to the motion tracker. The optical flow is then calculated, followed by the calculation of mean of magnitude change vector and form the histogram. Then, to detect whether there are any violent events, the Violent Flow (ViF) descriptors feature extraction method was applied to various machine learning techniques such as SVM, Random Forest, and others.

The dataset used was Crowded which contains 246 surveillance video clips. The optical flow approach attained a 90% accuracy in this framework's experimental results. Optical flow's strength in violence detection is that it enhances accuracy while being computationally inexpensive. Optical flow approaches, on the other hand, have a flaw in that the smoothness of their motion prevents discontinuities in motion across object boundaries in the scene.

2.4 Critical Remarks of previous works

For the comparison for all the approaches reviewed, the experimental results proved that deep learning approaches were more effective than non-deep learning approaches and conventional method in video anomaly detection issue. Non-deep learning approaches like MoSIFT and sparse coding, temporal augmented network faced difficulties in some difficult conditions such as fast motion, low resolution, and others although these methods have high accuracy in video anomaly detection. Conventional methods such as STIP spatio-temporal descriptors and Histogram-of-Oriented-Rectangles (HOR) to recognise the human actions are straightforward and simple, however these methods have the issues of the accuracy being either too low with correct input or too perfect with some inaccurate input which showed these methods' robustness to noises and inconsistent in video anomaly detection. Deep learning approaches like Fully Convolutional Neural Network (FCN), Social-LSTM and optical flow have showed effective performance towards video anomaly detection.

Among all the deep learning approaches, Fully Convolutional Neural Network (FCN) was the best deep learning approach. The dataset that was utilised to assess the performance of FCN are USCD and Subway datasets. The objects in the USCD dataset are walkers with a range of small crowd sizes to big crowd sizes. An anomaly is defined as an object that appears unexpectedly, such as an automobile, wheelchair, or bicycle. Subway dataset captured two sequences at the entrance and exit of a subway station. People entering and exiting the station usually behave normally. People going in the wrong way or avoiding payment are examples of abnormal events.

Using ROC curve, EER, and AUC, FCN's results are compared to those of state-of-the-art approaches. The frame level and pixel level measurements are utilised, with the frame level indicating that if one pixel finds an anomaly, it is abnormal, and the pixel level indicating that the frame shows an anomaly if at minimum 40% of the anomalous ground truth pixels are occupied by pixels discovered by the technique.

The approach's frame-level EER for the USCD dataset is 11 percent, compared to the best overall result of 10%. The proposed methodology has a pixel-level EER of 17 percent, which is 2 percent better than any other state-of-the-art solution in the pixel-level EER metric. For Subway dataset, the method was evaluated in both entrance and exit scenes. The ROC analysis indicated that this method outperforms other methods.

On both departure and entry scenes, the suggested technique outperforms state-of-the-art algorithms by 90.2 percent and 90.4 percent, respectively, on AUC and EER metrics, respectively. The total time it takes to detect an anomaly in a frame is about 0.0027 seconds. As a result, this method achieved 370 frames per second, which is far quicker than any of the other state-of-the-art methods.

This method was the closest to achieve the objectives for this project's solution since its solution able to operate at a high speed and able to detect the anomalies in real-time with a high accuracy based on the human action datasets. Table 2.4 below compared the various methods discussed in this section to support the statement. Although the motion-aware feature with a temporal augmented network used the same dataset as this project and achieve a promising result in terms of precision and timing, but the solution did not run in real-time which causes this method not suitable for this project. The same reason applied to other methods discussed in the table.

However, although FCN was the closest to achieve the objectives for this project's solution, FCN has several convolutional layers which causes the training process to take a lot of time. This also means that high processing power is required to train the model and a good graphic card is needed to accelerate the training process. Memory demanding and time consuming of the FCN model during the training time which cause this solution is not optimal for the project due to lack of time and high processing power. So, instead of FCN, optical flow was chosen to be the most optimal method out of all approaches reviewed. Optical flow was able to detect the motion of fighting events accurately at a high speed in real-time which achieve the project's objectives. The strength of optical flow in fighting events detection is that it improves accuracy while being computationally inexpensive. It also does not require long processing time and high processing power which is suitable for the project.

CHAPTER 2 LITERATURE REVIEW

Methods	Type of recognition result achieved	Accuracy	Dataset Used	Timing
Motion-aware Feature with Temporal Augmented Network	To learn a motion-aware feature with a temporal augmented network and use an attention block to include the temporal context into the Multiple Instance Learning (MIL) ranking algorithm for video anomaly detection.	The motion-aware feature learned from temporal augmented network achieved a competitive performance in terms of anomaly detection AUC score which was 72.1%.	UCF Crime dataset	400+ fps but not in real-time
MoSIFT Feature and Sparse Coding	To detect violence by recognising distinctive local features based on their form and motion, extracting the low-level description of a query video using the MoSIFT technique, and further processing the selected MoSIFTs using sparse coding.	Performance is measured using mean prediction accuracy, standard deviation and AUC. The method achieved a $94.0 \pm 1.97\%$ on ACC \pm SD and 0.9666 on AUC for Hockey Fight dataset and $89.05 \pm 3.26\%$ on ACC \pm SD and 0.9357 on AUC for Crowd Violence dataset.	Hockey Fight and Crowd Violence dataset	-
STIP Spatio-temporal Descriptors	To recognise the human action and create compact and descriptive motion representations using the spatio-temporal descriptors.	The STIP approach was compared with the MoSIFT approach using the same datasets. MoSIFT shows promising performance with the highest accuracy of 89.5% while the STIP's highest accuracy was only 59.0% under all situations.	Hockey Fight dataset	-
Histogram-of-Oriented-Rectangles (HOR)	To recognise human behaviours based on their posture rather than having to cope with a complex representation of dynamics.	Perfect accuracy rate of 100% on Weizmann dataset and an 89.4% accuracy rate on KTH dataset. But the approach achieved strong recognition rates even with inaccurate silhouettes, demonstrating its robustness to noise.	Weizmann and KTH dataset	-

Fully Convolutional Neural Network (FCN)	To detect anomalies based on two different datasets such as objects that appears unexpectedly and human behaviour in the subway station in real-time with high accuracy and speed.	For UCSD dataset, the approach's frame-level EER which was 11% and pixel-level EER which was 17% surpassed any other state-of-the-art method in terms of performance. For Subway dataset, the method had a better AUC and EER measures on both exit and entrance scenes which is 90.2% and 90.4% respectively.	USCD and Subway dataset	The total time it takes to detect an anomaly in a frame is about 0.0027 seconds which achieve 370 frames per seconds.
Social-LSTM	Jointly reason across numerous people to forecast human trajectories in a scene with two publicly available datasets which can help to design an intelligent tracking system in smart environment.	Outperforms state-of-the-art approaches in term of error reduction which was 1.07 on ETH dataset and 0.77 on UCY dataset.	ETH and UCY human trajectory dataset	Forecast trajectories for a fixed period of 4.8seconds
Optical Flow	To detect violence by using optical flow methodologies and used it to perform classification by applying various machine learning techniques.	Have achieved a 90% accuracy with Crowded dataset which contains 246 clips of surveillance videos.	Crowded dataset	-

Table 2.4 Comparison for different approaches

CHAPTER 3

Proposed Method/Approach

In this chapter, we present the design specifications of our project including general work procedure, system performance and verification plan. We also explained our system design and the implementation challenges.

3.1 Design Specifications

3.1.1 Methodologies and General Work Procedure

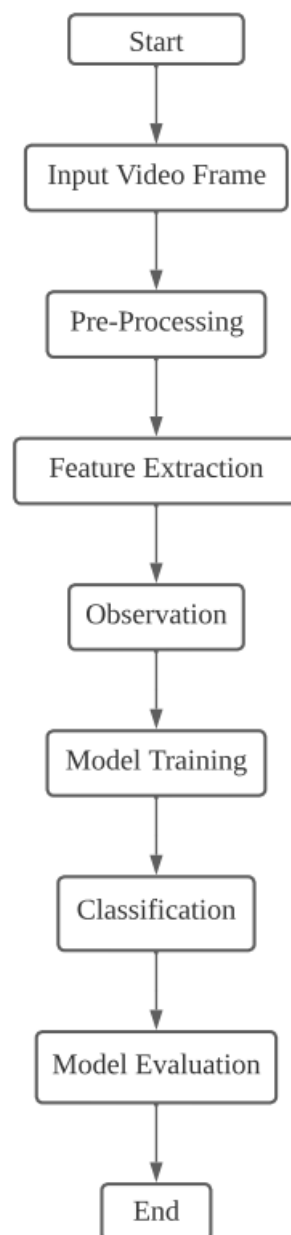


Figure 3.1.1 Flowchart of General Work Procedure

Figure 3.1.1 is the flow chart of the general work procedure for the project. First, the input video frames will undergo pre-processing steps such as conversion to grayscale to suppress noises in the background.

Then, the important features of the input video frames are extracted using optical flow. Optical flow was used to find out the pattern of observable motion of image objects between two consecutive frames generated by the movement of human in the dataset videos. It's a two-dimensional vector field in which each vector is a displacement vector indicating the movement of points from the one frame to next. Many existing anomaly detection approaches have used optical flow and it was generally best suited for detecting regions of unusually high velocity. [16] In this project, the fighting scenes should have the regions of unusually high velocity which is the reason why optical flow was used. There are two types of optical flow which are sparse and dense optical flow. Sparse optical flow was used in the preliminary work, and it chose a sparse feature set of pixels such as interesting features like corners to track its motion.

After extracting the important features of the input video frames and drawing the movement pattern using optical flow, various kinds of methods are used for observation purposes to prove that the optical flow generated was meaningful and it was suitable for the project solution. Then, the meaningful data of optical flow was obtained and fetched to train the classifier model. Support Vector Machine (SVM) was chosen to be classification model in this project. Classification are then performed using the trained SVM to detect the fighting events. Lastly, model evaluation will be performed to evaluate the performance of the SVM classifier.

3.1.2 Tools to use

Hardware

1. Laptop

Operating System	Window 10 64-bit
Processor	Intel(R) Core (TM) i5-8300H CPU @ 2.30GHz
RAM	8 GB
Graphic Card	GTX 1050

Table 3.1.2: Hardware Specification

Software

1. Microsoft Visual Studio 2019

It is an IDE that programmers use to do tasks like create, edit, build, debug, and deploy applications.

2. OpenCV

It is an open-source computer vision, machine learning, image processing, and real-time operation library. OpenCV is capable of a wide range of tasks, including image and video processing as well as object detection.

3. Jupyter Notebook

It is an open-source web tool that lets you make and share documents with live codes, documentation, graphs, plots, and visualizations.

3.1.3 System Performance Definition

For this project solution, the system aims to achieve equal to or higher than 90% in terms of accuracy and 30 frames per second. A few categories of fighting events should be detected accurately and achieve optimal frames per second to give early warning of such incidences. The few categories of fighting events should be categorized in fighting videos that contains interesting patterns, fighting videos that contains non-interesting patterns and normal videos.

3.1.4 Verification Plan

The system should classify a few categories of fighting events with high accuracy and timing, but it may affect by some situations that can cause some issues during classification. A situation was shown below:

- Fighting event happens on cluttered background with irrelevant object and people passing by in the fighting scene; the system should be able to recognize the fighting event accurately.

Therefore, several verification steps should be done to ensure the accuracy and consistency of the system. The verification steps are shown at below:

I. The cluttered background.

Procedure Number	P1
Method	Testing
Applicable Requirements	Recognize the fighting event happening in the background.
Purpose/Scope	To recognize the category of fighting event from a cluttered background such as the video with irrelevant object and many people passing by.
Items Under Test	Fighting scene video.
Precautions	If majority of the noises is not removed, it may cause inaccurate result.
Equipment/Facilities	Laptop.
Acceptance Criteria	The system classified the fighting events accurately and display the timing, accuracy, and the category of fighting event on the screen.
Procedures	1. Input video frames as input 2. Timing, accuracy, and the category of fighting event is displayed by system after performing classification.
Troubleshooting	Repeat the procedure.

Table 3.1.4.1: Verification Plan P1

3.2 System Design/Overview

3.2.1 Flowchart

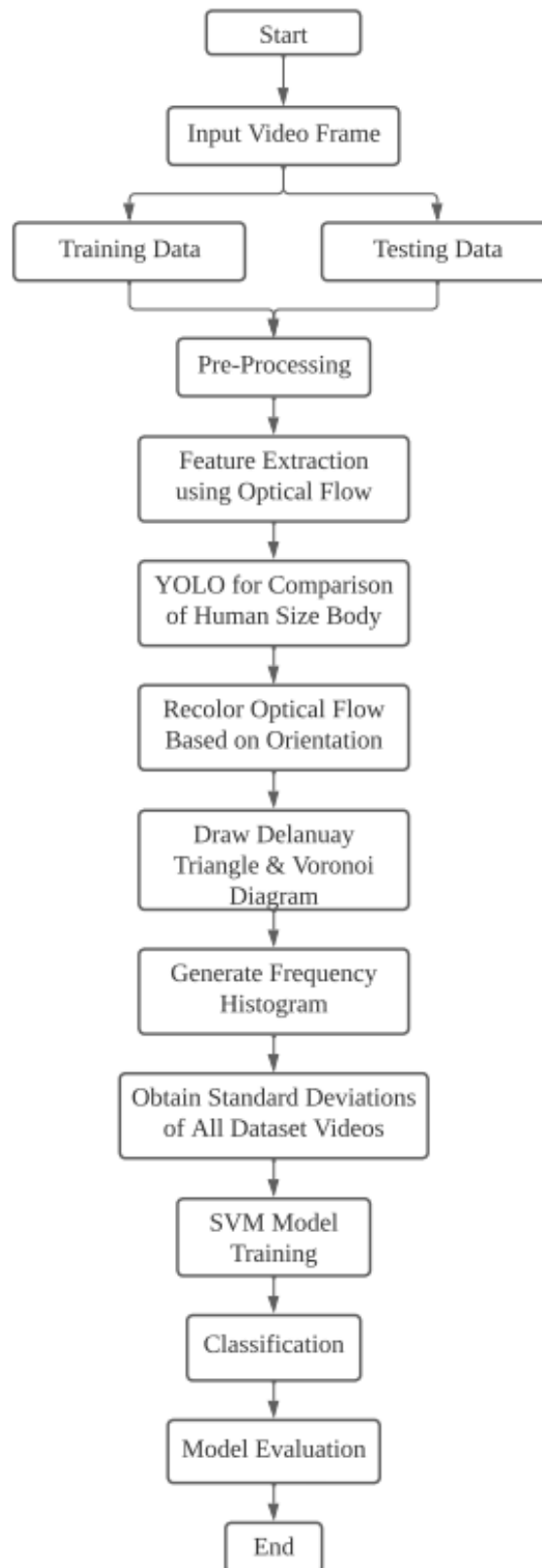


Figure 3.2.1: Flowchart of System Design

3.2.2 Descriptions of System Design

First, the input video frames are the fighting and non-fighting videos selected from the UCF_Crimes dataset. The input video frames will be split into 80% training data and 20% testing data. The training data and testing data then undergoes pre-processing steps such as noise removal by conversion of the video frames into grayscale video frames. Another pre-processing step was dilation was done to increase the white region for the difference between two consecutive frames by adding pixels to the boundaries of the objects. The important features between two consecutive frames of input videos are then extracted using optical flow. The optical flow of the important features was calculated, and the tracks are drawn out using random colour lines.

Next is the observation process to prove that the optical flow generated was meaningful and it was suitable for the project solution. The first method was using YOLOv3 (You Only Look Once) approach. The YOLOv3 (You Only Look Once) approach is one of the most popular deep learning-based object detection algorithms. Object detection is handled as a regression problem in the YOLOv3 approach. Using a single feed forward CNN, it predicts the class probabilities and bounding box offsets from the images. [17] YOLOv3 was used to compare the size of human body to the optical flow generated to validate that the optical flow generated on the fighting scenes was accurate and meaningful. The second method is to use the orientation of good feature points of optical flow to recolor the optical flow. The orientation value represented the Hue value for HSV. The saturation (S) and intensity (V) values are manually set to maximum value which is 255 to show the color better. The HSV value are converted to RGB value to draw the optical flow. The purpose is to show that the movement of same direction should generate the same color of optical flow and vice versa. So, the optical flow of fighting scenes should have many different colors since fighting involved complex movements and many different directions while the color of optical flow of normal scenes should be the same for the people walking in the same direction.

The next method is to draw Voronoi Diagram and Delaunay Triangle using the feature points of optical flow. It was drawn to construct and outline proximal regions around individual data points using polygonal borders. In our solution, the good feature points of optical flow represented the individual data points. The purpose was to show the difference of features structures and the proximity between fighting and normal videos.

The frequency histogram of orientation of optical flow was also generated for both fighting and normal videos to observe the value differences between the videos. The orientation value of optical flow for fighting video will normally fluctuate greatly while it will not fluctuate much for normal video.

After observation, the optical flow program was run to obtain the standard deviations of orientation of optical flows on all the dataset videos. Normalization was performed on all the orientation values obtained. The normalized values were then used to fit and train the classification model which is Support Vector Machine (SVM). After the model training, classification was done to detect the fighting events on the dataset videos. The last step is to perform model evaluation to determine the system performance by using confusion matrix, classification report, AUC-ROC curve and also the learning curve of the trained SVM.

3.3 Implementation Issues and Challenges

The method that was originally to be used in this project was Fully Convolutional Neural Network (FCN). FCN was used to tackle the issues of CNN method such as patch-based methods are too slow for CNN approaches, and because CNN training is completely supervised, finding anomalies in real-world videos is constrained by the inability to learn massive volumes of data from non-existing classes of anomalies. [5] The difference between FCN and CNN is to replace the last fully connected layer of CNN with a convolutional layer which indicates that FCN only contain convolutional layers without any fully connected layers. However, one of the implementation challenges for this approach is that FCN is significantly slower due to the operation such as maxpool. The FCN has several convolutional layers which causes the training process to take a lot of time. This also means that high processing power is required to train the model and a good graphic card is needed to accelerate the training process. Memory demanding and time consuming of the FCN model during the training time under time restriction condition in this project caused the project's solution to be changed to optical flow approach.

Another implementation challenge was when recolouring the optical flow using the orientation values, the solution had to convert the HSV value one by one using OpenCV function to obtain the RGB value to recolour optical flow. This has greatly reduced the frame processed per seconds and increased the timing of the program. However, this

challenge will only cause memory demand and time-consuming issues and will not affect the accuracy result of the solution.

Last implementation challenge is that some of the fighting events happening in the dataset videos might not be detected by the optical flow. It was caused by the reasons such as the human involved in the fighting events are very far away from the surveillance camera so their movement cannot be detected and traced. Another reason is that if there are obstacles that blocked the sight of the fighting events, the optical flow might not be able to detect the movement of the fighting events. To overcome this challenge, the fighting events have been categorized to fighting videos that contains interesting patterns, fighting videos that contains non-interesting patterns and normal videos. The fighting videos that contain the above reasons that causes the fighting events unable to be detected in the videos are categorized as fighting videos that contains non-interesting patterns. Fighting videos that contains interesting patterns are the videos that optical flow can detect the fighting event accurately while the normal videos are the videos that does not involved the fighting event.

Chapter 4

Project Implementation

In this chapter, we present the dataset used for the project and explain the implementation details.

4.1 Dataset

Dataset used for this project is called UCF-Crime which consists of surveillance videos that cover 13 real-world anomalies, including fighting, vandalism, etc. This dataset also contains surveillance videos of normal event which no crime happened. The dataset used in this project has 50 fighting videos and 50 normal videos.

4.2 Implementation Details

4.2.1 Pre-Processing

Firstly, the input video which are the fighting and normal videos selected from the UCF_Crimes dataset are imported. The input videos dataset was split into 80% training data and 20% testing data. Then, two consecutive frames which are the suitable previous and current frames of the input videos are captured. The difference between the two consecutive frames was calculated by subtraction to capture the motion. The frames then undergo pre-processing steps which are conversion to grayscale and dilation. Conversion to grayscale was done using OpenCV function `COLOR_BGR2GRAY` which convert the RGB-coloured input video frames to gray colour. Dilation was done to increase the white region for the difference between two consecutive frames by adding pixels to the boundaries of the objects. Figure 4.2.1 show the pre-processing steps done on the input video frames which are conversion to grayscale and dilation process.



Figure 4.2.1 Pre-Processing Steps

4.2.2 Optical Flow

In the optical flow program, initialization was done to get the suitable previous and current frames of the input videos. If there is no motion in the frames of input videos, it will be unable to generate the optical flow. After getting the suitable previous and current frames, OpenCV function called `goodFeaturesToTrack()` was used to find the good points which are the corners in the first frame. OpenCV Mat is an n-dimensional dense array class which usually used to define an image. Mat object named `OPcanvas` image was created to draw the optical flow. Then, OpenCV function called `calcOpticalFlowPyrLK()` was used to calculate the optical flow by passing in the previous frame, previous points and next frame. The good points of the frames then get selected and the tracks for the good points are drawn. After that, the previous frame and previous points are updated to the new suitable frame and points. These steps are repeated until the input video's last frame is reached and an output video of the optical flow will be generated. Figure 4.2.2.1 below is the screenshot of the output video of the optical flow on fighting video.

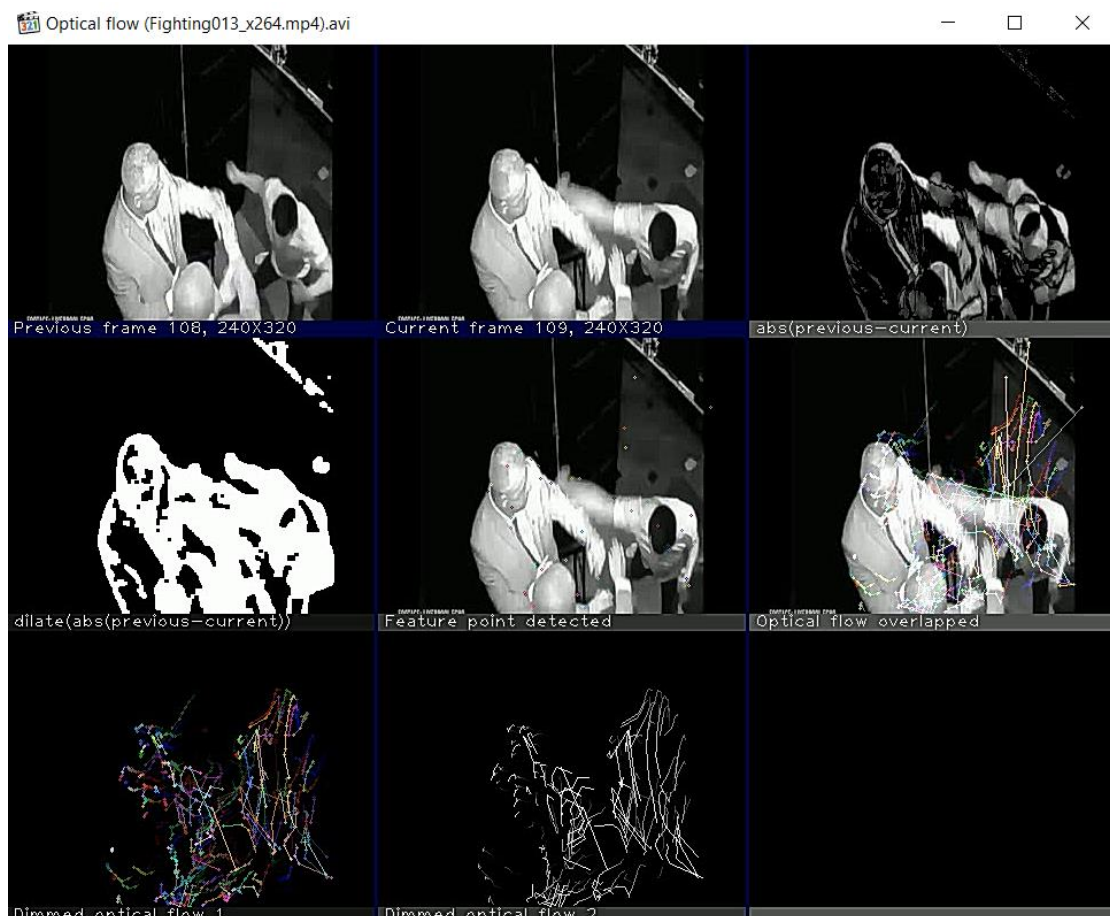


Figure 4.2.2.1 Optical Flow of Fighting Video


```

height = 240    width = 320
FPS = 30       total frames = 1434

----- Original dimensions:
height = 240    width = 320
----- Dimensions after resize():
height = 240    width = 320

Time spent for this operation: 34.358
Frame processed per second: 41.737

```

Figure 4.2.2.2 Running Information of Optical Flow

Figure 4.2.2.2 is the screenshot of the running information of optical flow in Figure 4.2.2.1. The optical flow program was able to process 41 frames per second. Based on the observation of the output video, it was deduced that the optical flow generated on the fighting scenes are very meaningful and it can be used to classify the fighting scenes. To support this statement, the optical flow of normal videos in the dataset was generated using same program. The goal was to see the difference of optical flow generated between fighting video and non-fighting video.

Figure 4.2.2.3 below is the screenshot of the output video of the optical flow on non-fighting video. We can see that the optical flow generated on the fighting video and non-fighting video has a very huge difference since the fighting scenes contain regions of unusually high velocity.

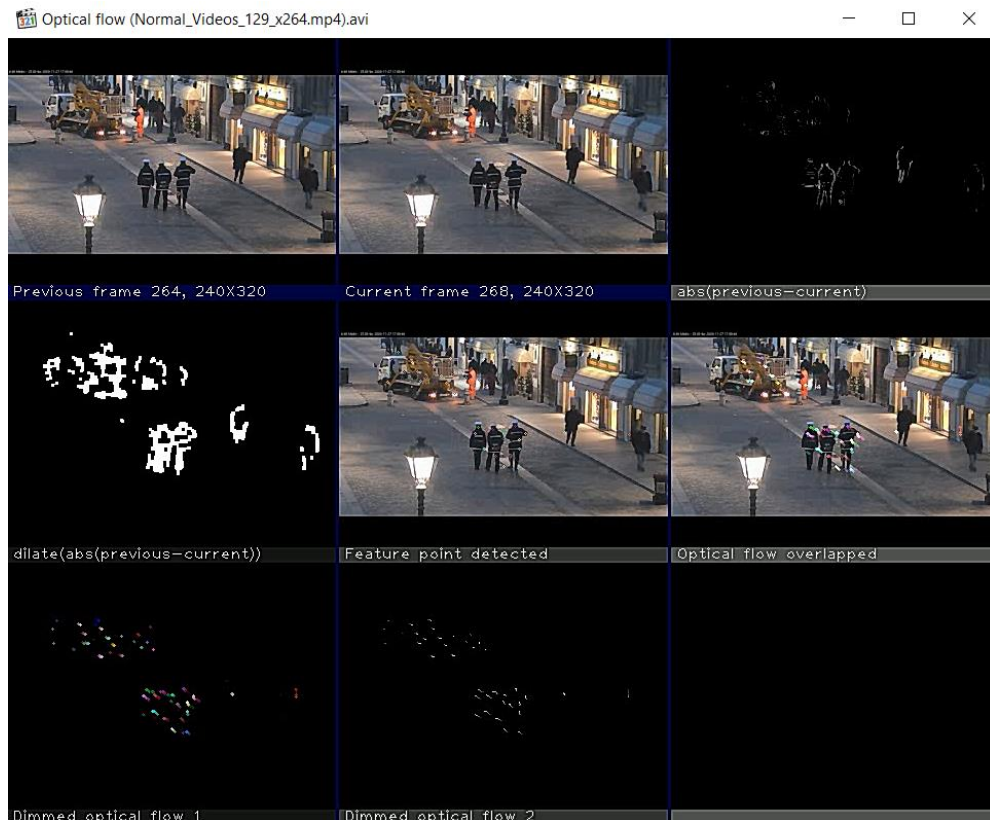


Figure 4.2.2.3 Optical Flow of Non-Fighting Video

4.2.3 YOLOv3

Then, YOLOv3 was used to compare the size of human body to the optical flow generated to validate that the optical flow generated on the fighting scenes was accurate and meaningful. In the YOLOv3 program, the neural network was trained with 100 types of objects including person, car, bicycle, etc. The predicted bounding box was drawn and the label for the class name and its confidence score also displayed on the top of the bounding box. Only the bounding boxes with the high confidence scores was displayed on the output video. Non-maximum suppression was performed to eliminate redundant overlapping boxes with lower confidence.



Figure 4.2.3.1 YOLO3 of Fighting Video

Figure 4.2.3.1 above is the screenshot of the output video of the YOLOv3 on the fighting video. It was observed that the the bounding box generated on Figure 4.4 have the same size compared to the optical flow generated on Figure 4.2.2.1. Thus, we can conclude that optical flow generated on Figure 4.2.2.1 was accurate and meaningful.

4.2.4 Observation using Orientation of Optical Flow

The good feature points are used to generate the optical flow before. Besides that, each good points also contains the orientation value of optical flow generated. The orientation value of optical flow generated can be obtained by using `angleCal()` function which was implemented to calculate the angle which ranged from 0 to 360 degree. Then, the orientation value of optical flow can be set as Hue value for HSV. The saturation (S) and intensity (V) values are manually set to maximum value which is 255 to show the colour better. Since OpenCV uses one byte of 256 values to represent each

component, the Hue value which ranged from 0 to 360 was divided by 2 which means that red colour contains value 0 and 180. The HSV was then converted to RGB values by using OpenCV function COLOR_HSV2BGR. Figure 4.2.4.1 was the lookup table created for HSV value from 0 to 179 to convert to RGB value.

HSV value: 0, 255, 255	BGR value: [0, 0, 255]
HSV value: 1, 255, 255	BGR value: [0, 8, 255]
HSV value: 2, 255, 255	BGR value: [0, 16, 255]
HSV value: 3, 255, 255	BGR value: [0, 25, 255]
HSV value: 4, 255, 255	BGR value: [0, 33, 255]
HSV value: 5, 255, 255	BGR value: [0, 42, 255]
HSV value: 6, 255, 255	BGR value: [0, 51, 255]
HSV value: 7, 255, 255	BGR value: [0, 59, 255]
HSV value: 8, 255, 255	BGR value: [0, 67, 255]
HSV value: 9, 255, 255	BGR value: [0, 76, 255]
HSV value: 10, 255, 255	BGR value: [0, 85, 255]

Figure 4.2.4.1 Lookup Table for HSV to RGB conversion

Then, the RGB values converted are used to draw the optical flow. This is to show that the movement of same direction should generate the same colour of optical flow and vice versa. We can observe that the optical flow generated during fighting scenes will have many different coloured while the optical flow generated during normal videos will have the same colour on the movement of same direction.



Figure 4.2.4.2 Coloured Optical Flow of Normal Video



Figure 4.2.4.3 Coloured Optical Flow of Fighting Video (1)



Figure 4.2.4.4 Coloured Optical Flow of Fighting Video (2)

From Figure 4.2.4.2 above, we can observe that in the normal video, the objects that are moving in the same direction will have the same colour of optical flow generated. From Figure 4.2.4.3, we can observe that in the fighting video, the people who are moving in the same direction before they started fighting have the same coloured of optical flow generated. However, from Figure 4.2.4.4, when they started fighting, the optical flow generated have many different colour since fighting introduced many different directions and will generate orientation values that have high variations.

4.2.5 Delaunay Triangulation & Voronoi Diagram

Voronoi Diagram was drawn to construct and outline proximal regions around individual data points using polygonal borders where Delaunay Triangulation was the dual graph of Voronoi Diagram. Firstly, a vector points was created and the good feature points of optical flow was inserted into the vector points. The vector points are then inserted into Subdiv2D class. It is a planar subdivision which can be used to compute Delaunay triangulation or Voronoi Diagram. Draw_point() function was then used to draw out the vector points. After computation using Subdiv2d class, the points drawn are used to show animation by drawing the Delaunay triangle using the draw_delaunay() function and drawing the Voronoi Diagram using the draw_voronoi() function.

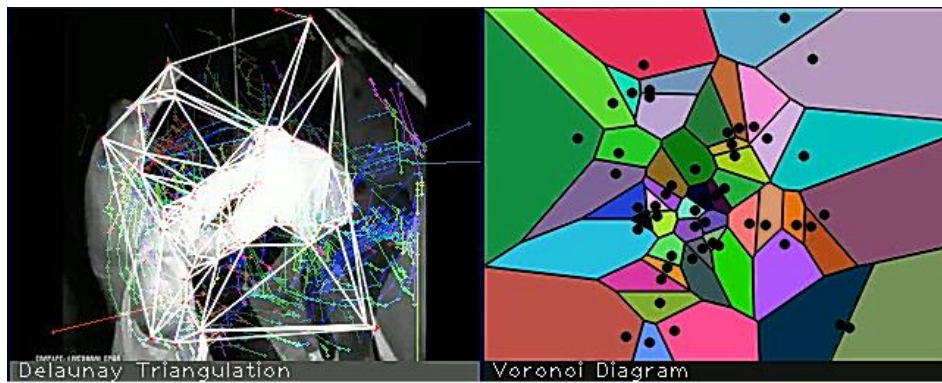


Figure 4.2.5.1 Delaunay Triangulation & Voronoi Diagram for Fighting Video



Figure 4.2.5.2 Delaunay Triangulation & Voronoi Diagram for Normal Video

From the above two figure which represents the Delaunay Triangle and Voronoi Diagram drawn for both fighting and normal videos, we can observe that the features structure of the Delaunay triangles formed are very different. Since fighting contains complex movement and high speed, the Delaunay triangle and Voronoi diagram formed on fighting video will change its structures rapidly while the Delaunay triangle and Voronoi diagram formed on normal video will not.

4.2.6 Generate Frequency Histogram using Orientation of Optical Flow

To generate the frequency histogram, the orientation of optical flow was stored and passed to a vector for storing. A function called drawHist() was created to get the vector that contains orientation values of optical flow and use OpenCV drawing function rectangle() to generate the frequency histogram. The histogram size which is the number of bins per each used dimension is set to be 180 since the range value of orientation of optical flow is between 0 and 180.

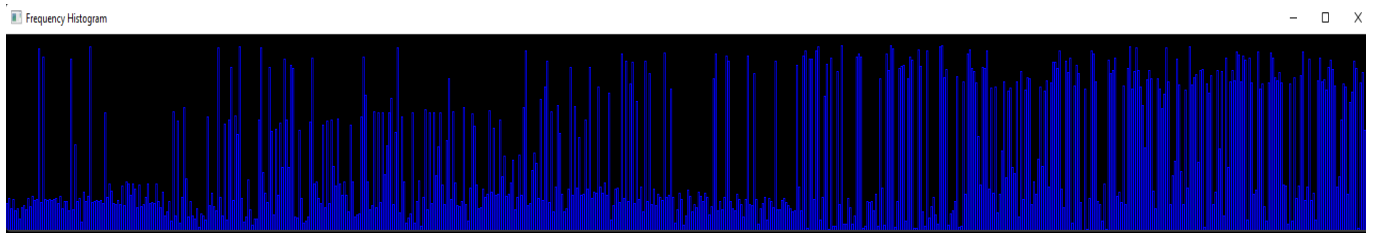


Figure 4.2.6.1 Frequency Histogram for Fighting Video

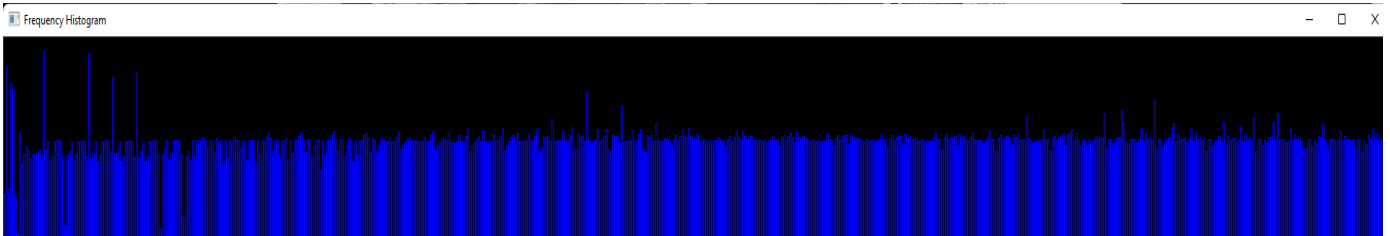


Figure 4.2.6.2 Frequency Histogram for Normal Video

From the figure above, the Y-axis of the frequency histogram represents the value of orientation of optical flow which range between 0 to 180 while the X-axis represents the index value of the vector that contains the orientation values. From Figure 4.2.6.1 and Figure 4.2.6.2 above, we can observe that the frequency histogram generated for fighting video contains the orientation values that fluctuate greatly while the orientation values of normal video does not fluctuate much. This show that the orientation of optical flow generated was meaningful and it can be used to train the classifier model.

4.3 Model Training

For model training, Support Vector Machine (SVM) was chosen to be classification model in this project. The type of SVM chosen was Classification SVM Type 1 which also known as C-Support Vector Classification. The kernel type of the model was set to be linear and the hard limit on iterations was set to be 100. To train the model, the standard deviations of orientation of optical flows have to be obtained first. Using the vector that store the orientation values of optical flow, OpenCV function `meanStdDev()` was called to calculate the mean and standard deviation of the vector elements. The standard deviations of orientation of optical flows was then obtained and stored after running the program on all the dataset videos and it will be utilized to fit and train the SVC model. After setting up the training and verification data, normalization was performed on all the data. The normalized data was then fetched to the SVC model to train the model.

Chapter 5

Experimental Result

5.1 System Performance

After setting up and training the model, the system performance for implemented system was evaluated. The normalized testing data was used to calculate the accuracy for the system implemented. Accuracy is the fraction of predictions the classifier model got right and measure the algorithm's performance in an interpretable way. The predicted test data label using the trained model was compared to the original test data label to calculate the accuracy score. The training and classification process was done on both Visual Studio 2019 and Jupyter Notebook with C++ and Python programming language as shown in Figure 5.1.1 and Figure 5.1.2. Jupyter Notebook was used to visualize our system performance for evaluation using Python.

```
Test on training data
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1

Test on verification data
0 0 0 0 0 0 1 0 0
1 1 1 1 1 1 1 1 1
```

Figure 5.1.1 Training and Classification Process on Visual Studio 2019 using C++

```
from sklearn import metrics
#accuracy
print("accuracy:", metrics.accuracy_score(testy,y_pred))

accuracy: 0.95
```

Figure 5.1.2 Training and Classification Process on Jupyter Notebook using Python

The accuracy score value for the test data was 0.95 which represents 95% of accuracy. This proved that the implemented system has achieved the system performance definition which was the accuracy of the system should be higher than 90%. Confusion matrix was also used to evaluate our system performance with the testing data. It is a matrix with the inputs versus the prediction results. Figure 5.1.3 below is the confusion matrix generated using our testing data and it was labelled with the classes which are fighting and normal.

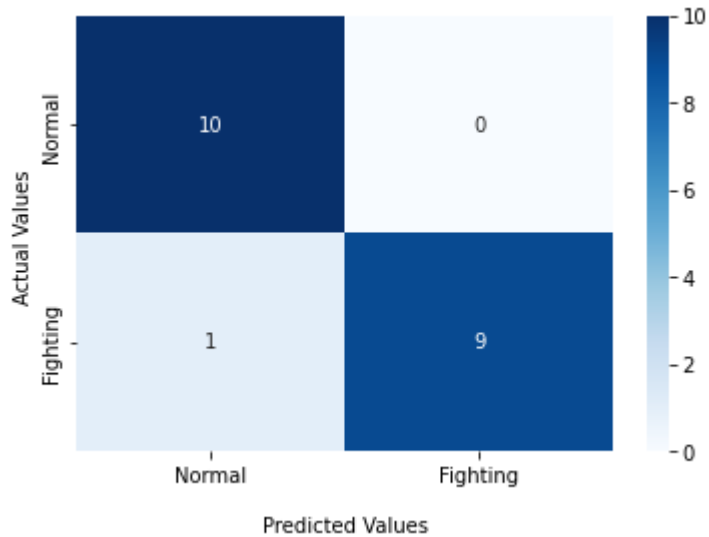


Figure 5.1.3 Confusion Matrix Visualization

There are four ways to check if the predictions are right or wrong.

1. True Positive - the case was positive and predicted positive.
2. True Negative - the case was negative and predicted negative.
3. False Positive (Type 1 Error) - the case was negative but predicted positive.
4. False Negative (Type 2 Error) - the case was positive but predicted negative.

From the confusion matrix in Figure 5.1, the fighting case represents the Positive class and the normal case represents the Negative class. We can observe that the classifier model has a very high accuracy and very less false predictions. There is only one result of fighting case misclassified as normal case. This misclassified result is one of the False Negative case (Type 2 Error). A classification report showing the classification metrics which is related to the confusion matrix was generated. Table 5.1 below is the classification report of the testing data.

Classification Report				
	precision	recall	f1-score	support
0	0.91	1.00	0.95	10
1	1.00	0.90	0.95	10
accuracy			0.95	20
macro avg	0.95	0.95	0.95	20
weighted avg	0.95	0.95	0.95	20

Table 5.1 Classification Metrics Report

Precision refers to the accuracy with which positive predictions are made. The ratio of true positives to the sum of true positives and false positives for each class is used to compute it. The percentage of positives accurately detected is known as recall. The ratio of true positives to the sum of true positives and false negatives for each class is used to compute it. The F1 score is a weighted harmonic mean of precision and recall, with 1.0 being the best and 0.0 being the worst. The amount of actual occurrences of the fighting and normal classes in the dataset is known as support.

From the report, 0 represents the fighting class and 1 represents the normal class. The fighting class has slightly lower precision score compared to normal class since it was observed there is a misclassified result for the fighting class from the confusion matrix. Other than that, we can observe that the precision, recall and f1-score values of both fighting and normal classes have a very high value. This have proved the validity of the implemented system. It has a high accuracy of the positive prediction and high ratio of positive samples that are correctly detected and a high F1 score.

Next, AUC-ROC curve was generated to evaluate the system performance. AUC represents the degree or measure of separability, whereas ROC is a probability curve. It indicates how well the model can distinguish between classes. The AUC indicates how well the model predicts fighting classes as fighting case and normal classes as normal case. The higher the AUC, the better the model is at distinguishing between fighting and normal events. An excellent model has an AUC close to 1, indicating that it has a high level of separability. AUC approaching 0 indicates a poor model, which means it has the lowest measure of separability.

The relative operating characteristic curve (ROC) was plotted using the True Positive Rate (TPR) and False Positive Rate (FPR) and shown in Figure 5.1.4 below. Area under curve(AUC) was also calculated. From the ROC curve, we can observe that the orange line is very close to the conceptual optimal location with the recall value of 1 and the area under curve has value of 0.95 which indicates that our system has a large recall and low false positive rate (FPR).

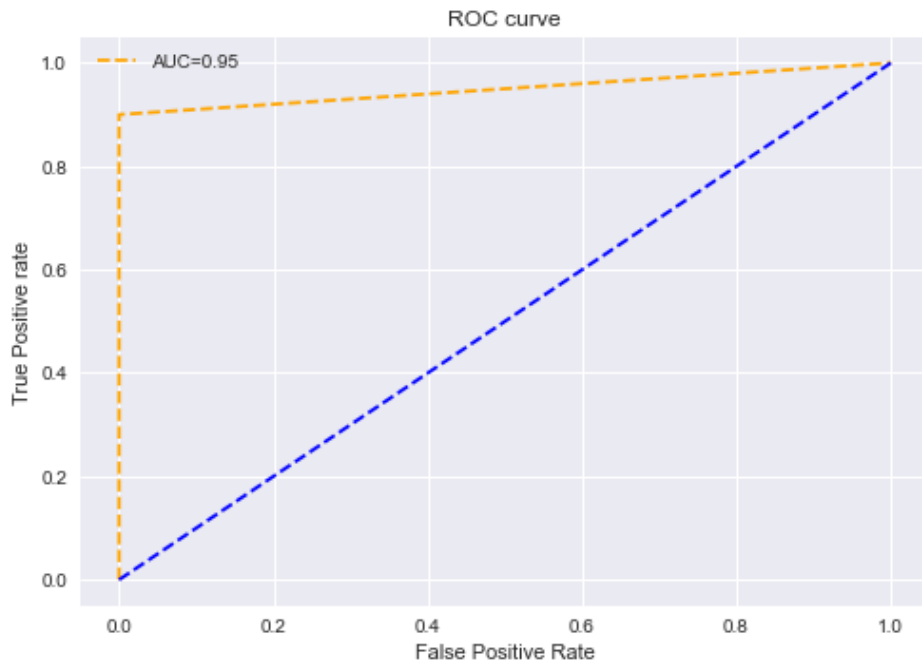


Figure 5.1.4 AUC-ROC Curve

The learning curve for the SVC model using the training and validation set were plotted and shown in Figure 5.1.5 below to show the relationship between training score and test score and evaluate the underfitting or overfitting of the model. Underfitting happens when the model fails to sufficiently learn the problem and performs poorly on a training dataset and does not perform well on testing data. Overfitting happens when a model learns the training dataset too well, performing well on the training dataset but does not perform well on testing data. A good fit model will suitably learn the training dataset and generalizes well to the testing data.

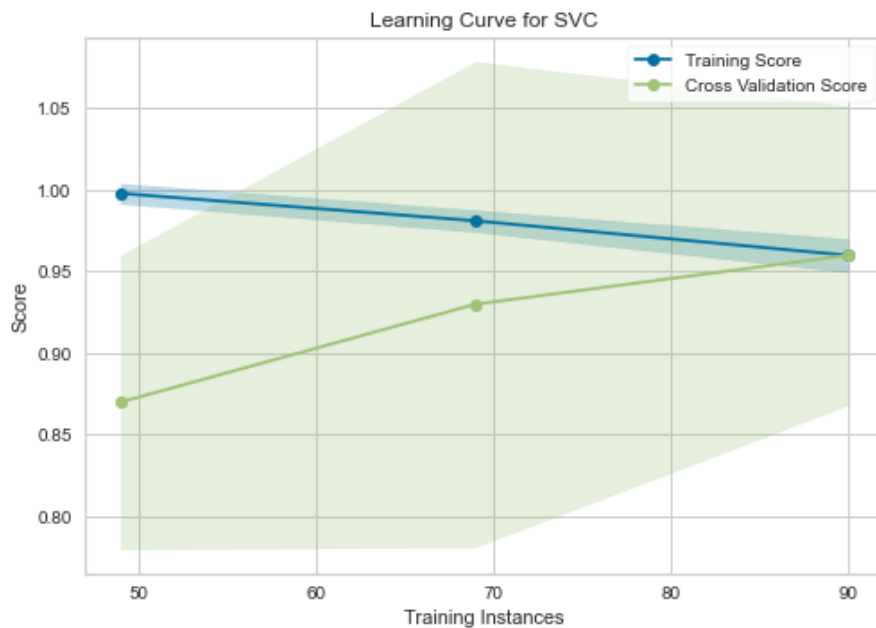


Figure 5.1.5 Learning Curve

In Figure 5.1.5 above, the training set's accuracy score is labelled "Training Score," whereas the testing set's accuracy score is labelled "Cross-Validation Score." The SVC model's training score is substantially higher than the test score until 90 training instances. If the current dataset has fewer than 90 training examples, increasing the number of training instances will improve generalization. However, after 90 training instances, the model is unlikely to benefit significantly from adding more training data. The graph plotted showed that our trained model has a good fit since the gap between the accuracy score for training and testing set are not that big.

5.2 Comparison of System Performance

The experimental results of the implemented system was compared with another similar approach for violence detection. The compared approach was a system employing optical flow approach to detect violence in sensitive locations proposed by Biswas and others. [12] In their system, the optical flow approach attained a 90% accuracy in their framework's experimental results for the Crowded dataset which contains 246 surveillance video clips. By comparison, our implemented system achieved an accuracy rate of 95% by applying optical flow approach for the UCF_Crimes dataset which contains 50 fighting surveillance video clips and 50 normal surveillance video clips. This indicated that our system's accuracy rate is much higher than their approach. Thus, our implemented system outperformed their optical flow approach in terms of accuracy score for fighting event detection.

5.3 Error Analysis & Future Work

From the confusion matrix in Figure 5.1.3 on previous section, we know that the fighting class has a misclassified result which the fighting case was predicted as normal case. Therefore, the error analysis will be focused on the fighting class type. From the implementation challenges mentioned before, one of the challenges is that some of the fighting events happening in the dataset videos might not be detected by the optical flow. It was caused by the reasons such as the human involved in the fighting events are very far away from the surveillance camera so their movement cannot be detected and traced. Another reason is that if there are obstacles that blocked the sight of the fighting events, the optical flow might not be able to detect the movement of the fighting events. The mentioned challenge was the reason for the error on misclassified result of fighting class. The following figures are some of the examples of the fighting event that could be misclassified.



Figure 5.3.1 Fighting Events which Humans Involved are Far Away



Figure 5.3.2 Fighting Events which Obstacles Blocked the Sight

The above figures contain 4 different dataset fighting videos which have the scenarios that might cause misclassification result. Figure 5.3.1 shows that when the human involved in the fighting events are far away from the surveillance camera, the good feature points might not be captured, thus optical flow cannot be generated. It was the same from Figure 5.3.2 which shows the scenario where there are obstacles blocking the sight of the fighting event happening.

Future work that can be done is to detect the connected regions of the optical flow. For each region, the histogram of the optical flow's strength with reference to the orientation value can be build up. Then, the histogram generated was compared between the normal actions and fighting actions. The histogram then can be summarised and used for classification process. The purpose of this is to reduce misclassification problems by only focusing on each of the region. The connected regions of the optical flow could be the normal actions such as people walking on the street, or it could be the abnormal actions such as fighting events. When there is a region that contains abnormal actions in the dataset videos, it can be classified as a fighting event happening in the video. When there is two group of people just walking on the street in different direction from the normal video, the optical flow of two group will not cross-intersect and cause the orientation value of optical flow to fluctuate by focusing on each connected regions of optical flow. Another future work to tackle the mentioned implementation challenge is to increase and improve the dataset quantity and quality. By improving the dataset quality, the videos such as video with low resolution, video with low frame per second, video with fighting event happened far away from the camera will be reduced and it can reduce the misclassification issue. By increasing the dataset quantity, it can help the model exposed to more features which can lead to lower estimation variance which is testing error and hence improving the prediction performance for the classification model.

5.4 Contributions

For the final delivery of this project, the program was able to classify between the fighting and normal dataset videos. The system has the capability of running in real-time and perform high accuracy of correct fighting event detection. The whole process of violence detection is automatic which means that human labour was not needed whereas the manual anomaly detection necessitates the use of human operators and is

labour-intensive, making it prone to mistakes and exhaustion. This also indicate that our system will help to alleviate the waste of labour and time and improve the community safety.

The system was able to process about 40 frames per seconds which is decent for automatic anomaly detection and also achieved our project's objective of 30 frames per seconds. Besides that, the system also achieves 0.95 in accuracy of detecting fighting events which make the system a trustable system to be used for automatic anomaly detection.

Other than detecting the fighting event using optical flow, the system also contributed on the observation for optical flow using various process. The observation processes included are using YOLO to compare the human size body, recolor the optical flow based on the orientation value, draw the Delaunay triangle and Voronoi diagram, and also generate the frequency histogram for the orientation value of optical flow. These observation processes contributed on convincing that optical flow is a suitable solution for fighting detection.

Chapter 6

Conclusion

6.1 Project Review, Discussions and Conclusions

In conclusion, recognising abnormal occurrences such as fighting is one of the most important tasks in video surveillance, but it is also one of the most difficult due to ambiguous nature of the anomaly and the complex nature of human behaviours. In comparison to normal activity, anomalous events are uncommon. This led to the project's problem statement, which is labour and time waste. Anomalous activities such as fighting, riots, and vandalism should all be detected automatically and in a timely manner. Thus, the motivation of this project is to detect a few categories of fighting events to timely signal such incidences as a warning. Aside from that, the project's innovation is its adaptation of automatic anomaly detection, which eliminates the need for manual anomaly detection and its contribution is the development of intelligent computer vision algorithms for automatic video anomaly detection, which will alleviate the waste of labour and time and improve community safety in our country.

For the solution, the input video frames will first be split into training and testing data and undergo pre-processing steps such as conversion to grayscale to reduce noise and dilation process to increase the white region. Then, the important features between two consecutive frames of input videos are then extracted using optical flow. The optical flow of the important features was calculated, and the tracks are drawn out using random colour lines. Next is the observation process to prove that the optical flow generated was meaningful and it was suitable for the project solution. The observation processes included are using YOLO to compare the human size body, recolour the optical flow based on the orientation value, draw the Delaunay triangle and Voronoi diagram, and generate the frequency histogram for the orientation value of optical flow. After observation, the standard deviations of orientation of optical flows on all the dataset videos was recorded and normalized. The normalized data was used to fit and train the SVM classification model. Last step is to perform classification to detect the fighting events on dataset videos. The trained model was also evaluated using confusion matrix, classification report, AUC-ROC curve, and the learning curve.

The implemented system has achieved a 95% accuracy score and process about 40 frames per second in classifying the fighting events which fulfil the system performance definition which was the system should achieve equal to or higher than 90% in terms of accuracy and 30 frames per second. The implemented system also achieved the project's main objectives by detecting a few categories of fighting events which was categorized as fighting video with interesting pattern, fighting video with non-interesting pattern and normal video. The implemented system also achieved the project's sub objectives by achieving real time processing in object detection in a high speed and accuracy. The implementation challenges encountered are the high demand of memory and high consumption of time of the FCN model which caused the project's solution to be changed to optical flow approach under time restriction condition. Some of the fighting events happening in the dataset videos might not be detected by the optical flow is also one of implementation challenges. It was caused by reasons such as human involved in the fighting events are very far away from the camera and there are obstacles that blocked the sight of the fighting events which cause the optical flow could not be generated. From this project, a lot of knowledge on the image processing and computer vision field about the various techniques for recognition and classification process have been grasped. Besides than technical aspects, time management and communication skills was also improved throughout this project.

6.2 Novelties

The novelty of this project is that we proposed a system that included various of observation processes to prove that the optical flow was a viable solution for fighting event detection. Compared to past studies which only focuses on extracting the important features using optical flow and using it to perform classification, this project focuses on using optical flow to extract the features, proving that optical flow approach was suitable for fighting event detection and using it to perform classification. The observation processes which help to prove that the optical flow was a suitable solution for fighting event detection are using YOLO to compare the human size body, recolour the optical flow based on the orientation value, draw the Delaunay triangle and Voronoi diagram, and generate the frequency histogram for the orientation value of optical flow. The fighting events detected also have been categorized as fighting video with

interesting pattern, fighting video with non-interesting pattern and normal video in this project.

6.3 Contributions

The system was able to distinguish between the fighting and typical dataset movies for the project's final delivery. The system is capable of running in real-time and detecting accurate fighting events with high accuracy. The entire process of detecting violence is automated, which eliminates the need for human labour, whereas manual anomaly detection requires the use of human operators and is labor-intensive, making it prone to errors and exhaustion. This also means that our system will help to reduce labour and time waste while also improving community safety.

The system was able to process around 40 frames per second, which is adequate for automatic anomaly identification and also met our project's 30-frame-per-second goal. Aside from that, the system detects fighting events with an accuracy of 0.95, making it a reliable method to employ for automatic anomaly identification. The system also assisted in optical flow observation utilising various processes. Using YOLO to compare the human size body, recoloring the optical flow based on the orientation value, drawing the Delanuy triangle and Voronoi diagram, and generating the frequency histogram for the orientation value of optical flow are among the observation methods mentioned. These observation procedures contributed on convincing that optical flow is a viable solution for detecting fighting events.

6.4 Future Work

The connected regions of the optical flow can be detected in future work. The strength of the optical flow can be used to build up histogram for each region with reference to the orientation value. The resultant histogram was then compared to normal actions and fighting actions. After that, the histogram can be summed up and used in the classification process. The goal is to eliminate misclassification issues by concentrating on each region separately.

Increasing and improving the dataset quantity and quality is another future work to address the identified implementation difficulty. Videos with low resolution, low frame per second, and video with fighting events that occurred far away from the camera will be minimised as the dataset quality improves, and the misclassification issue will be

CHAPTER 6 CONCLUSION

addressed. By increasing the dataset size, the model will be exposed to more features, which will result in decreased estimation variance, which is testing error, and hence improved classification model prediction accuracy.

BIBLIOGRAPHY

- [1] W. Sultani, C. Chen, and M. Shah, “Real-World Anomaly Detection in Surveillance Videos,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 6479–6488, 2018, doi: 10.1109/CVPR.2018.00678.
- [2] R. Arroyo, J. J. Yebes, L. M. Bergasa, I. G. Daza, and J. Almazán, “Expert video-surveillance system for real-time detection of suspicious behaviors in shopping malls,” *Expert Syst. Appl.*, vol. 42, no. 21, pp. 7991–8005, 2015, doi: 10.1016/j.eswa.2015.06.016.
- [3] N. O’Mahony *et al.*, “Deep Learning vs. Traditional Computer Vision,” *Adv. Intell. Syst. Comput.*, vol. 943, no. Cv, pp. 128–144, 2020, doi: 10.1007/978-3-030-17795-9_10.
- [4] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, “Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach,” *Procedia Comput. Sci.*, vol. 132, pp. 679–688, 2018, doi: 10.1016/j.procs.2018.05.069.
- [5] M. Sabokrou, M. Fayyaz, M. Fathy, Z. Moayed, and R. Klette, “Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes,” *Comput. Vis. Image Underst.*, vol. 172, pp. 88–97, 2018, doi: 10.1016/j.cviu.2018.02.006.
- [6] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, “Social LSTM: Human trajectory prediction in crowded spaces,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 961–971, 2016, doi: 10.1109/CVPR.2016.110.
- [7] D. Weinland, R. Ronfard, and E. Boyer, “A survey of vision-based methods for action representation, segmentation and recognition,” *Comput. Vis. Image Underst.*, vol. 115, no. 2, pp. 224–241, 2011, doi: 10.1016/j.cviu.2010.10.002.
- [8] B. R. Kiran, D. M. Thomas, and R. Parakkal, “An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos,” *J. Imaging*, vol. 4, no. 2, 2018, doi: 10.3390/jimaging4020036.
- [9] Chalapathy, R. and Chawla, S., “Deep Learning for Anomaly Detection: A Survey,” *arXiv*, no. January, 2019.
- [10] Y. Zhu and S. Newsam, “Motion-aware feature for improved video anomaly

BIBLIOGRAPHY

- detection,” *30th Br. Mach. Vis. Conf. 2019, BMVC 2019*, 2020.
- [11] L. Xu, C. Gong, J. Yang, Q. Wu, and L. Yao, “VIOLENT VIDEO DETECTION BASED ON MoSIFT FEATURE AND SPARSE CODING Institution of Image Processing and Pattern Recognition , Shanghai Jiao Tong University , China School of Computing and Communications , University of Technology , Sydney , Australia,” *2014 IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 3562–3566, 2014.
- [12] M. Biswas *et al.*, “State-of-the-Art Violence Detection Techniques: A review,” *Asian J. Res. Comput. Sci.*, vol. 13, no. 1, pp. 29–42, 2022, doi: 10.9734/ajrcos/2022/v13i130305.
- [13] E. Bermejo Nievas, O. Deniz Suarez, G. Bueno García, and R. Sukthankar, “Violence detection in video using computer vision techniques,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6855 LNCS, no. PART 2, pp. 332–339, 2011, doi: 10.1007/978-3-642-23678-5_39.
- [14] N. Ikizler and P. Duygulu, “Histogram of oriented rectangles: A new pose descriptor for human action recognition,” *Image Vis. Comput.*, vol. 27, no. 10, pp. 1515–1526, 2009, doi: 10.1016/j.imavis.2009.02.002.
- [15] J. J. P. Suarez and P. C. Naval, “A survey on deep learning techniques for video anomaly detection,” *arXiv*, no. September, 2020.
- [16] D. Ryan, S. Denman, C. Fookes, and S. Sridharan, “Textures of optical flow for real-time anomaly detection in crowds,” *2011 8th IEEE Int. Conf. Adv. Video Signal Based Surveillance, AVSS 2011*, pp. 230–235, 2011, doi: 10.1109/AVSS.2011.6027327.
- [17] F. Wang, X. Yang, Y. Zhang, and J. Yuan, “Ship Target Detection Algorithm Based on Improved YOLOv3,” *ACM Int. Conf. Proceeding Ser.*, pp. 162–166, 2020, doi: 10.1145/3422713.3422721.

WEEKLY LOG

FINAL YEAR PROJECT WEEKLY REPORT

(Project I / Project II)

Trimester, Year: Y3S3	Study week no.:1
Student Name & ID: FOO WEN SHUN 18ACB03066	
Supervisor: Prof. Dr Leung Kar Hang	
Project Title: Fighting video analysis employing computer vision technique	

1. WORK DONE

Revise back Project I report about the work that have been done.

2. WORK TO BE DONE

Start working on the report and change the Chapter 3: Proposed Methods content to optical flow.

- Need you to show / send me the running results on all the video of the fighting scene.
- Need to see which show interesting pattern that can be detected as fighting; also need to see which show non (or no) interesting pattern to distinguish them. You can then classify the scene into 2 classes. Concentrate on those you can handle first. If have time, come back to tackle the other class.

3. PROBLEMS ENCOUNTERED**4. SELF EVALUATION OF THE PROGRESS**


 Supervisor's signature



 Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project I / Project II)

Trimester, Year: Y3S3	Study week no.:3
Student Name & ID: FOO WEN SHUN 18ACB03066	
Supervisor: Prof. Dr Leung Kar Hang	
Project Title: Fighting video analysis employing computer vision technique	

1. WORK DONE

Run the optical flow program on all the fighting videos. Also classified the scenes into two classes which are interesting pattern and non-interesting pattern between them.

2. WORK TO BE DONE

Concentrate on the class that have interesting pattern first.

3. PROBLEMS ENCOUNTERED

4. SELF EVALUATION OF THE PROGRESS



Supervisor's signature

10 Feb 2022



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT*(Project I / Project II)*

Trimester, Year: Y3S3	Study week no.:4
Student Name & ID: FOO WEN SHUN 18ACB03066	
Supervisor: Prof. Dr Leung Kar Hang	
Project Title: Fighting video analysis employing computer vision technique	

1. WORK DONE

Have reverse the color of the dilation window by using thresholding binary inverse.

Seem incorrect. I expect to see 2 colors, black and white but cannot find the black color.

2. WORK TO BE DONE

Have to combine the optical flow generated with the reversed color window and add in the color into the optical flow generated using HSV based on the direction of the movement.

Extract the code to draw the white optical flow and then draw them on the last window. After you can do it, you should detect the orientation of the optical flows and redraw them using color as discussed in last week.

3. PROBLEMS ENCOUNTERED**4. SELF EVALUATION OF THE PROGRESS**

 Supervisor's signature

17 Feb 2022

 Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project I / Project II)

Trimester, Year: Y3S3	Study week no.:5
Student Name & ID: FOO WEN SHUN 18ACB03066	
Supervisor: Prof. Dr Leung Kar Hang	
Project Title: Fighting video analysis employing computer vision technique	

1. WORK DONE

Enlarged the window, overlay the optical flow into the reversed color window. Tried putting color into the optical flow. Figuring out how to detect the orientation of optical flow.

2. WORK TO BE DONE

Have to detect the orientation of optical flows and draw the optical flow based on the orientation.

3. PROBLEMS ENCOUNTERED

4. SELF EVALUATION OF THE PROGRESS



Supervisor's signature

23 Feb 2022



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project I / Project II)

Trimester, Year: Y3S3	Study week no.:6
Student Name & ID: FOO WEN SHUN 18ACB03066	
Supervisor: Prof. Dr Leung Kar Hang	
Project Title: Fighting video analysis employing computer vision technique	

<p>1. WORK DONE</p> <p>Detected the orientation of optical flow. Tried to use the angle calculated of optical flow to build the hsv image but does not work.</p> <p>Not sure of your meaning on "build the hsv image"</p>
<p>2. WORK TO BE DONE</p> <p>Have to redraw the optical flow using color based on the orientation of optical flow. If the current solution does not work, have to find a new way to do it.</p> <p>Have sent you suggestion on how to overcome it. Check with me again if you don't understand my suggestion.</p>
<p>3. PROBLEMS ENCOUNTERED</p>
<p>4. SELF EVALUATION OF THE PROGRESS</p>

Supervisor's signature

2 Mar 2022

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project I / Project II)

Trimester, Year: Y3S3	Study week no.:7
Student Name & ID: FOO WEN SHUN 18ACB03066	
Supervisor: Prof. Dr Leung Kar Hang	
Project Title: Fighting video analysis employing computer vision technique	

1. WORK DONE Created a lookup table for hue value from 0 to 179.
2. WORK TO BE DONE Have to transform the calculated orientation to rgb color using the lookup table created.
3. PROBLEMS ENCOUNTERED
4. SELF EVALUATION OF THE PROGRESS



Supervisor's signature

10 Mar 2022



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT*(Project I / Project II)*

Trimester, Year: Y3S3	Study week no.:9
Student Name & ID: FOO WEN SHUN 18ACB03066	
Supervisor: Prof. Dr Leung Kar Hang	
Project Title: Fighting video analysis employing computer vision technique	

1. WORK DONE

Convert the optical flow colors based on the orientation of the optical flow. Have ran the program on many videos and sort it based on interesting and non-interesting pattern. Also ran the program on normal videos to show the difference of optical flow colors between normal videos and fighting videos. Modified the report.

2. WORK TO BE DONE

Have to perform classification based on the optical flow generated.

3. PROBLEMS ENCOUNTERED

1. detect connected regions. 2 For each region, build up the histogram of flow strength wrt orientation. 3. Compare pattern of normal action against abnormal ones.

4. SELF EVALUATION OF THE PROGRESS


 Supervisor's signature

24 Mar 2022



 Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project I / Project II)

Trimester, Year: Y3S3	Study week no.:10
Student Name & ID: FOO WEN SHUN 18ACB03066	
Supervisor: Prof. Dr Leung Kar Hang	
Project Title: Fighting video analysis employing computer vision technique	

1. WORK DONE Draw Delaunay Triangle and Voronoi Diagram on fighting and normal videos.
2. WORK TO BE DONE Have to perform classification based on the optical flow generated.
3. PROBLEMS ENCOUNTERED
4. SELF EVALUATION OF THE PROGRESS



Supervisor's signature

31 Mar 2022



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT*(Project I / Project II)*

Trimester, Year: Y3S3	Study week no.:11
Student Name & ID: FOO WEN SHUN 18ACB03066	
Supervisor: Prof. Dr Leung Kar Hang	
Project Title: Fighting video analysis employing computer vision technique	

1. WORK DONE

Generated frequency histogram of orientation of optical flow for normal and fighting videos. Performed classification using the standard deviation of orientation of optical flow by SVM. Evaluated the performance of the classification result using confusion matrix and AUC-ROC curve and learning curve.

2. WORK TO BE DONE

Complete the report. *Need detail of the generated histogram like the labels on the x and y axis and how it is generated.*

3. PROBLEMS ENCOUNTERED**4. SELF EVALUATION OF THE PROGRESS**

Supervisor's signature

6 Apr 2022

Student's signature

POSTER



FIGHTING VIDEO ANALYSIS EMPLOYING COMPUTER VISION TECHNIQUE

Foo Wen Shun
Supervisor: Prof. Leung Kar Hang

INTRODUCTION
 Detecting anomaly event is one of the most important tasks in video surveillance. Anomalous events usually rarely occur as compared to normal activities which leads to the waste of labour and time. Thus, there is an acute need for such a software to detect a few categories of fighting events to timely signal such incidences as a warning.

METHODOLOGY & DISCUSSION

```

    graph TD
      A[Input Video Frame] --> B[Training Data]
      A --> C[Testing Data]
      B --> D[Pre-Processing]
      C --> D
      D --> E[Feature Extraction using Optical Flow]
      E --> F[Observation Process]
      F --> G[SVM Model Training]
      G --> H[Classification]
      H --> I[Model Evaluation]
      I --> J[Classified Fighting Events]
    
```

First, the input video frames will be split into training and testing data and undergo pre-processing steps such as conversion to grayscale and dilation. Then, the important features are extracted using optical flow. Next is the observation process on optical flow generated. Data of optical flows was then used to fit and train the SVM model and perform classification. The trained model was also evaluated using confusion matrix, classification report, AUC-ROC curve, and the learning curve.

CONCLUSION
 To summarize, the implemented system was able to detect a few categories of fighting events by achieving real time processing in a high speed and accuracy. This project contributed to prove that optical flow approach was suitable for fighting event detection using various observation process. In the end, this project help to eliminates the need for manual anomaly detection, alleviate the waste of labour and time and improve community safety in our country.



PLAGIARISM CHECK RESULT

Fighting video analysis employing computer vision technique

ORIGINALITY REPORT

14% SIMILARITY INDEX	7% INTERNET SOURCES	10% PUBLICATIONS	4% STUDENT PAPERS
--------------------------------	-------------------------------	----------------------------	-----------------------------

PRIMARY SOURCES

1	deepai.org Internet Source	1%
2	arxiv.org Internet Source	1%
3	Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, Silvio Savarese. "Social LSTM: Human Trajectory Prediction in Crowded Spaces", 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016 Publication	1%
4	Alexandre Alahi, Vignesh Ramanathan, Kratarth Goel, Alexandre Robicquet, Amir A. Sadeghian, Li Fei-Fei, Silvio Savarese. "Learning to Predict Human Behavior in Crowded Scenes", Elsevier BV, 2017 Publication	1%
5	Submitted to University of Bedfordshire Student Paper	1%
6	Submitted to Universiti Tunku Abdul Rahman Student Paper	<1%

PLAGIARISM CHECK RESULT

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1 of 1



**FACULTY OF INFORMATION AND COMMUNICATION
TECHNOLOGY**

Full Name(s) of Candidate(s)	FOO WEN SHUN
ID Number(s)	1803066
Programme / Course	CS
Title of Final Year Project	Fighting video analysis employing computer vision technique

Similarity	Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)
Overall similarity index: <u>14</u> % Similarity by source Internet Sources: <u>7</u> % Publications: <u>10</u> % Student Papers: <u>4</u> %	
Number of individual sources listed of more than 3% similarity: <u>0</u>	
Parameters of originality required and limits approved by UTAR are as Follows: (i) Overall similarity index is 20% and below, and (ii) Matching of individual sources listed must be less than 3% each, and (iii) Matching texts in continuous block must not exceed 8 words <i>Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.</i>	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor

Name: Leung Kar Hang

Date: 20 April 2022

Signature of Co-Supervisor

Name: _____

Date: _____



UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	18ACB03066
Student Name	Foo Wen Shun
Supervisor Name	Prof. Leung Kar Hang

TICK (✓)	DOCUMENT ITEMS
	Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Plastic Cover (for hardcopy)
✓	Title Page
✓	Signed Report Status Declaration Form
✓	Signed FYP Thesis Submission Form
✓	Signed form of the Declaration of Originality
✓	Acknowledgement
✓	Abstract
✓	Table of Contents
✓	List of Figures (if applicable)
✓	List of Tables (if applicable)
	List of Symbols (if applicable)
✓	List of Abbreviations (if applicable)
✓	Chapters / Content
✓	Bibliography (or References)
✓	All references in bibliography are cited in the thesis, especially in the chapter of literature review
	Appendices (if applicable)
✓	Weekly Log
✓	Poster
✓	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
✓	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

(Signature of Student)

Date: 20/04/2022