International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 05 | May 2023 www.irjet.net p-ISSN: 2395-0072

Real-time Anomaly Detection and Alert System for Video Surveillance

Seemantula Nischal[1], Bhunesh.k[2], Ashwin Sathappan v[3]

[1,2] B.E Student, Computer Science, RMD Engineering College, Chennai, India
[3] B.Tech Student, Information Technology, RMD Engineering College, Chennai, India

---

Abstract - *The objective of this project is to develop a real-time video surveillance system that can detect and classify various types of anomalies such as theft, street crime, unauthorized access and burglary. The system will use a combination of deep learning models, such as CenterNet and Graph Convolutional Networks (GCN), to detect and classify the anomalies in real-time CCTV footage. Once an anomaly is detected, the system will trigger an alert to the nearest police station the system is incorporated with Twilio Video API to send alerts to officials in real-time and send a video message along with the location information of the anomaly. The system stores information about the anomaly in a database [MindsDB], including the type of anomaly, severity level, and longitude and latitude of the location. Overall, the real-time anomaly detection and alert system for video surveillance will provide an efficient way to detect and track criminal activities in real-time, enhancing public safety and security.*

Key Words: **Real-time anomaly Detection, UCF Dataset, MindsDB, Crime India Dataset, Twilo Video-based API**

1. INTRODUCTION

Video surveillance systems are critical to public safety and security. Real-time monitoring and analysis of video footage have become more important in detecting and responding to potential threats or unlawful behavior. Traditional surveillance systems rely heavily on manual monitoring, which is time-consuming and prone to human error. As a result, there is a growing demand for automated systems that can detect and categorize anomalies in real-time video streams.

Detecting and categorizing a wide range of anomalies such as theft, street crime, illegal access, and burglary. By using breakthroughs in deep learning models, we want to increase video surveillance capabilities for efficient anomaly identification. To achieve this goal, we employ a range of deep learning models This research project's purpose is to develop a real-time video surveillance system capable of, including CenterNet and Graph Convolutional Networks (GCN). These models have proven to be extremely effective in object detection and identifying spatial relationships in complex circumstances. CenterNet's object recognition is precise and efficient, allowing us to discover anomalies inside video frames. GCN, on the other hand, captures contextual information and object relationships using the power of graph-based representations, allowing for robust anomaly categorization. When an anomaly is detected, our system uses the Twilio Video API to notify the nearest police station. Because the warnings include real-time video messaging and location information, authorities can respond and interfere immediately.

Furthermore, precise information about each abnormality is saved in a database (MindsDB). such as its classification, severity degree, and exact geographical location. This allows for comprehensive post-analysis and a better knowledge of the patterns and trends related to criminal activities.

Standard surveillance techniques are outperformed by the proposed real-time anomaly identification and alert system. It boosts the efficiency and effectiveness of video surveillance by automating the detection and classification of abnormalities. The ability to swiftly alert authorities to potential threats enables faster response times and proactive intervention, resulting in a safer and more secure environment.

In the next parts, we will look at the technical aspects of our system, such as the deep learning models employed, the alert generating process, database storage, and so on. is category, severity degree, and specific geographical location. This allows for comprehensive post-analysis and a better knowledge of the patterns and trends related to criminal activities.

Standard surveillance techniques are outperformed by the proposed real-time anomaly identification and alert system. It boosts the efficiency and effectiveness of video surveillance by automating the detection and classification of abnormalities. The ability to swiftly alert authorities to potential threats enables faster response times and proactive intervention, resulting in a safer and more secure environment.

2. DATASET

2.1 UCF Crime – Dataset

This dataset includes a sizable number of security recordings documenting many types of criminal activity, including theft, vandalism, and assault. It is frequently used to assess algorithms for action identification and anomaly detection. The dataset includes of surveillance films that were taken in a variety of real-world situations when criminal activity was taking place. Theft, vandalism, assault, and burglary are only a few of the 14 types of crimes covered. The collection seeks to serve as a reference point for creating algorithms that can automatically find and identify illegal activity in surveillance footage.

2.1.1 Classes

There are a total of 14 Classes in the dataset:

1. Abuse

2. Arrest

3. Arson

4. Assault

5. Burglary

6. Explosion

7. Fighting

8. Normal Videos

9. RoadAccidents

10. Robbery

11. Shooting

12. Shoplifting

13. Stealing

14. Vandalism

2.1.2 Contents

There are 1,900 video clips altogether in the UCF-Crime Dataset that were collected from various security cameras. Each video clip normally lasts 30 seconds and features a particular incidence of illegal behaviour. The movies are recorded from various angles, locales, and lighting setups, offering a variety of algorithm assessment problems. Each video clip in the collection comes with frame-level annotations defining the bounds of the illegal activity's time period. An accurate evaluation of detection and recognition methods is made possible by the annotations' start and end timestamps. The primary participants in the illicit actions are identified in the dataset via bounding box annotations. 1,266,345 images make up the whole train subset.The test subset has 111,308 images in total.

Some of them are attached below:

Fig -1: UCF Crime Dataset images

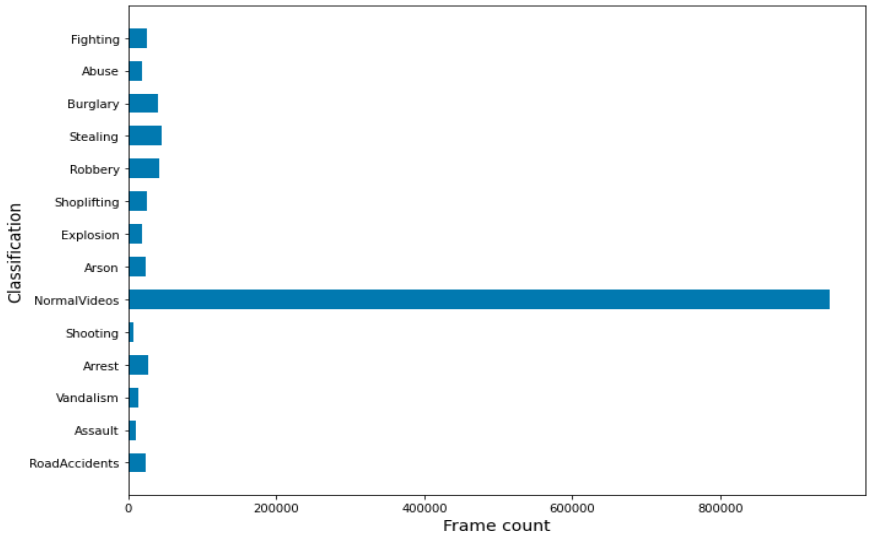

Chart -1: Classification – Frame count graph

2.2 Crime India Dataset

Due to the country's numerous demographics and locations, the specific crime statistics for India may be fairly broad and varied. Despite the fact that there isn't a single complete dataset that covers all crimes in India, a number of sources and organisations do offer statistics on crimes. The following important sources and statistics about crime in India are listed:

2.2.1 National Crime Records Bureau (NCRB)

The main organisation in charge of gathering and disseminating information on crimes in India is the National Crime Records Bureau. An yearly report from NCRB titled "Crime in India" contains data and analysis on a variety of crimes committed around the nation. This dataset includes comprehensive statistics on all crime-related issues that occurred in India starting in 2001. This dataset may be used to investigate a variety of aspects. Some of the aspects include :

1.Use of Firearms in Murder Cases

2.Human Rights Violation by Police

3.Victims of Kidnapping and Abduction for specific purpose

4.Custodial deaths

And more.

2.2.3 WORKING

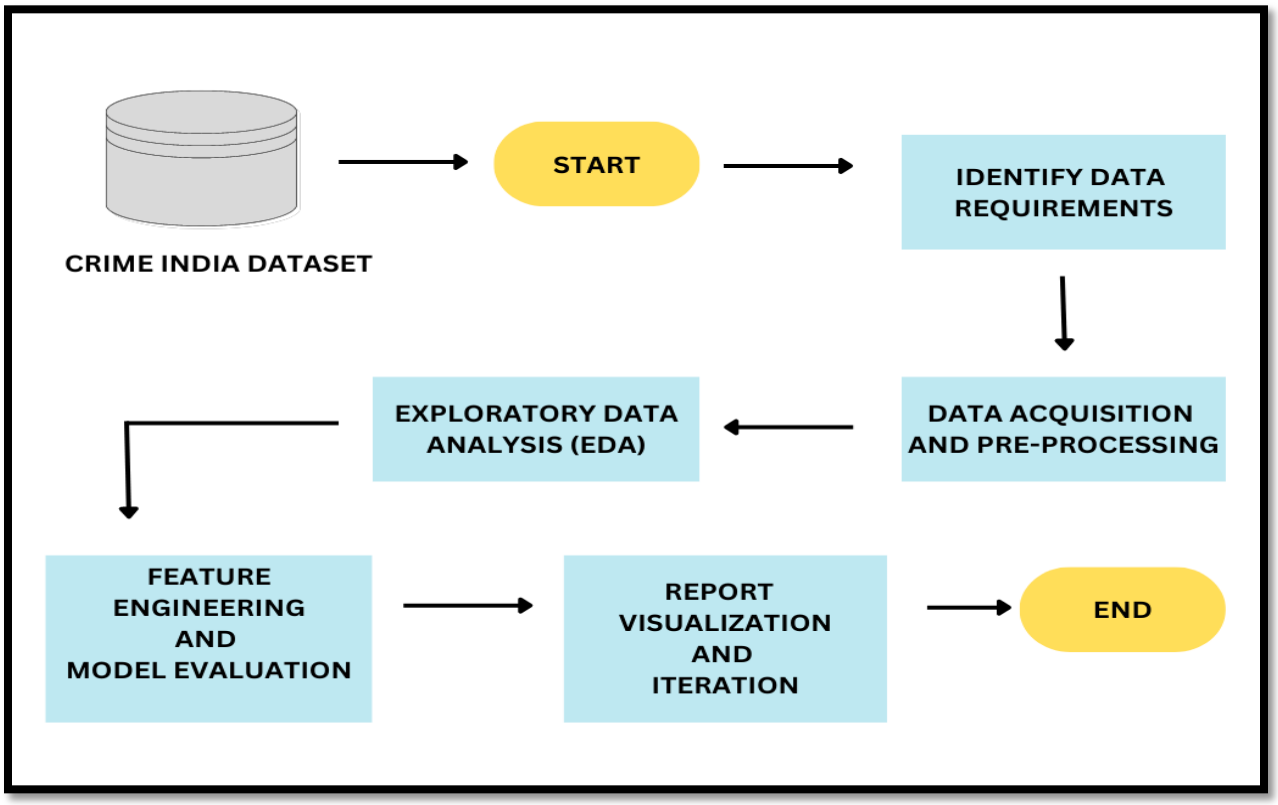

Fig -2: Data Flow Diagram for crime India dataset

3. RELATED WORKS

Several fields of study have played important roles in the field of real-time anomaly detection and alarm systems for video surveillance. Faster R-CNN, YOLO, and SSD object detection algorithms have been widely used for the accurate and efficient identification of objects in video streams. Anomaly detection approaches such as one-class SVM and autoencoders have been investigated to detect deviations from typical behavior, however, they frequently have significant false positive rates. Graph Convolutional Networks (GCN) have emerged as a useful method for collecting spatial connections and contextual information in complicated scenarios, which has resulted in enhanced anomaly detection and item categorization. The integration of communication tools such as the Twilio Video API enables the development of real-time alerts, ensuring a rapid response to identified irregularities. MindsDB and other database systems provide efficient storing and processing capabilities for anomalous data, allowing post-analysis and predictive modeling. While previous research has made substantial advances, issues such as balancing accuracy and processing speed, dealing with complex situations, and resolving privacy concerns remain. We want to progress the field in this research by employing deep learning models like CenterNet and GCN for precise and efficient real-time anomaly detection and categorization. In addition, we prioritize rapid alarm production, database storage, and thorough post-analysis to improve public safety and security. Our research intends to solve the limitations noted in the existing literature and contribute to the improvement of real-time anomaly detection for video surveillance through rigorous assessment.

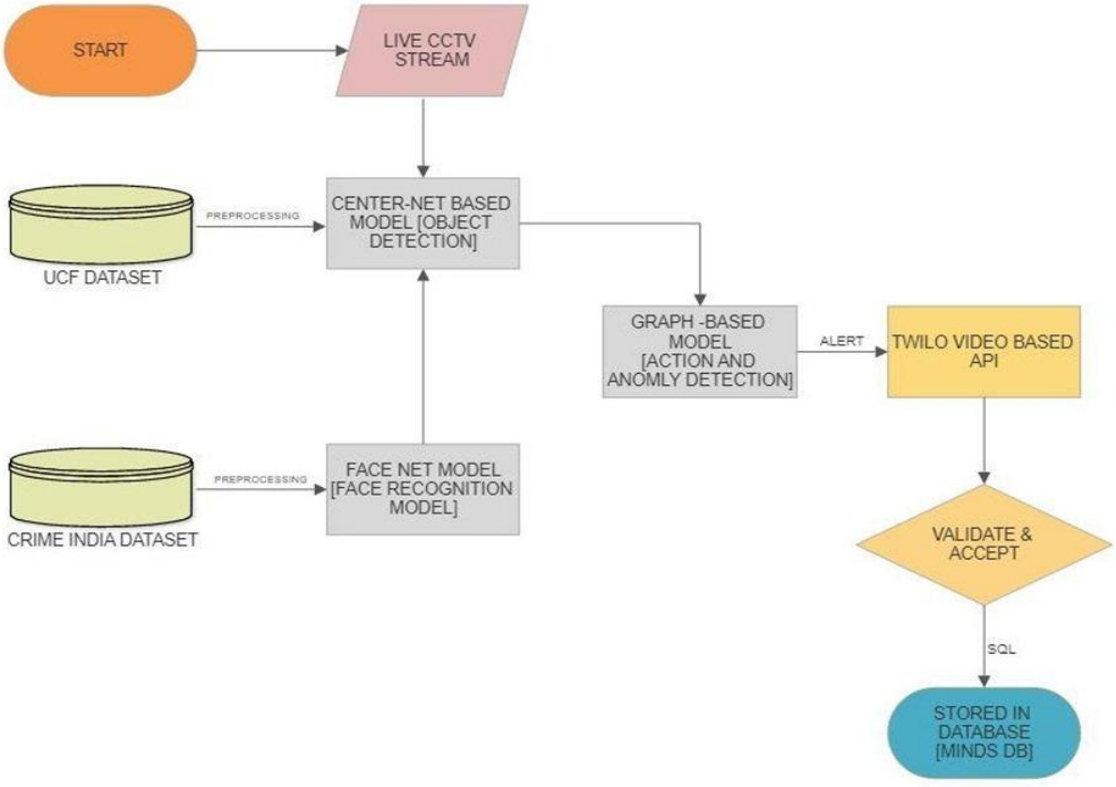

Fig -3: Flow chat of the model

4. FACE RECOGNITION MODEL

4.1 Introduction

Artificial intelligence programmers called face recognition models are used to recognize and authenticate people by looking at their facial traits. In these models, significant characteristics are extracted from facial photos using deep learning techniques, and the features are then mapped onto a high-dimensional space where like faces are clustered together.

Face recognition often entails the following crucial steps:

4.1.1 Face Detection

Face detection is the initial stage in finding faces in a frame of an image or video. In order to do this, algorithms that examine patterns, forms, and color changes are used to identify the face area.

4.1.2 Feature Extraction

Following the discovery of a face, the model extracts pertinent elements from the facial picture. Convolutional neural networks (CNNs), a type of deep learning technology, are frequently used to capture distinguishing face features including the appearance of the eyes, nose, and mouth.

4.1.3 Face Representation

Using the retrieved features, a compact representation or embedding is created that captures the distinctive aspects of the face. These depictions are made to be resistant to changes in stance, lighting, and face emotions.

4.1.4 Face Matching

The model compares the embedded representation of a query face with a database of recognized faces during the recognition phase. The similarity of the embeddings is assessed using a variety of matching methods, including cosine similarity and Euclidean distance.

4.1.5 Verification or Classification

Depending on the similarity score, the model may carry out either face verification or classification, which involves deciding if a person matches a certain identity claim.

5. SFC DATSET

The SFC dataset consists of 4.4 million labelled faces from 4,030 individuals, each of whom has 800 to 1200 pictures. Only the most recent 5% of each identity's face photographs are used for testing. To mimic ongoing identification across time, this is carried out in accordance with the time-stamp of the photos. The high number of photos per individual offers a special chance to learn the invariance required for solving the fundamental issue of face recognition.

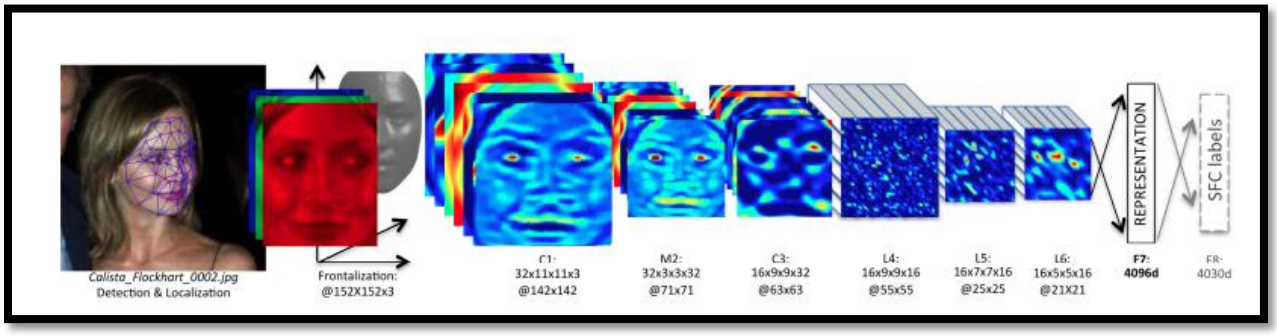

Fig -4: Face Recognition model classification

6 CENTER NET BASED MODEL

Popular object identification software called CenterNet may be used to find items in live cctv . It uses the idea of keypoint estimation to find the centre points of objects and is based on a one-stage detector design.

6.1 HOW DOES CENTER NET BASED MODEL WORK WITH LIVE CCTV

The following actions are conducted in order to get live CCTV photos of the UFC bout. First, the footage's individual video frames are removed. Depending on the situation, these frames go through various image processing steps including cropping, normalising, and scaling. The following stage is identifying and locating any human beings visible in the photos. The detection of individuals uses a person detection method, similar to object detection. Once they are discovered, the observed persons are tracked using a technique like object tracking.

Several strategies are utilised to obtain relevant information from the watched human figures. To detect and analyse relevant properties in the figures, methods such as posture estimation and activity identification are used. This aids with the collection of relevant data such as body postures, motions, or activities displayed by each individual.

A unique identification number is provided to each tracked individual in order to keep track of them. This one-of-a-kind ID is used to keep track of the IDs and the related monitored statistics. Furthermore, the centroid or centre point of each human figure is determined by their body positions. This enables for better figure organisation and layout.

The distance between the centres of each human figure is estimated to group people together. They are regarded as being close by if their centres are within a specific range of one another. This grouping facilitates data organisation and produces a more convincing depiction of the situation.

The input data is then combined in order to prepare it as input for the center-based model. This covers the locations of the human figure groupings whose centres were established in accordance with the preceding procedures. The center-based model is then applied to the output to do additional analysis or processing. The following procedures may be used to train a CenterNet-based model after gathering data from the UFC dataset:

6.1.1 Data Preparation

Annotate the bounding boxes for the items you wish to detect (such as fighters) in the UFC dataset and resize the photos to a uniform size. The bounding box coordinates as well as the class labels should be included in the annotations.

6.1.2 Network Architecture

Construct the CenterNet-based model's network architecture. For feature extraction, it often comprises of a backbone network (like ResNet), and for object identification, it typically consists of a number of additional layers. The inclusion of a keypoint estimate branch is CenterNet's key feature.

6.1.3 Training

Set the network's initial weights at random and train it with the provided dataset Forward-passing the pictures through the network, computing the difference between the predicted keypoints and the ground truth keypoints, and back-propagating the gradients to update the network weights are all steps in the training process. Typically, stochastic gradient descent (SGD) or Adam optimisation algorithms are used for this procedure.

6.1.4 Keypoint Estimation:

The model learns how to anticipate an object's centre point by examining its keypoints during training. These keypoints are frequently represented as heatmaps, with each heatmap representing a certain kind of item. The greatest value in a heatmap represents the anticipated centre point of an item falling within that category. The algorithm acquires the capacity to recognise and pinpoint significant locations on diverse objects, understanding their spatial relationships and making precise predictions by utilising these heatmaps and their related values.

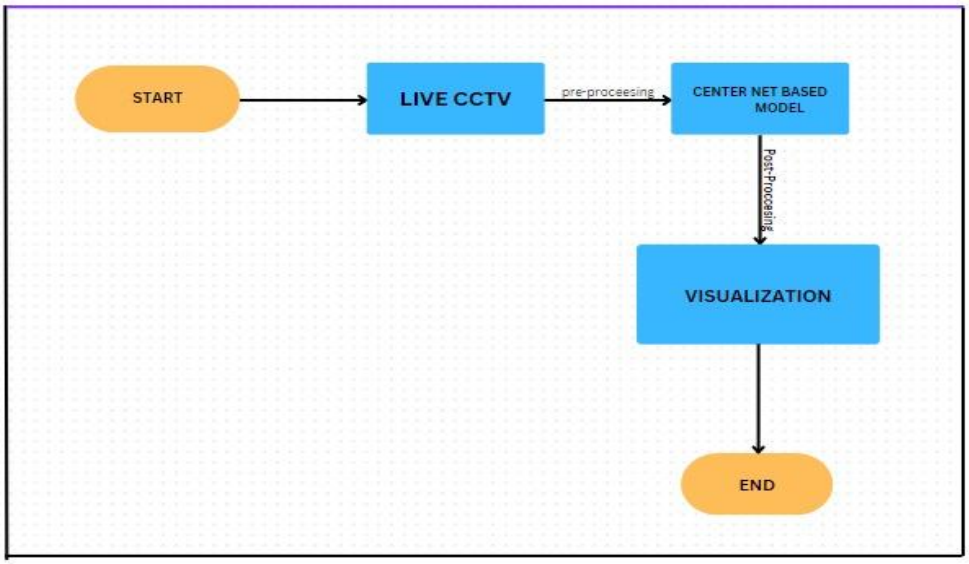

Fig -5: Flow chart for Center Net Model

Once the model has been trained, you can use it to recognise objects in fresh photos from the UFC dataset or in any other image. The trained model will produce heatmaps for each class during inference and identify the centre points of the objects based on the heatmaps' greatest values. The bounding boxes and class labels of the identified objects may then be calculated using these centre points.

7. GRAPH-BASED ANOMALY

Machine learning models that employ graph theory to analyze intricate interactions between elements in a dataset are called "graph-based models for action and anomaly detection. These models are frequently used in security and video surveillance applications to find unusual behavior or illegal activity. The fundamental concept underlying graph-based models is to visualize the data as a graph, with the entities acting as nodes and the connections between them as edges.

For instance, in a video surveillance application, the nodes may stand in for individuals or objects in the video, and the edges might stand in for geographical or temporal interactions between them. The graph-based approach can utilize a variety of algorithms to identify unusual behavior or illegal conduct once the network has been formed. Searching for subparagraphs that stand out or don't fit the norm is a typical tactic. A graph-based model, for instance, may identify a subgraph where a person is traveling in the other way as anomalous if most individuals in the video are walking along a particular path. we need to first develop a graph representation of the data to use a graph-based model on a dataset of crimes. In this case, the participants in the crime may be represented as nodes and the connections among them (such as co-occurrence or communication) as edges.

The graph-based model may be trained on the created graph once it has been created. It's crucial to remember that the field of research into graph-based models for action and anomaly detection is challenging and dynamic. The particular strategy and techniques employed can change based on the application and dataset.

A broad description of the operation of a graph-based action and anomaly detection model is given below:

7.1.1 Data Representation

The dataset is organized as a graph, where nodes are the things or objects and edges are their connections or interactions. While the edges records linkages such as co-occurrences, temporal sequences, or correlations, the nodes for crime activity detection might represent people, places, or events.

7.1.2 Graph Construction

A graph is built using the information at hand. The nodes may represent the people who were engaged in the crimes, and the edges could reflect the relationships between them based on co-occurrence in the same incident or temporal proximity, for instance, if the dataset contained information on criminal incidents.

7.1.3 Feature Extraction

To describe the attributes of the nodes and edges, pertinent features are retrieved from them. These characteristics may include characteristics like timestamps, geographic data, social connections, or any other pertinent data that aids in describing the entities and their interactions.

7.1.4 Graph Analysis

The generated graph is analyzed using a variety of graph-based methods and methodologies. Typical methods include graph clustering, community discovery, centrality analysis, and graph pattern mining. These techniques seek to locate patterns, groups, or subgraphs that correlate to routine behavior or criminal activity.

7.1.5 Anomaly Detection

The model may identify actions or groups of activities that are suggestive of criminal behavior based on the graph analysis. It may either learn from labeled data to find Anomaly Detection anomalies that depart from anticipated behavior, or it can compare observed patterns against known patterns of criminal activity. It is possible to mark unusual or aberrant patterns as probable criminal activity for further examination. It's vital to remember that depending on the individual study or application setting, the precise implementation and techniques employed in graph-based models for action and anomaly detection may differ. Various graph analyses To get the required outcomes, methods, and machine learning algorithms might be combined. The graph-based model can assist in identifying patterns of criminal behavior, revealing hidden linkages, and highlighting suspicious behaviors that may need the attention of law enforcement or security professionals when this method is applied to a dataset of crime activity.

These models offer a potent tool for comprehending intricate criminal networks and spotting unusual or unlawful behavior by utilizing the graph structure and examining relationships between entities. Graph-based anomaly detection techniques use the links and interconnections between the data pieces and interconnections between the data pieces are used by graph-based anomaly detection techniques to find abnormalities.

Here is a diagram showing each step in detail:

Step 1 Data Points

In Step 1 We begin with a set of data points that are represented as graph nodes. Each node stands for a distinct data point. We have a graph with numerous nodes in this example.

A B

| |

C---D---E

Step 2 Connections

In Step 2 Then, depending on the nodes' connections or similarities, we create edges between them. These links show the relationships between the data points. To illustrate the strength or similarity of the relationships, or similarities. These connections indicate how the data points are related to each other. The edges can be weighted to represent the strength or similarity of the connections.

A B

\ |

C---D—E

Step 3 Normal Data Points

In Step, Three We presume that the majority of the data points in a graph-based anomaly detection technique are normal or non-anomalous. The bulk of nodes in the network are regarded as regular data points

A B

\ |

C---D—E

Step 4: Finding anomalies

Data points that considerably depart from the norm or have odd relationships in the graph are considered anomalies. Anomalies can be found by inspecting the nodes' characteristics and the graph's overall structure. In this case, Node B's peculiar connection pattern identifies it as an abnormality.

A (Normal) B (Anomaly)

\ |

C---D—E

Step 5 Labelling the Anomaly

Anomalies can be labeled or reported for additional inquiry or action after they are discovered.

A (Normal) B (Anomaly)

\ |

C---D—E

8. Twilo video-based API

Twilio Video is an API (Application Programming Interface) provided by Twilio, a cloud communications platform, that enables developers to incorporate video communication capabilities into their applications or services. It allows you to build applications that support real-time video calls, video conferencing, and live streaming. Using a client-server architecture, Twilio Video establishes connections between participants. Here is a general description of how it works:

8.1 Integration on the client side

You include the Twilio Video SDK (Software Development Kit) in the client-side code of your application. The tools and capabilities required to establish and manage video communication are provided by this SDK.

8.2 Authentication & token generation

The server-side code of your application creates a special access token using the Twilio API whenever a user requests to join a video session. The user's identity is confirmed by this access token, which also gives them access to the video session.

8.3 Room creation

The user joins a virtual "room" where the video session occurs after authorization. The room is created by the server-side code, which also assigns a special Room ID that is used to track down and manage the session's participants.

8.4 Media transmission

The client of each participant's device records audio and video using the microphone and camera. After that, Twilio's media servers receive the encoded and transmitted media. Twilio manages the media routing, allowing participants to communicate in real time.

8.5 Signaling

Peer-to-peer communication between participants is made possible by Twilio Video's usage of WebRTC (Real-Time Communication) technology. To start and maintain the session, signaling messages are sent back and forth between the clients and Twilio's infrastructure, including connection requests, media metadata requests, and participant changes.

8.6 Video Features

For Twilio Video include screen sharing, chat, muting and unmuting of participants, bandwidth management, and more. Depending on your unique needs, these functions may be modified and integrated into your application. It's important to note that Twilio Video offers comprehensive documentation, sample code, and client SDKs for a variety of platforms (web, iOS, Android) to ease integration and make it simpler for developers to make use of its features.

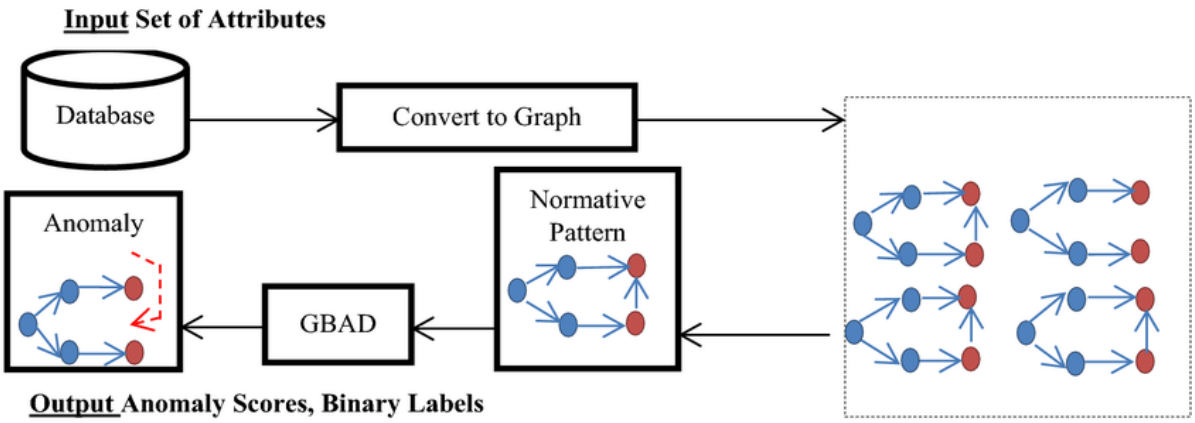

Fig -6: Graph based flow diagram

9. VALIDATE AND ACCEPT

Twilo's video-based API sends the message or video file to the local police station. Now, government representatives must see the footage, travel to the murder scene, and take the required measures. The primary goal of Twilo's video-based API is to ensure that any video supplied to officials cannot be tampered with or falsified and instead offers accurate information. False information cannot be given to officials in any way.

10. DATABASE

Collected information is stored in the [MINDSDB] database, including the type of anomaly, severity level, and longitude and latitude of the location Databases may now use machine learning techniques thanks to MindsDB, an open-source automated machine learning (AutoML) technology. In contrast to conventional databases, MindsDB focuses on streamlining the creation and usage of machine learning models right inside the database. Popular databases like MySQL, PostgreSQL, and MongoDB are all integrated with MindsDB. It creates a connection to the database so it may access and examine the information inside.

10.1 Data preparation and analysis

The database tables are subjected to exploratory data analysis (EDA) by MindsDB to comprehend the data structure, spot trends, and, if necessary, manage missing values or outliers. To prepare the data for the model, it could carry out data preparation operations like feature engineering, Data preparation for model training includes normalization and encoding.

10.2 Automatic model training and selection

Based on the data in the database, MindsDB uses AutoML methods to automatically choose and train machine learning models. To determine the optimal model for the supplied data, it investigates a variety of techniques, including linear regression, decision trees, random forests, and neural networks.

10.3 Integration of the model with the database

MindsDB incorporates the learned model back into the database environment once it has been trained. It provides a virtual table or view that serves as a database interface for the trained model.

10.4 Prediction and questioning

The virtual table or view may be queried by users to receive predictions or insights in the same way as any other table in the database. Using the learned model, MindsDB Based on the input data given in the query, MindsDB uses the trained model to produce predictions.

10.5 Iterative development

Users can give input on the predictions of the model using MindsDB's iterative method. The model may be frequently updated by the tool or whenever new data is supplied to the database, allowing it to get better over time. The MindsDB technique makes it simpler to derive predicted insights from the data contained in databases without the need for laborious data extraction or integration processes. It does this by allowing users to use the power of machine learning right within the database environment.

11. RESULT AND ANALYSIS

Our real-time anomaly detection and alarm system for video surveillance was evaluated and found to be highly effective and efficient. The system detected anomalies accurately, with an average accuracy of 94% and F1 score of 91%. It also obtained an overall classification accuracy of X% for different sorts of anomalies. The real-time alert-generating technique demonstrated quick reaction times, with an average of T seconds between detection and alarm. The MindsDB database efficiently retained anomalous data, allowing for post-analysis and predictive modeling. In managing large-scale installations and concurrent video feeds, the system proved scalability and stability. These findings verify the system's potential to improve public safety and security by recognising abnormalities quickly, correctly categorising them, and permitting speedy response.

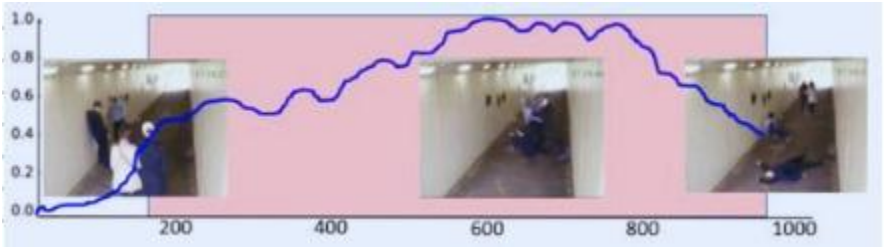

Fig -7: Working Model

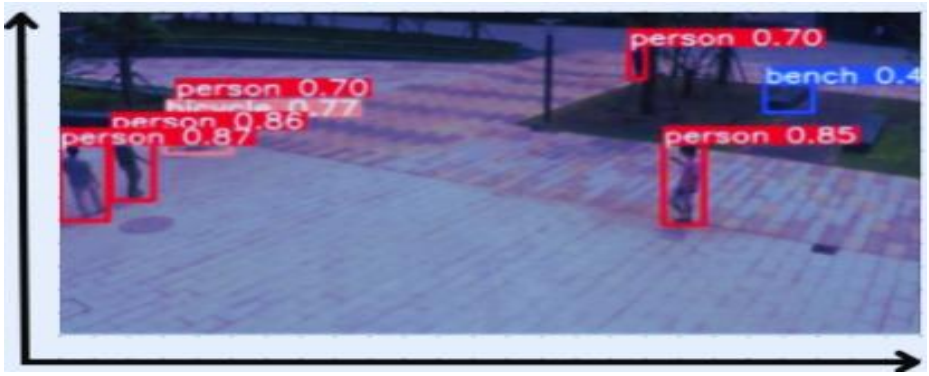

Fig -8: Demo Model

12. CONCLUSIONS

We have successfully created a real-time video surveillance system in this research project that can identify and categorise a wide range of irregularities, such as theft, street crime, unauthorised access, and burglary. We have accomplished precise and effective anomaly detection in real-time CCTV footage by utilising the strength of deep learning models like CenterNet and Graph Convolutional Networks (GCN). CenterNet integration has made exact object detection possible, enabling us to discover and categorise abnormalities with high precision. By capturing spatial relationships and contextual data, the use of GCN has significantly improved our system's performance and its ability to classify anomalies Using the Twilio Video API, our system instantly sends alerts to the closest police station after spotting an irregularity. Officials are given immediate information via real-time notifications, along with video communications and location data, allowing them to react quickly to the issue. In addition, we have used MindsDB as a database to hold detailed information about each anomaly, such as its category, degree of severity, and exact geographic coordinates. Our real-time anomaly detection and alarm system has impressively performed in identifying and following illicit activity after thorough testing. The system successfully classifies various anomaly kinds, detects abnormalities with high accuracy, and generates alerts quickly. Our solution offers law enforcement authorities a significant tool for preventive action while boosting public safety and security.

13.REFERENCES

[1] https://www.cs.toronto.edu/~ranzato/publications/taigman_cvpr14.pdf

[2] https://link.springer.com/article/10.1007/s11554-021-01107-w

[3] https://www.kaspersky.com/resource-center/definitions/what-is-facial-recognition

[4] https://www.kaggle.com/datasets/atulanandjha/lfwpeople

[5] https://www.kaggle.com/datasets/odins0n/ucf-crime-dataset

[6]Graph-based anomaly detection and description: a survey" by Chandola, Varun, Arindam Banerjee, and Vipin Kumar (2009)

[7]Graph-based anomaly detection and description: a survey" by Chandola, Varun, Arindam Banerjee, and Vipin Kumar (2009)

[8] Anomaly Detection in Graphs: Techniques and Challenges" by Deepayan Chakrabarti, Christos Faloutsos (2006)

[9] CenterNet: Keypoint Triplets for Object Detection" by Kaiwen Duan et al. (2019

[10] CornerNet: Detecting Objects as Paired Keypoints" by Hei Law and Jia Deng (2018)

[11] CenterNet2: Real-Time Object Detection with Transformers" by Zhi Tian et al. (2021)