BSCS-F22-007

03-134192-067 SYED MUAHMMAD RAZA ALI

03-134192-042 USAMA SAEED

ANOMALY DETECTION SMART CAMERA

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

Supervisor: Junaid Nasir

Department of Computer Sciences

Bahria University, Lahore Campus

June 2023

Certificate

We accept the work contained in the report titled

"**ANOMALY DETECTION SMART CAMERA**"

Written by

SYED MUAHMMAD RAZA ALI

USAMA SAEED

As a confirmation to the required standard for the partial fulfilment of the degree of Bachelor of Science in Computer Science.

Approved by:

Supervisor: Junaid Nasir

____(Signature)

June 20, 2023

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name	Signature
03-134192-067	Syed Muhammad Raza Ali	
03-134192-042	Usama Saeed	

Date : June 20, 2023

Specially dedicated to

my beloved grandmother, mother and father

Syed Muhammad Raza Ali

my beloved grandmother, mother and father

Usama Saeed

ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this project. We would like to express our gratitude to my research supervisor, Mr Junaid Nasir for his invaluable advice, guidance and his enormous patience throughout the development of the research.

In addition, we would also like to express my gratitude to our loving parent and friends who had helped and given me encouragement.

Syed Muhammad Raza Ali
Usama Saeed

ANOMALY DETECTION SMART CAMERA

ABSTRACT

This report introduces a technique to identify default camera events using image analysis. The key feature of our project is to ensure good image quality and to provide appropriate platform for monitoring surveillance videos. The approach of our project is to remove the reduced referenced features in most regions of the surveillance image and then to detect anomaly related scenarios by studying the variation of features when the viewing field changes. Real-time alerts for video surveillance systems have made anomaly camera detection increasingly important. However, existing methods are inadequate in detecting various abnormalities and are incapable of self-study to improve their performance in case of failures. This report proposes Anomaly camera detection method that uses morphological analysis and in-depth reading to detect a wide range of anomalies. Morphological analysis is used for detecting simple anomaly cameras detection to improve processing speed, while in-depth reading is used for identifying complex anomaly camera distractions to enhance accuracy. The proposed technique has been tested and proven to have an accuracy rate of over 95% results are further elaborated in the report. Our project starts with the previous process of video capture with limited video, inverting, sliding, sorting, resizing, and extracting features are also done in this process. Next, a network feed process is requested to generate an output matrix. Based on the output matrix, the known Face, object or character can be determined. This project is designed to customize the network to each user. Recommendations for future development and conclusions are also included in the report.

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

FR	Full Reference
NR	No Reference
RR	Reduce Reference

CHAPTER 1

INTRODUCTION

1.1 Background

When it comes to anomaly camera detection, a reference image is typically the initial image that is described by a person. However, in the picture quality testing, there are three distinct methods for drawing and analysing images, each with their own assumptions about image quality. These methods are full reference (FR), no-reference (NR), and reduced reference (RR). The main difference between these methods is the size of the reference data. FR metrics are based on the size of the reference image and involve pixel-based comparisons between the reference and target images. On the other hand, NR metrics only use the current frame and do not require any reference data, resulting low accuracy in this process. The RR metric falls somewhere in between, with reference data that is smaller than the size of the inserted image.

Most camera detection methods use the FR method, relying on background images and mathematical models as reference frames to eliminate variations in sound and crowded scenes. These methods typically involve calculating the relationship between the background image model and the nonparametric kernel density method for features such as Ku, entropy, edges, and standard cross. The degree of change in these features determines the image decision results, including closure, image stabilization, and FOV transformation. Other methods, such as those used by Aksay, apply wavelet domain techniques to background images in order to detect destruction or interference with the camera view. Meanwhile, Saglam and Temisel [1]have developed a dynamically updated framework for background images, which uses region-based feature output to detect camera interference. While these methods are successful in modelling the background, they often fail to account for real-world challenges such as heavy movement/crowds and lighting problems.

An unconventional camera detection method has been proposed that deploys morphological analysis and in-depth reading to detect a wide range of anomalies. This method has been tested and proven to have an accuracy rate of over 95%. Unlike traditional methods, it does not rely on background modelling and mathematical models as reference frames, making it more effective in dealing with real-life challenges.

1.2 Problem Statements

Anomaly detection is an important problem that has been researched in various areas of research and application domains. Many anomaly detection methods are specially designed in some application domains. This study seeks to provide a comprehensive and complete overview of Anomaly detection related research. We've collected the existing strategies that are divided into different categories based on the basic approach adopted by each technology. In each section, we identified the most important, strategic ways to distinguish between normal and unusual behaviour. If you use the method given in in a particular context, these ideas can be used as guidelines for evaluating the effectiveness of technology in that domain. At each stage, we provide a basic way to find the Anomaly, as well then show that the different strategies available in that category are a variety of basic approaches. This template provides a simple and concise understanding of existing strategies in each section. In addition, in each section, we will consider the pros and cons of strategies in that category. We also provide a discussion on the complexity of computer strategies as it is an important issue in the real-world application domains. We hope this research will provide a better understanding of the various guidelines in which the research has been conducted this article, as well as how strategies developed in one place can be applied to the domains it was not intended to begin.

In this research, algorithms for detecting incomprehensible video surveillance feeds are developed. and tested. In 2013, another research was written in FOI by Viktor Edman when the crowd-tracking algorithm, using the Gaussian probability hypothesis (GM-PHD) filter, was developed [1]. The current thesis will use the output in this algorithm as inputs to various machine learning models to analyse the surrounding. Unlike other machine learning fields, such as speech recognition, there is still no best solution for any of the features that can be used for human visual perception, including anomaly detection, and how compatible models should be designed. This study will use a variety of methods and techniques to determine one's individual strengths and weaknesses and what parameter settings are we agreeing on. In addition, this study will also try to determine where some of these algorithms can be used in conjunction with a large number of non-labelled data. No other previous study or project has performed the same experiments where an anomaly discovery algorithm is used in a group of people where local relationships of groups have been used.

1.3 Aims and Objectives

The purpose of this thesis is to improve the safety and efficiency of surveillance high-security institutions by creating a model that uses the output from the filter for the extraction of information about the work at the scene.

- To make it easy to recognize any human who is registered in restricted vicinity.
- To have secure and proper surveillance of everyone 24/7.
- To catch and record any intruder who's trying to invade the boundary.
- To catch and record any person who carries a weapon with him/her near a restricted vicinity area.
- To catch and record who would've threat for restricted vicinity area.
- To catch and record any fire and emergency near restricted vicinity areas.
- To catch and record any mob of people which would threaten for restricted vicinity area.

To tackle all of these situations and many more which could cause harm to a highly restricted area we are designing this anomaly detection. It would add on any security for any restricted area. It is modernizing way of checking all of the actions which would have been a threat to these kinds of areas. Without using manpower on walls of the restricted area we can have an eye on every activity which is just going outside of the wall of the restricted area.

1.4 Scope of Project

Disruptions occur due to changes and can cause major upsets that affect your business. But if you can see changes occurring in your area near real-time, you can prevent these as well as take precautions. In today's digital business environment where most of the business is working with applications, responding to disruptions is no longer an option.

Take a bank, for example, that does not know what all the dangerous events will look like. In this case, it is not possible for a bank to foresee all cases, draft rules to identify unidentified data, or even develop statistical models to prevent it. Only a machine learning system that adapts to continuous change can protect you from future unknown IT problems.

Logic Monitor has provided Anomaly Detection of metrics for a long time already, but so far, we have not yet logged. We will be making Anomaly Detection logs available to all our customers in the form of our new product, LM Logs. Our basic algorithmic method of logs will make it easier for you and your teams to filter out signals from sound, and resolve problems faster than before, LM logs are coming soon.

Anomaly detection in high-dimensional data is becoming a fundamental research area that has various applications in the real world. As such, a large body of research has been devoted towards addressing this problem. Nevertheless, most existing surveys focus on the individual aspects of anomaly detection or high dimensionality. For example, Agrawal and Agrawal provide a review of various anomaly detection techniques, with the aim of presenting a basic insight into various approaches for anomaly detection. present several real-world applications of graph-based anomaly detection and concluded with open challenges in the field. Chandola [2] present a survey of several anomaly detection techniques for various applications. Hodge and Austin present a survey of outlier detection techniques by comparing techniques' advantages and disadvantages. Patcha and Park have conducted a comprehensive survey of anomaly detection systems and hybrid intrusion detection systems by identifying open problems and challenges. Jiang [2]present a survey of advanced techniques in detecting suspicious behaviour; they also present detection scenarios for various real-world situations. Sorzano [2] categorize dimensionality reduction techniques, along with the underpinning mathematical insights. Various other surveys can also be observed, such as those by Gama Gupta [3], Heydari ,and Jindal and Liu [4] Pathasarathy, Phua [5] Tamboli, and Spirin which further highlight the problems either in anomaly detection or in high-dimensional data.

A limited amount of work has been done that emphasizes anomaly detection and high dimensionality problems together, either directly or indirectly. Zimek [6] present a detailed survey of specialized algorithms for anomaly detection in high-dimensional numerical data; they also highlight important aspects of the curse of dimensionality. Parsons [7] present a survey of the various subspace clustering algorithms for high-dimensional data and discuss some potential applications in which the algorithms can be used.

To the best of the authors’ knowledge, no surveys directly highlight the problems of anomaly detection and high dimensionality in big data. In this survey, we present an integrated overview of these two problems from the perspective of big data.

CHAPTER 2

LITERATURE REVIEW

2.1 BACKGROUND

In order to counter acts and threats of various societies in today's era, governments and some organizations focus on mass surveillance. In some cases, it is necessary to analyse the behaviour of people and vehicles to determine what happened at the scene. Specifically, it may be very important to determine if an event exists as normal or an anomaly. In part because of the low prices of hardware, conventional surveillance does nothing that seems to not work in this function as the number of cameras available to the operator, in general, exceeds the operator's ability to self-monitor.

In addition, the effectiveness of standard surveillance is limited by the ability of users. Monitoring surveillance cameras requires focusing on only what operators can hold for a short period, especially from a general point of view behaviour may be repetitive, dull, and tedious. This has led to information on surveillance cameras will be used extensively during event analysis instead as a deterrent, discovery tool. To improve safety and security performance systems, it is necessary to build an algorithm that can help operators to operate with high efficiency or even full-time employment. Such algorithms can do just that allowing the system to run and interact in real-time with the environment, instead of just like a forensic tool. The algorithm can also increase the chances of discovery as it can use a temporary pattern of people or crowds that can be difficult to distinguish with the human eye. However, for the algorithm to be reliable in the real world, it is required that it has a low false positive detection value and a low value of this strange mysterious.

In this thesis, anonymous event is defined as a low-level event with a common model of crowd behaviour. abnormality can be spread internally by one mob, e.g., fighting, or between two or more mobs, e.g., clashes between mobs or arguments during demonstrations. In this thesis, there is no speculation about the origin of anomaly will be done, the event will be considered unusual regardless of whether it is broadcast in or between the crowds.

Conflicting discovery models can be divided into two main groups: explicit detection and deviation methods. On the way to finding clarity, both normal and abnormal behaviour is imitated. The new visuals are then classified as alternatives to these categories. The advantage of the obvious finding is that the anomaly stage can be explained in great detail. However, it is not possible to model all types of abnormal behaviour when the scene is extremely dense or when information about these events is scarce. Deviation methods try to model the normal behaviour of an object or event and determine what is anomaly in different events in this model. One of the most common and popular models used in this method is Hidden Markov Model (HMM). This model is often adapted to suit a variety of conditions and, as a result, several different varieties have been created.

The field of anomaly detection and visual recognition can be divided into three main categories of methods, these are the elements of the framework, the manner of operation of man, and the performance of the crowd. Locations are defined in the way it is used in video feed analysis. The framework of the draft event uses information about the whole scene, usually without first finding people or crowds in it. In one's work How to do it, people are first seen at the scene. This information can be used to decide what a person does or does something unusual. The drawback of this method is that can be difficult to identify individuals and their actions when a person is fully or slightly hidden for various reasons or when many people are present at the scene. The last method, how to do a crowded event, uses information about a group of people as a single unit, which does not require information about individuals. This method can be helpful as it can be difficult to model Complex relationships of pedestrians within proximity to each other. Human activities and methods of mob action are sometimes called the microscopic method and macroscopic method, respectively.

2.2 RELATED WORK

Anomaly event detection details and the most recent research findings will be discussed in this chapter. Modern intelligent video surveillance systems rely heavily on computer vision analytics for automatic anomaly detection, which not only significantly improves monitoring efficiency but also alleviates the need for live surveillance. Peculiarities in recordings are comprehensively characterized as occasions or exercises that are uncommon and imply unpredictable way of behaving. Anomaly detection aims to locate the anomaly events in video sequences in time or space. Frame-level detection is the process of identifying the start and end frames of an anomaly event in a video using temporal localization. The more difficult aspect of spatial localization is determining which pixels in each anomaly frame correspond to the anomaly event. Pixel-level detection is typically the name given to this setting. In this paper, we offer a brief synopsis of the most recent developments in video anomaly detection research and highlight a few potential future research avenues. Anomaly detection is a process that involves identifying unusual occurrences, items, or observations that deviate significantly from the majority of the data. These anomalies are often linked to issues such as bank fraud, structural defects, medical problems, or errors in written text. They are sometimes referred to as outliers, novelties, noise, deviations, or exceptions. By detecting anomalies, businesses can take proactive measures to address potential issues before they escalate into larger problems.

In computer vision, anomaly event detection is one of the most challenging problems and has attracted lots of research efforts in the past decades. The hypothesis that anomalies are uncommon and that behaviours that seriously differ from normal patterns are regarded as anomalous is the primary focus of the first category of anomaly detection methods. Regular patterns are encoded using a variety of statistical models, including Gaussian process-based models, the social force model, Hidden Markov-based models, spatial-temporal Markov random field-based models, and dynamic model combinations. Anomalies are treated as outliers in these approaches.

Sparse reconstruction, which is used for normal pattern learning, is the second type of anomaly detection approach. Particularly, a dictionary is constructed by employing sparse representations of normal behaviour; anomalies are identified when they have high errors. As of late, with the promising forward leap of profound learning, a few scientists build profound brain networks for peculiarity recognition, including video expectation learning, and deliberation highlight learning.

The goal of anomaly detection is to find patterns, anomalies, or data points that don't follow the usual distribution. Uses of oddity discovery incorporate security reconnaissance, misrepresentation recognition in monetary exchanges, issue identification in assembling, interruption location in a PC organization, observing sensor readings in an airplane, spotting expected dangers or clinical issues in wellbeing information, and prescient support. Today, because of modest innovation, we have an enormous number of recordings for example reconnaissance recordings which are the fundamental source to catch continuous unusual exercises yet for programmed identification of various inconsistencies, we really want to plan a framework that accepted such observation recordings as crude info and afterward produce a valuable result.

Table 2.1: Processing Time (in seconds) of Board for Different Machines

Act	PC-I	PC-II	PC-III
Face	08	09	28
Gun	13	15	43

This is the time taken by all of the PCs in which we ran our projects one by one and there was no major difference between the two PCs which were highly upgraded and the one which was not very well upgraded took some time to do the work.

2.3 COMPARISON TABLE WITH EXISTING STUDY

Table 2.2: Comparison Table

Feature	Anomaly Detection Studied	Anomaly Detection made by us
Types of Attacks Detected	Gun	Gun, Mob, Fire
Attack Background Data required	Yes, the background data is required	No, the background data is not required
False Alarm Rate	Rate of False Alarm is high	False Alarm Rate is low
Need Update	Yes, the system needs updates	No, the system does not require updates
Attack type	Cannot be Defined	Cannot be Defined

Protecting Tool Identification	No, tool can be identified	Yes, tool can be detected

In this comparison table, we can see we are a more advanced and updated version of this anomaly detection smart camera, and with fewer updates and requirements easier to use.

2.4 CHAPTER SUMMARY

Surveillance is the monitoring of behaviour, activities, or information to influence, manage, or direct. The activity assessment process is complex and cannot be done at once but by performing the necessary breakdown, the task becomes easy. Unintentionally some researcher confuses the term of activity with the term of action, gesture, motion, and interaction but in reality, they can be part of activity but not equal in definition.

In order to guarantee the safety of the public, security surveillance is increasingly being used in public places like hospitals, intersections, shopping malls, and banks. However, the monitoring capabilities and law enforcement agencies are incompatible. Subsequently, the outcome is that there are clear imperfections in the utilization of observation cameras. Abnormality occasion identification in observation recordings is a significant exploration subject in PC vision, which has been broadly utilized in numerous protections related situations, including auto collision examination, violations or criminal operations reconnaissance, criminology examination, and brutality cautioning. Behaviour or appearance patterns that deviate from normal patterns are frequently referred to as anomalies due to the fact that anomalous events rarely occur in real life.

In current wise camera reconnaissance frameworks, programmed irregularity identification through PC vision examination assumes an urgent part which essentially increments checking proficiency as well as lessens the weight on live observing. Peculiarities in recordings are comprehensively characterized as occasions or exercises that are uncommon and imply unpredictable way of behaving. Anomaly detection aims to locate the anomaly events in video sequences in time or space.

CHAPTER 3

DESIGN AND METHODOLOGY

In this project, we attempted to create an integrated framework for reconnaissance security that gradually distinguishes the weapons and faces, and if the identification is positively true, it will warn/brief the security personnel on how to handle the situation by arriving at the scene of the incident via IP cameras. We offer a paradigm that gives a computer a visionary sense to recognize dangerous weapons as well as faces and can also inform a human administrator when a pistol or firearm or an unknown person is visible in the vicinity.

3.1 PROPOSED METHODOLOGY

To train machine learning models, the most significant and critical aspect of any application is to have a desired and appropriate dataset. As a result, we can manually gather several photos.

We gathered at least 50 photos for each weapon class. One of the greatest ways to collect photographs for building one's dataset is to use live capture. These photographs were then stored in a folder. Photographs must be saved in ".jpg" format; if the images have different extensions, it will be difficult to train with them and will result in errors. Alternatively, because the photographs are processed in batches, the sizes of all the images are changed to the same width and height before training.

Object detection is largely concerned with computer vision, which entails recognizing things in electronic images. Object identification is one of the fields that has profited greatly from recent advances in deep learning. In the output layer, we combine a CNN with a SoftMax and ReLU activation. We also add the first layers for data augmentation, standardization, and scaling. Convolutional neural network (CNN) technology will be used in this case.

- **Input level**

The input layer is the first layer of the CNN that receives the input image data. An input image is usually represented as a matrix of pixel values, each pixel representing the intensity or colour of a particular point in the image. The size of the input layer is determined by the size of the input image and the number of channels (such as RGB or grayscale).

- **Convolutional layer**

In convolutional layers, most feature extraction is done in CNN. Applies a set of learnable filters (also called kernels) to the input image. Each filter is a small weight matrix that is "swept" over the image and computes the dot product between the filter and the corresponding part of the input image. This creates a feature map that highlights specific patterns in the image. The size and number of filters used in the convolutional layers are hyper parameters that can be tuned during training.

- **ReLU Activation Function**

A ReLU layer applies a nonlinear activation function called a Rectified Linear Unit (ReLU) to the output of a convolutional layer. The ReLU function simply applies a threshold operation. If the input is greater than zero, the output will equal the input, Otherwise the output will be zero. This introduces non-linearity into the network, allowing it to learn more complex and abstract features.

- **Pooling layer**

Pooling layers are used to down sample the feature maps produced by the convolutional layers. This reduces the spatial dimensionality of the feature map while preserving the most important information. The most common pooling operation is max pooling, which takes the maximum value within a small window (e.g., 2x2) of the feature map. This helps reduce the computational complexity of the network and prevent overfitting. Fully Connected Layer: A fully connected layer is a traditional neural network layer that connects every neuron in the previous layer to every neuron in the next layer. Typically used for final classification tasks. The size of the fully connected layer is determined by the number of neurons and the number of classes in the output layer.

- **Output level**

The output layer is the last layer in the network that does the output prediction or classification. The number of neurons in the output layer corresponds to the number of classes in the problem being solved (such as binary or multiclass classification). The output of the output layer is usually passed through a SoftMax function that converts the raw output values to probabilities. The class with the highest probability is chosen as the final prediction or classification.

3.2 Methodology Approach

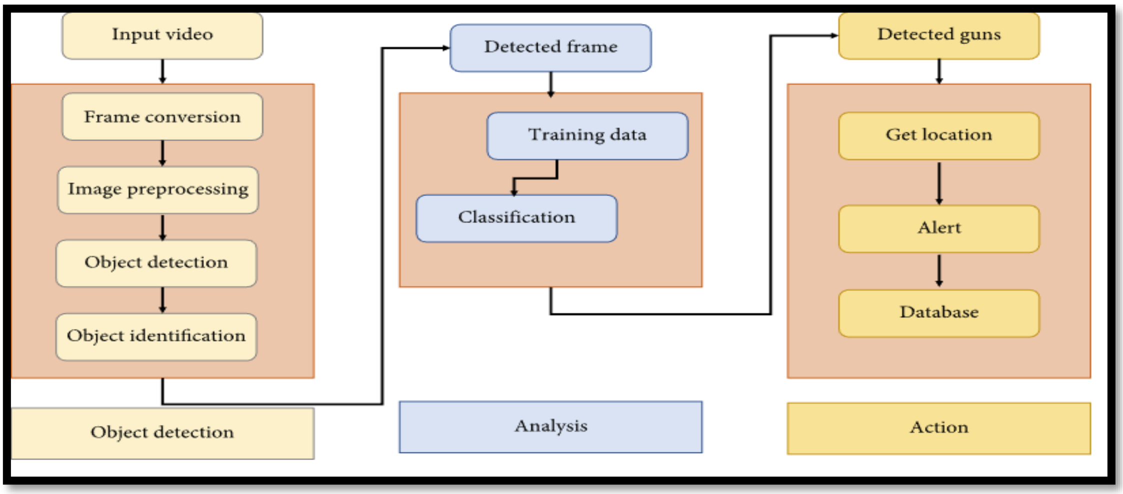


Figure 3.1: Project Diagram

Project Diagram Points:

- Starts from face and weapon detection, followed by pre-processing and then feature extraction finally classification
- Using extraction technique after pre-processing stage
- Which includes using detected faces and convert them to grey scale and save in another directory for further processing

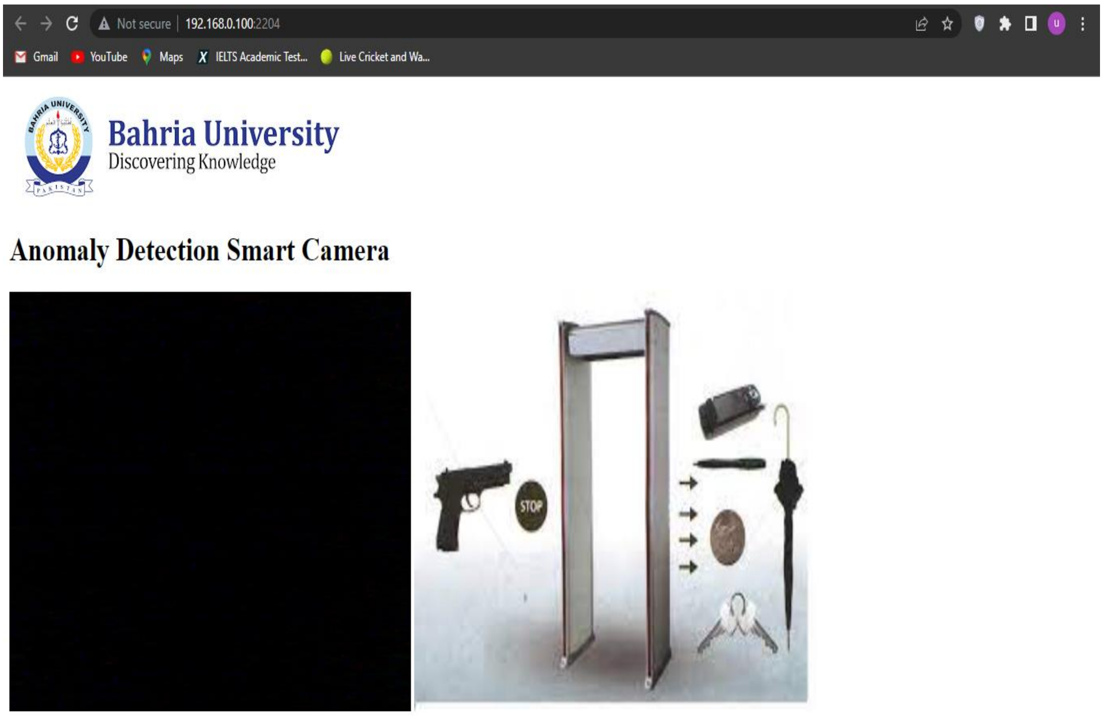
3.3 Interface

The following program creates a Flask web application that does face recognition and weapon detection in a live video stream from a webcam using computer vision and deep learning.

The program has been split down as follows:

- Importing the necessary libraries
- For building the web application, we use Flask.
- For rendering HTML templates, use render template.
- Streaming video response generation response.
- For computer vision tasks, use cv2.
- For loading a pre-trained deep learning model, use Keras load model function.
- warns against suppressing caution.
- For building a graphical user interface, thank you tkinter.
- For manipulating images, use Image, ImageTk, and ImageOps from PIL.
- For array operations, use numpy.
- for system-specific settings and operations, use sys.
- time for activities involving time.
- in order to communicate with the operating system.
- Loading the deep learning model and setting up the variables
- process_counter to keep track of the processing steps.
- warnings.filterwarnings("ignore") to suppress warnings.

Creating a Flask application:



Anomaly Detection Smart Camera

Figure 3.2 Anomaly Detection Interface

3.4 PROCESS MODEL

It's essentially an object detector that has been pre-trained. In a CNN model, a convolutional layer is liable for removing significant level qualities like edges from the info picture.

This produces activation maps or feature maps. These feature maps indicate the presence of detected features based on the input. As a result, the amount of pre-processing required is significantly reduced in comparison to previous classification methods. CNN is learned through a few emphases and preparing.

CNN architectures were initially rather linear. Numerous new variants have recently been introduced. These network models already have rich feature representations because they were trained on a lot of photos. The process of selecting a pre-trained network to use as a starting point for learning a new task is known as transfer learning. We utilized the assistance of a pre-prepared model to distinguish the weaponry.

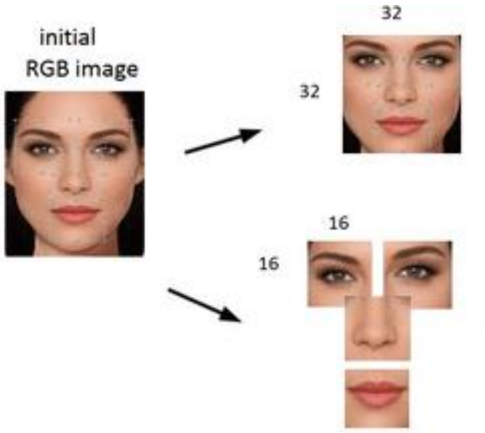


Figure 3.3: Pre-Prepared Model

3.4.1 Object Detector Method

Object detection is a computer vision task that involves identifying and localizing objects within an image or a video. It is a fundamental step in various applications, such as autonomous driving, surveillance systems, and image recognition. Local Binary Pattern (LBP) is a texture descriptor used in computer vision for representing local patterns in an image. It characterizes the texture of an image by encoding the relationships between a pixel and its neighbouring pixels. The LBP operator compares the intensity value of a central pixel with its surrounding pixels and encodes the result as a binary code. The Local Binary Pattern Histogram (LBPH) algorithm extends the LBP operator to perform object detection and recognition. It involves the following steps:

- **Image Segmentation:**

The input image is divided into small, overlapping regions called patches or cells.

- **LBP Calculation:**

For each pixel in a cell, the LBP operator is applied by comparing its intensity value with the neighbouring pixels. The result is a binary code that represents the local pattern around that pixel.

- **Histogram Calculation:**

After calculating the LBP codes for all pixels in a cell, a histogram is constructed by counting the occurrences of different LBP patterns within that cell. Each pattern is assigned a bin in the histogram.

- **Local Pattern Concatenation:**

The histograms from all cells are concatenated to form a feature vector that represents the entire image. This concatenation preserves the spatial information of the local patterns.

- **Training or Recognition:**

The LBPH algorithm can be used for both training and recognition phases. In the training phase, the LBPH model is trained using a set of labelled images. The feature vectors from the training images are used to create a model that can recognize different objects. In the recognition phase, the LBPH model is applied to new images to identify and localize objects based on their learned patterns.

LBPH has been widely used in various applications due to its simplicity and efficiency. It is robust to illumination changes and provides good performance for texture-based object detection and recognition tasks. However, it may struggle with complex object appearances or when dealing with large variations in scale, pose, or occlusion. In such cases, more advanced object detection algorithms like convolutional neural networks (CNNs) are often employed.

3.5 Train and Test Data

Dataset should be divided into 3 subsets, namely: Training, Dataset to be used while training. Validation, Dataset to be tested against training data. Test, Dataset to be tested against data after model is trained. The 70% of data will be used in the training set and in it most data will be used to train the given model and for that the model will get 70% of images from all classes and this will improve the accuracy and this model will use 10 % of data that will be used in testing and it will test data against the trained model or training set of data and also 10% of data will be used for validation and in it the actual data will be given and it will validate the data against test and trained data. The dividing portion will be changed for better accuracy. This function will be partitioning data as we want that from 25 it will get 20 of training data and 2 of validation data and 3 of test data.

Table 3.1 Training and Testing Data

Category	Label	Total	Train (70%)	Test (30%)
Gun	1	510	358	154
Fire	0	1405	983	421
Mob	2	631	442	189

3.6 Data Pre-processing

Data pre-processing is an important step in any data science project because it helps ensure that your data is clean, complete, and in a suitable format for analysis. This process typically involves several steps:

- **Data augmentation**

Data augmentation creates new variations of existing images by applying various transformations such as image rotation, mirroring, cropping, scaling, shearing, and adding noise. This technique helps increase the size and diversity of the training data set, which can improve model performance and reduce overfitting.

- **Data normalization**

Data normalization involves scaling the pixel values of an image to a common range (usually between 0 and 1). This ensures that all images are of similar scale, which is important for many machine learning algorithms.

- **Data resizing**

Resizing the data will change the resolution of the image to a common size. This is important to ensure that all images are the same size, which is required for many machine learning algorithms. Additionally, resizing reduces the amount of computation and helps speed up the training process.

- **Data labelling and breakdown**

Data tagging involves tagging images with relevant information, such as the presence or absence of certain diseases or traits. This is important for supervised learning, where the model is trained on labelled data. A data split splits the data set into a training set, a validation set, and a test set. The training set is used to train the model and the validation set is used to evaluate the model and optimize the hyper parameters during training. A test set is used to evaluate the final performance of the model using unseen data.

Taken together, these pre-processing techniques help ensure that image data is well formatted and ready for machine learning analysis. Using these techniques, you can increase the size and diversity of your training data set, standardize your data to improve model performance, and label and segment your data for supervised learning.

3.6.1 Data Augmentation

Data augmentation is a technique used in machine learning and computer vision to increase the size and variety of training datasets by creating new, modified versions of existing data. For images, data augmentation involves applying different transformations to images to create new variations that can be used to train machine learning models.

Common image enhancement techniques include:

- **Image rotation**

Rotates an image by a specified angle to create a new image with the same properties but a different orientation.

- **Image flipping**

Flip an image horizontally or vertically to create a mirror image of the original image.

- **Image cropping**

Select part of the original image and create a new image containing the selected part.

- **Image scaling**

Resize the image to create a larger or smaller version of the original image.

- **Image noise**

Add noise to your images to simulate different lighting and camera conditions. By applying these transformations to images, data augmentation can reduce overfitting and improve generalization of machine learning models. It also helps increase the size and diversity of the training dataset, which leads to improved model performance and accuracy.

3.7 MODULES DISCUSSION

Module-1: Starts with the detection of weapons and faces, moves on to pre-processing, feature extraction, and classification.

Module-2: Model-2 is similar to Model-1, but after the pre-processing stage, it also includes an additional feature extraction method. which includes utilizing faces that have been detected, converting them to grayscale, and saving them in a different directory for later processing.

Module-3: The first subprocess of automatic face recognition systems is face detection. We can't move on to higher levels of face recognition if we can't accurately recognize faces. In module 3, face detection can be carried out.

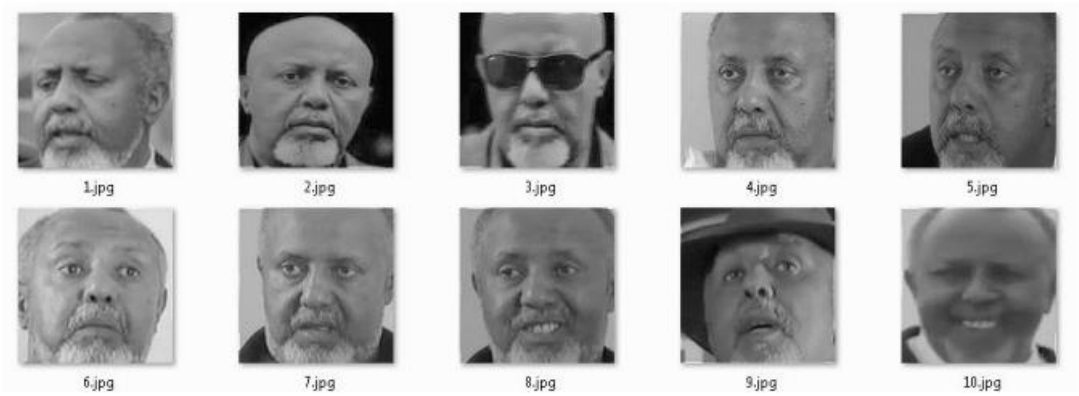


Figure 3.4 Module Description

3.8 Proposed Architecture:

The proposed architecture of the CNN model includes multiple layers. The input layer takes the raw image as input and applies a set of convolutional filters to extract features from the image. This is followed by a ReLU activation function that introduces non-linearity into the model. The output of the first set of convolutional and activation layers is then passed through a max pooling layer that reduces the spatial dimensions of the feature maps while preserving the important features. The process of convolution, activation, and pooling is repeated multiple times, creating deeper and more complex feature maps at each step. This is known as the convolutional block and helps the model to learn more intricate and abstract features from the input image. The final convolutional block is followed by a set of fully connected layers that act as a classifier, predicting the class of the input image. The output of the last fully connected layer is passed through a SoftMax activation function, which produces a probability distribution over the different classes. The proposed architecture can be further improved by adding more convolutional and fully connected layers, as well as applying regularization techniques like dropout to prevent overfitting.

3.8.1 Input layer

This is the first layer of the neural network and takes the raw image as input. The input layer is responsible for receiving image data and passing it to the next layer in the network for processing.

3.8.2 Convolutional layer

This layer applies a series of convolutional filters to the input image to extract features such as edges, shapes and textures. Each filter is a weight matrix applied to a small region of the input image, creating a feature map that emphasizes specific features in the image.

3.8.3 ReLU activation Function

This applies a Rectified Linear Unit (ReLU) activation function to the output of the previous convolutional layer. ReLU introduces nonlinearity into the model by setting all negative values to zero while leaving positive values unchanged.

3.8.4 Max pooling layer

This layer reduces the spatial dimensionality of the feature map while preserving important features. To do this, divide the feature map into non-overlapping regions and select the maximum value in each region. This reduces the computational requirements of the network and prevents overfitting.

3.8.5 DATABASE DESIGN (ERD)

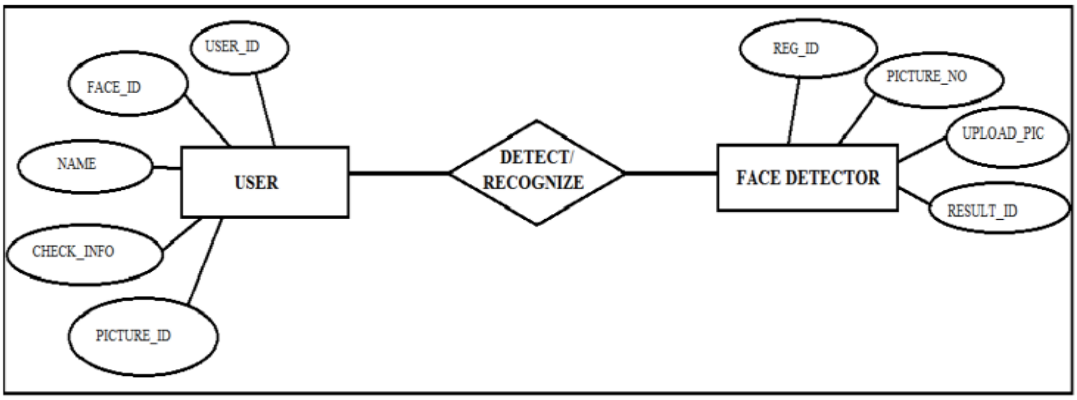


Figure 3.5: ERD Diagram

Left to Right in ERD

- The user will take an image using the smart camera.
- The user will initiate face detection in the real image.
- The user will trigger fire detection in the real time image.
- The user will activate gun detection in the real time image.

3.8.6 SEQUENCE

In the above cases, most of the methods mentioned earlier will not be enough to get the results you want. This is so because, with a camera mounted on a moving platform, usually there are complex, ever-changing domains. This is a violation common thought in many of the above ways, one of the smallest background changes that can be modelled as well as later extracted from videos. Moreover, in some cases, we are trying to see what caused by the presence of the front standing at the scene. This results in equality a large limit, as, without the addition of the previously recorded videos, one can no longer measure the background, thus tarnishing the image of many types of background/background classification methods.

3.8.7 CONTEXT

As shown in Fig. 1, mobile camera detection systems generally look for non-anomaly (as evidenced by a system operator) reference video and compare it with a video in search of situations where anomaly is present. Such a video comparison program is usually done by frame, thus requiring framework alignment and geometric alignment for both video sequences. Post-processing is usually done to apply some features of each program, as a temporary resemblance to a place of adoption

3.8.8 USE CASES

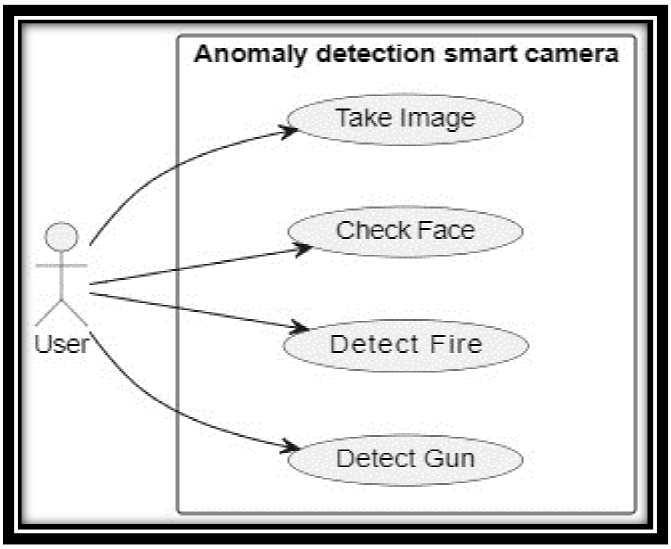


Figure 3.6: Use Case Diagram

Take Image: The user will take an image using the smart camera.

Check Face: The user will initiate face detection in the real image.

Detect Fire: The user will trigger fire detection in the real time image.

Detect Gun: The user will activate gun detection in the real time image.

3.8.9 Sequence Diagram:

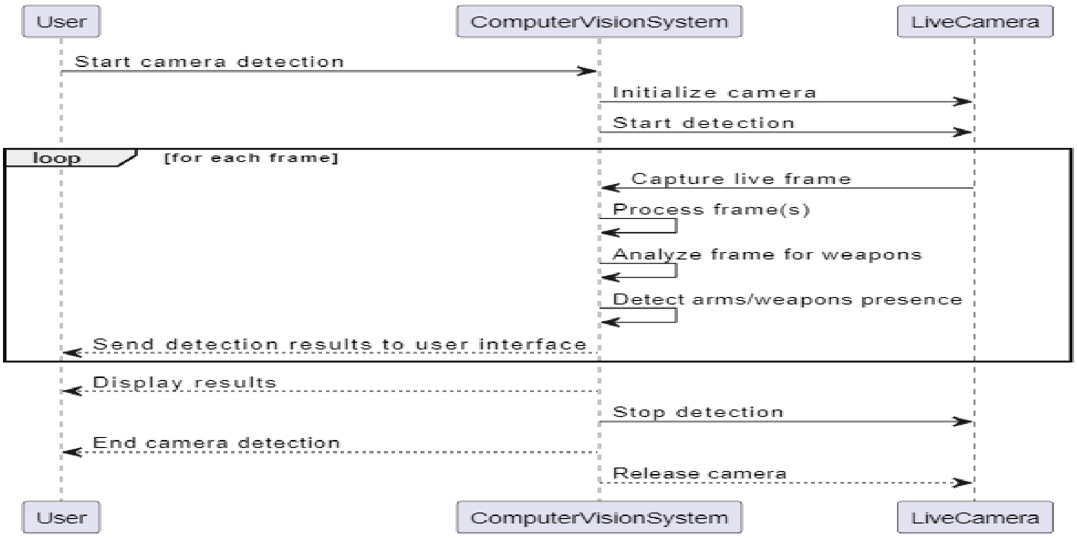


Figure 3.7:Sequence Diagram

CHAPTER 4

IMPLEMENTATION

4.1 Module development

As previously said, it is a pre-trained object detector. A pre-trained model simply means that it has been trained on another dataset. Training a model from the ground up takes a long time; it can take weeks or months to complete the process. A pre-trained model has seen a large number of objects and understands how each one should be categorized. The above-mentioned pre-trained model's weights were obtained by training the network on the Image dataset. As a result, it can only detect items that belong to the classes that were included in the dataset used to train the network.

Rather than being an image classifier, it is supposed to be a multiscale detector. As a result, the classification head is replaced by adding a detection head to this architecture for object detection. The output will now be a vector including the bounding box coordinates and probability classes.

The class of one object in a photograph, for example, is an example of image categorization. Object localization, on the other hand, is the recognition of the region of at least one article in a photograph.

4.2 Result and Discussion

Due to the small number of faces utilized in FYP, the system using manual face identification and weapon recognition did not achieve a recognition accuracy of over 90% *(approximation)*. In this experimental investigation, the system was evaluated under extremely rigorous conditions, and it is expected that real-world performance will be significantly more precise.

The fully automated face detection and identification system lacked the sturdiness required for high recognition accuracy. The only reason for this was that the face recognition subsystem did not show even a smidgeon of invariance to the segmented face image's scale, rotation, or shift issues.

This was one of the system requirements identified in the section. However, if additional processing is used, such as an eye detection technique, to further normalize the segmented face and weapon image, performance will be comparable to that of a manual face and weapon detection and recognition system.

CHAPTER 5

TESTING AND EVALUATION

5.1 CNN Model Accuracy

The deep learning model used for Anomaly detection achieved a training loss of 0.0025 and an accuracy of 100%. In the validation set, the model had a loss of 0.2274 and an accuracy of 95.7%. These metrics indicate that the model performed exceptionally well during training, accurately predicting the presence of anomaly object. However, there was a slight decrease in performance on the validation set, suggesting the need for further optimization to improve generalization to unseen data.

Table 5.1 CNN Model Accuracy

Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	0.3923	0.8413	0.4634	0.2930
2	0.4114	0.8367	0.4643	0.4648
3	0.3905	0.8485	0.4442	0.4414
4	0.3534	0.8596	0.3880	0.5000
5	0.3238	0.8811	0.3618	0.5117
6	0.3178	0.8772	0.3432	0.5273
7	0.2727	0.8981	0.3146	0.5469
8	0.2655	0.9020	0.3981	0.5508
9	0.2648	0.8975	0.2952	0.5664
10	0.2278	0.9177	0.2947	0.5898
11	0.2048	0.9184	0.3537	0.5898
12	0.1715	0.9288	0.2465	0.6133
13	0.1466	0.9484	0.2448	0.6734
14	0.1343	0.9484	0.2257	0.7124
15	0.1180	0.9608	0.2543	0.7914
16	0.1169	0.9628	0.2619	0.8614
17	1.1082	0.9595	0.1481	0.8957
18	0.0991	0.9602	0.2712	0.9267
19	0.0889	0.9732	0.2274	0.9267

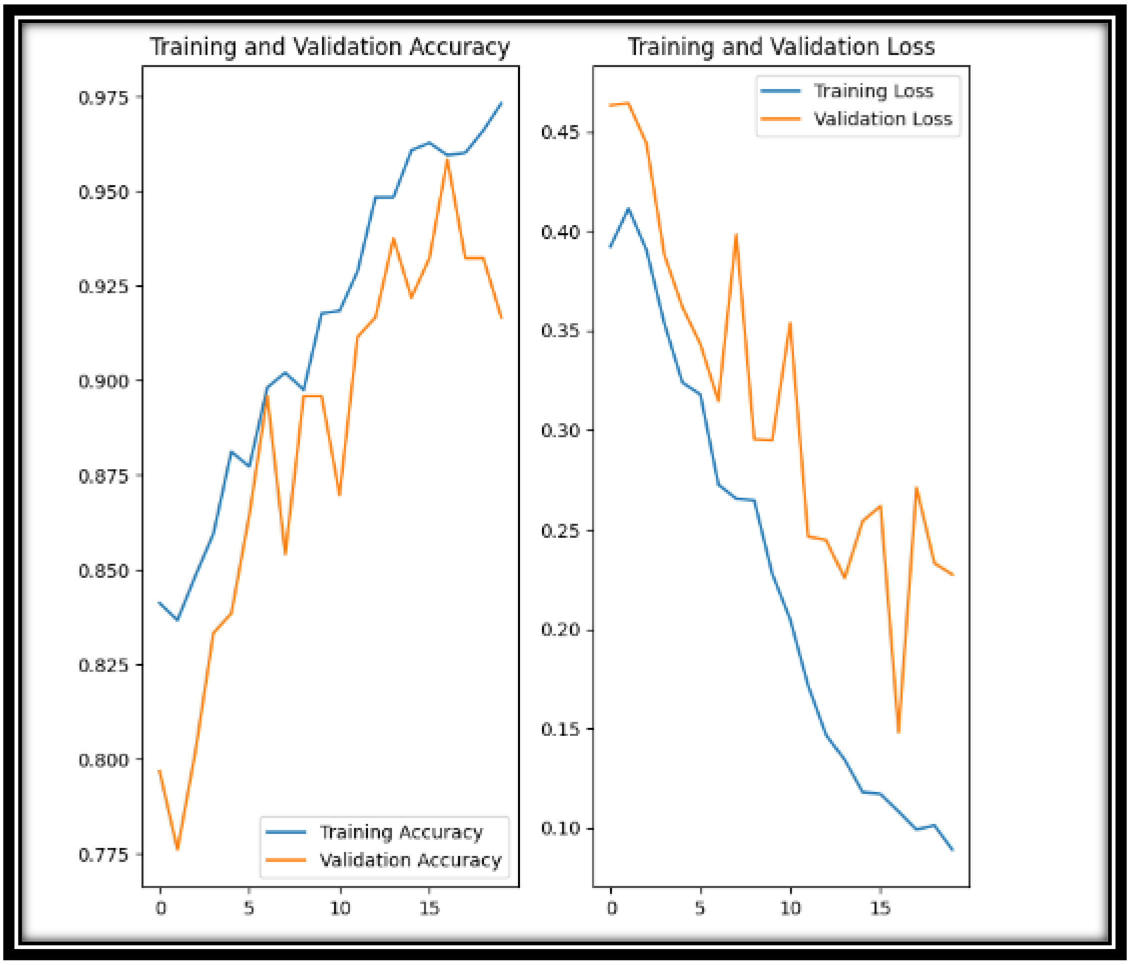


Figure 5.1 Loss and Accuracy Graph

5.2 Predictions

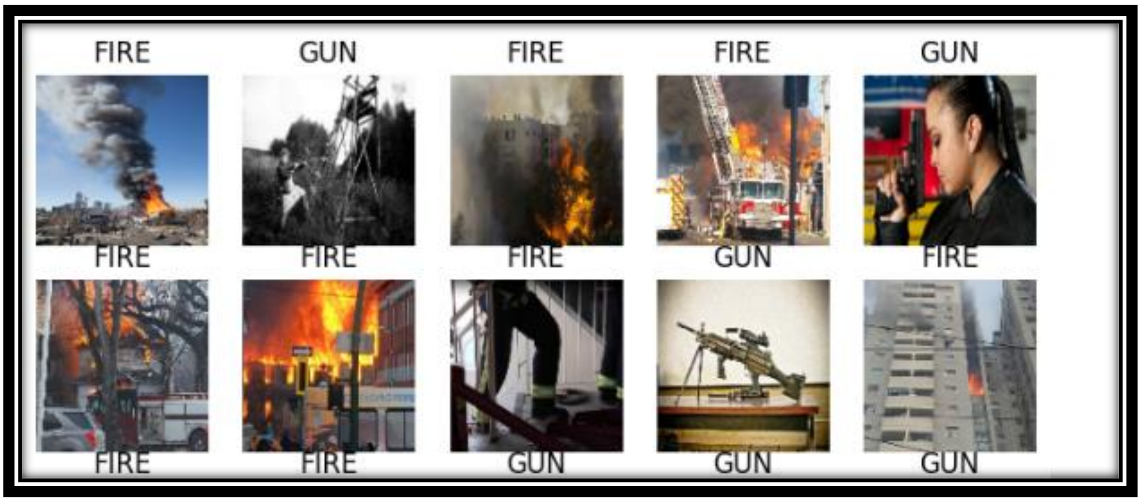
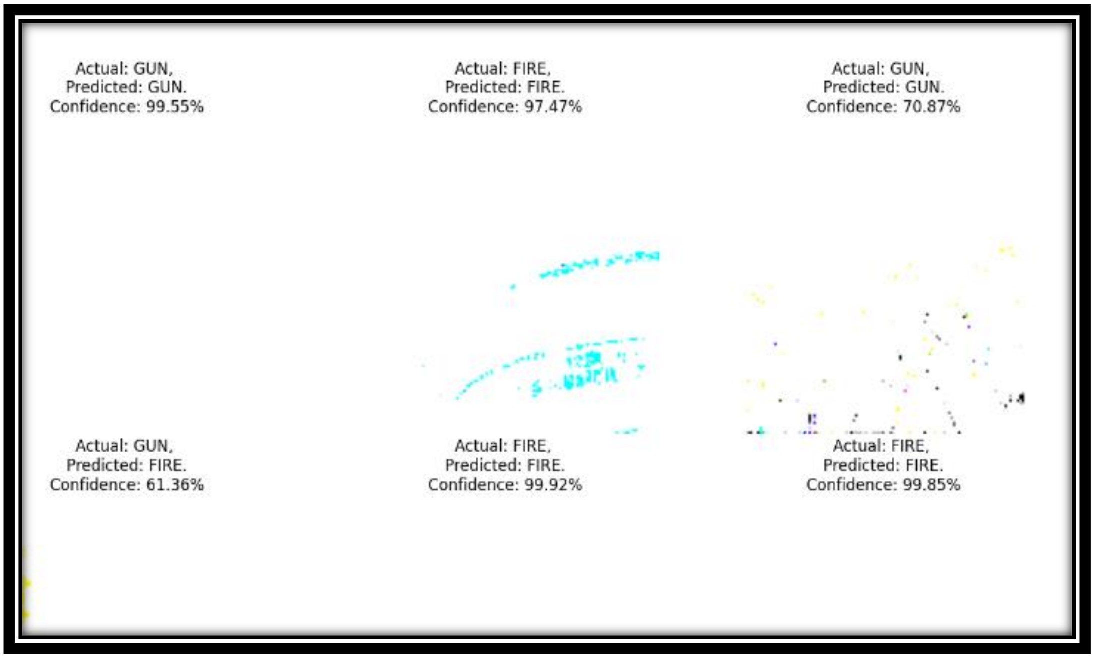


Figure 5.2 Label of Anomaly in Dataset



Actual: GUN,
Predicted: GUN.
Confidence: 99.55%

Actual: FIRE,
Predicted: FIRE.
Confidence: 97.47%

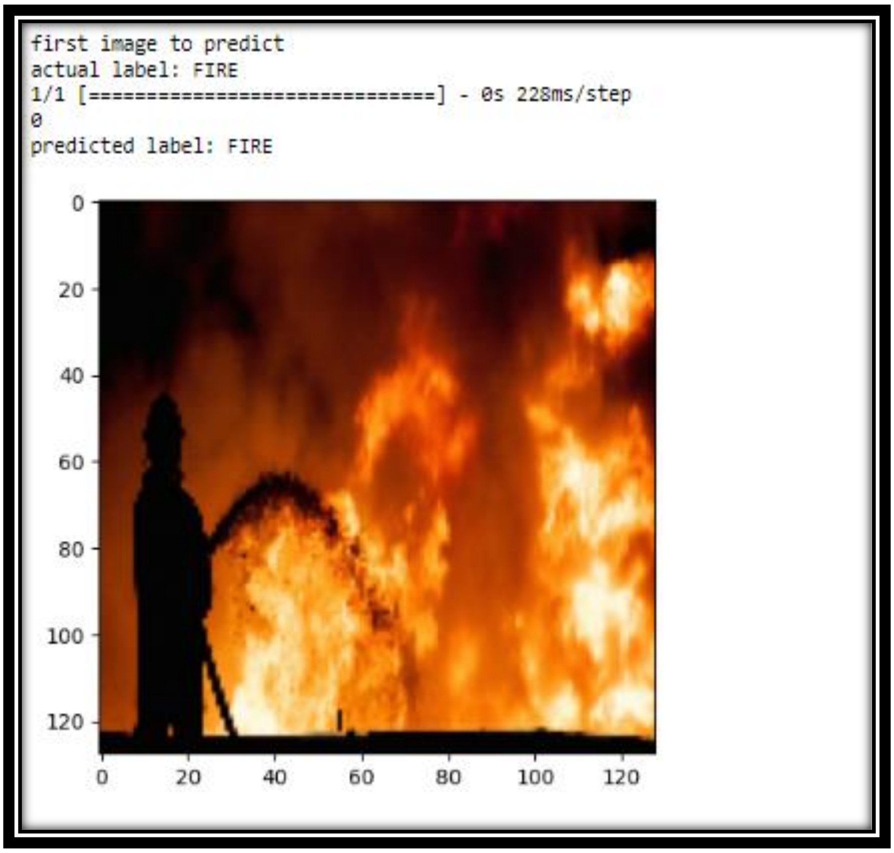
Actual: GUN,
Predicted: GUN.
Confidence: 70.87%

Actual: GUN,
Predicted: FIRE.
Confidence: 61.36%

Actual: FIRE,
Predicted: FIRE.
Confidence: 99.92%

Actual: FIRE,
Predicted: FIRE.
Confidence: 99.85%

Figure 5.3 Predicted pattern of Anomaly



```
first image to predict
actual label: FIRE
1/1 [==============================] - 0s 228ms/step
0
predicted label: FIRE
```

Figure 5.4 Prediction Test of the Anomaly

5.3 Test Plan

Monitoring videos are capable of capturing a variety of anomaly related events. In this project, we propose anomaly detection reading using regular and standard videos. To avoid misunderstanding the incomprehensible segments or clips in training videos, which are time-consuming, we suggest anomaly detection reading using a comprehensive framework for using flexible training labels, i.e., training labels (rare or regular) at video level. Instead of the clip level. On our way, we look at regular and unusual videos as wallets and video segments as an example in detailed reading (MIL), and we automatically read an in-depth abnormal model that predicts anomaly detection high points of abnormal video segments. In addition, we introduce barriers to hold and the temporary slowdown in the job loss rate in order to make it more affective during training

Confounded disclosure is the most common way of distinguishing surprising articles or occasions in informational collections, which are unique about the standard. Rather than standard detachment capabilities, befuddling location is many times utilized in unlabelled information, taking into account just the inside design of the data set. Various practical applications, including network access, fraud detection, health science, and medical science, address this challenge, which is known as anomaly uncontrolled discovery. Although numerous algorithms have been proposed in this field, the research community still lacks comparative spatial analysis and publicly accessible data sets, which is unfortunate. This study addresses these flaws by evaluating algorithms for detecting 19 unsupervised interfaces on ten distinct databases from various application domains. This paper aims to become a well-funded new foundation for research on unidentified anomaly discoveries by publishing data sets and source code. Moreover, this test uncovers the qualities and shortcomings of various strategies interestingly. Notwithstanding the befuddling discovery capability, computation exertion, the impact of boundary settings, and the bewildering conduct of the ground/region complex are shown. In conclusion, we offer recommendations for selecting an algorithm for common tasks in the real world.

Checking recordings are equipped for catching various confounding real factors. In this paper, we propose confounding perusing utilizing customary and standard recordings. We suggest using a comprehensive framework for using flexible training labels, also known as training labels (rare or regular) at the video level, to avoid misinterpreting the incomprehensible segments or clips in training videos, which can be time-consuming. rather than the clip level. On our way, we view customary and strange recordings as wallets and video portions as an illustration in definite perusing (MIL), and we naturally read a top-to-bottom unusual model that predicts befuddling high places of strange video fragments. In addition, to make things more complicated locally during training, we introduce barriers to sparsity and a brief slowdown in the rate of job loss. Observing cameras are progressively utilized in broad daylight places for example streets, convergences, banks, shopping centres, and so on. to increment public security. However, authorities have not kept up with the times. The outcome is that there is an unmistakable deficiency of observation cameras and a wasteful extent of cameras and human screens. The detection of unusual events like road accidents, criminal activity, and other illegal activities is an important function of video surveillance. As a rule, strange occasions are less inclined to happen contrasted with typical exercises.

As a result, developing intelligent computer vision algorithms for automatic video detection is an urgent requirement to cut down on labour and time waste. The anomaly detection system aims to determine when a function departs from normal patterns and the period during which the situation is ambiguous. In the meantime, confounding disclosure can be considered as video understanding at a strong level, befuddling sifting in like manner designs. If something is found to be anonymous, it can be further broken down into specific tasks using classification methods, for instance, an indicator. However, such solutions have limited functionality because they cannot be used to detect other anonymous events. Posting every one of the potential incidents is troublesome. As a result, the fact that the perplexing discovery algorithm does not rely on any prior knowledge of events is interesting. All in all, confounding disclosure ought to be made with negligible management. Strategies in light of strong coding are viewed as portrayals that accomplish the aftereffects of current abnormality location results. Assuming that only a small portion of the video contains standard events, these strategies make use of the first portion to construct a standard event dictionary. The fact that bizarre events do not accurately reconstruct in a standard event dictionary is then the central idea of anomaly detection. These strategies produce elevated degrees of deception for ordinary typical ways of behaving.

Quick advances in shut circle TV (CCTV) innovation and its framework parts like organization, stockpiling, and handling equipment have brought about an enormous number of observation cameras being utilized around the world, and surpassing 1 is assessed. billion worldwide by 2021's end. Video surveillance is a useful tool for law enforcement, transportation, environmental monitoring, and other fields. particularly to work on open well-being and security.

5.4 Testing Modules

Unidentified discovery One of the most difficult and long-lasting computer vision issues is finding anomalies. There are few attempts to detect violence or violence in videos in video surveillance applications. The idea is to use body language and posture to identify human violence. used data from video and audio to find malicious activity in surveillance footage. Gao and Co. proposed descriptions of the violence's progression to identify violence in crowd videos. Recently, a brand-new behaviour-based method for separating violent videos from non-violent ones.

It has always been of great interest to find "unusual" scenarios in data sets. This interaction is generally known as irregularity location or outer identification. Grubbs was given the first possible explanation in 1969 [1]: The hypothetical, also known as the external, is the one that appears to significantly deviate from other members of the sample in which it occurs. Albeit this definition applies today, the rationale for finding these outcasts is altogether different at this point. At that point, the primary justification for the disclosure was to avoid untouchables from later preparation information for example acknowledgment calculations were more delicate to outcasts in the information. Data cleaning is another name for this operation. The interest significantly decreased following the development of solid dividers.

However, things changed in the year 2000, when researchers began to pay even more attention to the source of the anomaly, which was often linked to intriguing events or suspicious data records. Numerous new algorithms have since been created to test this paper. Researchers used low representation to study a standard behavioural dictionary following the success of dictionary-learning and low representation methods in a few computer-assisted visual cues. Patterns with significant reconstruction errors were deemed abnormal behaviour throughout the experiment.

Several strategies for dividing the action in a video are suggested as a result of the successful demonstration of in-depth image classification. However, it is challenging and time-consuming to locate training annotations, particularly for videos.

Auto-learning autoencoders have recently been used to investigate the normal behaviour model and reconstruction losses for ambiguous discovery. Using only weakly labelled training data, our method detects anomaly by taking into account both normal and complex behaviour. A wide range of perplexing facts can be captured by monitoring videos. In this paper, we propose confounding perusing utilizing customary and standard recordings. We propose that instead of using training videos with weak labels, i.e., training labels (rare or regular) video-level, we learn by using a multi-example level framework to avoid misinterpreting the incomprehensible segments or clips in training videos. of the high-quality clip. On our way, we view customary and strange recordings as wallets and video portions as an illustration in definite perusing (MIL), and we naturally read a top-to-bottom unusual model that predicts befuddling high places of strange video fragments. In addition, to make things more practical during training, we introduce barriers to sparsity and a brief slowdown in the rate of job loss.

We are likewise presenting another huge data set for the initial 128 hours of recordings. Contains 1900 long, unidentified real-world surveillance videos that include 13 enigmatic scenes like fighting, car accidents, burglaries, etc. also, and general exercises. There are two ways to use this database. First, the findings of common confounding take into account every anomaly in one group and everything common in the other group. Second, by seeing every one of the 13 abnormal undertakings. According to the results of our tests, our MIL method of anomaly detection performs significantly better than standard methods at anomaly detection.

5.5 Test Cases and Evaluation

Research efforts on Anomaly related video viewing disclosures are dispersed on learning techniques as well as on strategies. Following at first, the scientists zeroed in on the utilization of different spatiotemporal highlights by hand as well as normal imaging procedures. Object-level knowledge and machine-learning techniques for tracking, segmenting, and integrating have recently been used to identify anomaly in video scenes. In this review, we plan to consolidate these methodologies and strategies to give a superior perspective on the various frameworks for tracking down uncertainty. What's more, the decision to observe target changes relies upon framework utilization. Refreshes made to date vary from reconnaissance targets. We have prioritized the following five categories of surveillance goals: vehicle, person, group, thing, and event. Confounded identification is a period delicate cycle, and organization delays and functional postpones make the cloud PC less proficient for touchy frameworks like befuddling recognition. As a result, the use of edge computing in conjunction with cloud computing, which speeds up random detection response times, is the subject of this study. The study also revealed the most recent methods for making computer-assisted video surveillance difficult to understand. There haven't been any previous surveys on how video surveillance and computer use can lead to a conflation in discovery. We update on the mysterious video surveillance discovery and its computer-generated appearance on the edge in this study. This audit will likewise address the difficulties and open doors engaged with finding confounding utilizing edge processing.

Due to the unpredictable nature of anonymous things, insufficient training data, and the circumstances in which events occur, it is extremely difficult to locate unusual events in videos. Lighting conditions (day and night), the surveillance area, the number of businesses on the scene, the variety of environments, and the most crucial working condition for camera imaging are additional features. Numerous publicly accessible databases provide a comprehensive list of data sets categorized for training and preventing abnormal discovery. Because the impact and effect of each database were different, the researchers selected specific data sets and test parameters for each application.

Discovering "unusual" databases in databases has always piqued interest in machine learning. This interaction is generally known as irregularity location or outer identification. Albeit this definition applies today, the rationale for finding these outcasts is altogether different at this point. At that point, the primary justification for the disclosure was to avoid untouchables from later preparation information for example acknowledgment calculations were more delicate to outcasts in the information. Data cleaning is another name for this operation. The interest significantly decreased following the development of solid dividers. However, things changed in the year 2000, when researchers began to pay even more attention to the source of the anomaly which was often linked to intriguing events or suspicious data records.

Contrary to the traditionally guarded circuit breaker, the performance of anomaly detection algorithms for unidentified anomaly detection algorithms is not as straightforward. Rather than just contrasting how much exactness or precision/memory, confounding requests ought to be thought of. An event that is categorized is incorrect in classification. In contrast, unsupervised anomaly was discovered. For instance, on the off chance that a major informational index contains ten befuddles and considers as a part of the main 15, this is as yet a decent outcome, regardless of whether it is perfect. In this instance, the standard test procedure for unregulated anomaly detection algorithms is to use the limit from the first to the last and measure the results according to the data. This creates the character of a single receiving user by producing N tuple values (real positive rate and false positive rate). Then, the region underneath the bend, which is significant, can be utilized as a proportion of the exhibition of the procurement. By following the evidence from the translation of the baffling field, AU also provides a good translation: The anomaly detection algorithm may then yield a random randomly selected event with a lower AUC than the random randomly selected event. As a result, we consider the AUC to be a comprehensive and comparable testing method. The AUC, on the other hand, only considers the level and completely disregards the points difference between related points. Different measures can be utilized to address these weaknesses by utilizing standard examinations. Schubert and Co. 38] look at changed ways to deal with position connections, the Spearman is our only model, and the Kendall τ model is an option in contrast to the AUC with an emphasis on coordinating outside gatherings. The fact that approaches like the area below the curve to remember accuracy or the correlation coefficient may better emphasize minute changes in acquisition performance may be a second potential drawback of using AUC for problems with unequal classes. However, because of your actual interpretation, AUC-based experiments have become the de facto standard for unidentified and abnormal findings. As a result, they also serve as a selective measure in our experiments. Most of the time, when it is used in a standard division function, it has a parameter that can be changed, like k. For unidentified disarray identification, AUC is electronic by changing the outer limit in the rundown of requested results. As a result, if the parameter needs to be checked, such as if k is different, more AUC values are displayed.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

We investigated and categorized a variety of anomaly detection methods used in a variety of video surveillance scenarios, as will be discussed in this section. As the setting oddity is emotional, we thought about observation situations with people on foot, swarm, traffic, ventures, and public spots. We talked about how anomaly detection methods have changed over time and the surveys that have been done so far. Anomaly detection learning strategies, methods, approaches, and scenarios were emphasized. This study means to give definite knowledge and related varieties in peculiarity discovery procedures. We predict that among the modelling algorithms discussed, reconstruction- and prediction-based methods will prevail in the coming years. We investigate how edge computing and recent advancements in anomaly detection can be combined to create applications for real-time, time-saving video surveillance. Utilizing LSTMs and GANs in conjunction with edge computing for anomaly detection has significant potential in the future. We also show that edge device anomaly detection hasn't been done much research and still needs a lot of work to be at the cutting edge of anomaly detection and intelligence surveillance.

We have investigated and distinguished the various baffling diagnostic techniques utilized in various video surveillance scenarios, as described here. We have taken into account pedestrian patrols, the crowd, traffic, industries, and public places because the anomaly is consistent with the context. Through the years and the surveys that have been conducted thus far, we have discussed the emergence of perplexing discovery strategies.

6.2 Future work

Disruptions happen due to changes and they may cause severe outages impacting your business. But if you can see the changes that are happening in your environment in near real-time, you can prevent these disruptions, and thus the outages. In today's digital business environment where a significant portion of the business runs through applications, being reactive to outages isn't an option anymore.

Take a bank, for example, that doesn't know what every high-risk incident will look like. In this case, it's impossible for the bank to search in advance for all incidents, write rules to identify anomalous data, or even build statistical models to prevent it. Only with a machine learning approach that adapts to the continuous change can safeguard against future unknown IT issues.

Logic Monitor has offered Anomaly Detection for metrics for quite some time already, but up until now, we haven't specialized in logs. In future we will be making Anomaly Detection for logs available to all of our customers in the form of our newest product, LM Logs™. Our groundbreaking algorithmic approach to logs will make it easier for you and your team to filter signals from the noise, and solve problems faster than ever before.

LM Logs is coming soon.

CHAPTER 7References

[1] Real-Time Anomaly Detection in Surveillance Cameras for Intelligent Video Analysis" Authors: Nguyen, Thanh N., and Thi-Lan Le Publication Year: 2019.

[2] VNAnomaly: A novel Vietnam surveillance video dataset for anomaly detection Tu N. Vu; Toan T. Dinh; Nguyen D. Vo; Tung Minh Tran; Khang Nguyen 2021 8th NAFOSTED Conference on Information and Computer Science (NICS).

[3] Anomaly Detection using CNN with SVM P. Sridhar; S.D. Arivan; R. Akshay; R. Farhathullah 2022 8th International Conference on Smart Structures and Systems (ICSSS).

[4] L. Dong, Y. Zhang, C. Wen and H. Wu, "Camera anomaly detection based on morphological analysis and deep learning," 2016 IEEE International Conference on Digital Signal Processing (DSP), 2016, pp. 266-270, doi: 10.1109/ICDSP.2016.7868559..

[5] Zhang, C., Chen, W., Chen, X., Yang, L., Johnstone, J.: A multiple instance learning and relevance feedback framework for retrieving abnormal incidents in surveillance videos. J. Multimedia 5(4), 310–321 (2010).

[6] Zhang, C., Chen, W., Chen, X., Yang, L., Johnstone, J.: A multiple instance learning and relevance feedback framework for retrieving abnormal incidents in surveillance videos. J. Multimedia 5(4), 310–321 (2010).

[7] Adam, E. Rivlin, I. Shimshoni, D. Reinitz, and M. Intelligence, “Robust real-time unusual event detection using multiple fixed-location monitors,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 3, pp. 555–560, 2008..

[8] Y. Cong, J. Yuan, and J. Liu, “Sparse reconstruction cost for abnormal event detection,” in Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 2011..

[9] Basharat, A. Gritai, and M. Shah, “Learning object motion patterns for anomaly detection and improved object detection,” in Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, June 2008..

[10] Shuai Bai, Zhiqun He, Yu Lei, Wei Wu, Chengkai Zhu, Ming Sun, andJunjie Yan. Traffic anomaly detection via perspective map based on spatial-temporal information matrix. In Proc. CVPR Workshops, 2019..

[11] Yang Cong, Junsong Yuan, and Ji Liu. Sparse reconstruction cost for ab-normal event detection. In CVPR 2011, pages 3449–3456. IEEE, 2011.

[12] Anomaly Detection Algorithm Based on Global Object Map for Video Surveillance System H.C. Shin; Jiho Chang; Kiin Na 2020 20th International Conference on Control, Automation and Systems (ICCAS).

APPENDIX A: Computer Programme Listing (CODE)

```
from flask import Flask, render_template, Response
import cv2
from keras.models import load_model
import warnings
from tkinter import *
from PIL import Image, ImageTk, ImageOps
import tkinter.font as font
import numpy as np
import sys
import time
import os

process_counter=0
warnings.filterwarnings("ignore")
model = load_model('model.h5')
data = np.ndarray(shape=(1, 224, 224, 3), dtype=np.float32)
size = (224, 224)

app = Flask(__name__)
#video = cv2.VideoCapture(0)
video = cv2.VideoCapture(1,cv2.CAP_DSHOW)

@app.route('/')
def index():
    #return "Server is Running "
    return render_template('index.html')



def gen(video):
    recognizer = cv2.face.LBPHFaceRecognizer_create()
    recognizer.read('trainer/trainer.yml')
    faceCascade = cv2.CascadeClassifier(cv2.data.haarcascades +
'haarcascade_frontalface_default.xml')
    font = cv2.FONT_HERSHEY_SIMPLEX
    id = 0
    names = ['None', '1', '2', '3', '4', '5']
    global model
    global process_counter
    global data
    global size
        # Define min window size to be recognized as a face
    minW = 0.1*video.get(3)
    minH = 0.1*video.get(4)
    while True:
        success, image = video.read()
        cv2image= cv2.cvtColor(image,cv2.COLOR_BGR2RGB)
        img = Image.fromarray(cv2image)
        gray = cv2.cvtColor(image,cv2.COLOR_BGR2GRAY)
        faces = faceCascade.detectMultiScale(
        gray,
        scaleFactor = 1.2,
        minNeighbors = 5,
        minSize = (int(minW), int(minH)),)

        for(x,y,w,h) in faces:


            cv2.rectangle(image, (x,y), (x+w,y+h), (0,255,0), 2)


            id, confidence = recognizer.predict(gray[y:y+h,x:x+w])
            if (confidence < 35):
                id = names[id]
                confidence = "  {0}%".format(round(100 - confidence))
                print("\n Detected  : ")
                print(str(id))
            else:
```

```
            id = "unknown"
            confidence = "  {0}%".format(round(100 - confidence))
            print(confidence)
        imagee = img
        imagee = ImageOps.fit(imagee, size, Image.ANTIALIAS)
        image_array = np.asarray(imagee)
        normalized_image_array = (image_array.astype(np.float32) / 127.0) - 1
        data[0] = normalized_image_array
        prediction = model.predict(data)
        print(prediction)
        #print(prediction[0][0])
        #print(prediction[0][1])
        #if(prediction[0][0]>prediction[0][1]):
        #  print("greater")
         # if prediction[0][0]>(9.7):
          #   cv2.putText(image, "weapon detected", (50, 50),
cv2.FONT_HERSHEY_SIMPLEX, 1, (255, 0, 0), 3)
          #else:
           # cv2.putText(image,  "            ", (50, 50), cv2.FONT_HERSHEY_SIMPLEX,
1, (255, 0, 0), 3)
        #else:
         #   cv2.putText(image,  "            ", (50, 50), cv2.FONT_HERSHEY_SIMPLEX,
1, (255, 0, 0), 3)

        ret, jpeg = cv2.imencode('.jpg', image)
        frame = jpeg.tobytes()
        yield (b'--frame\r\n'
               b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n\r\n')

@app.route('/camvid')
def camvid():
    global video
    return Response(gen(video),
                    mimetype='multipart/x-mixed-replace; boundary=frame')
```

```
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=2204, threaded=True)
```