Dr. A.P.J. Abdul Kalam Technical University, Lucknow

B.Tech. Program

Project report on

REAL TIME VIOLENCE DETECTION AND ALERT SYSTEM

Submitted by

Name	***Roll Number***
Prateek Kumar Rajput	***2000910310114***
Manjit Kumar Gautam	***2000910310094***
Yashaswani Srivastava	***2000910310197***

Under the Guidance of

Dr. Anuranjan Kansal
Assistant Professor

June, 2024

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
JSS ACADEMY OF TECHNICAL EDUCATION
C-20/1 SECTOR-62, NOIDA, UTTAR PRADESH.

Dr. A.P.J. Abdul Kalam Technical University, Lucknow

B.Tech. Program

Project report on

REAL TIME VIOLENCE DETECTION AND ALERT SYSTEM

Submitted by

Name	***Roll Number***
Prateek Kumar Rajput	***2000910310114***
Manjit Kumar Gautam	***2000910310094***
Yashaswani Srivastava	***2000910310197***

Under the Guidance of
Dr. Anuranjan Kansal
Assistant Professor

Submitted to the Department of Electronics & Communication Engineering in partial fulfillment of the requirements for the degree of Bachelor of Technology.

June, 2024

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
JSS ACADEMY OF TECHNICAL EDUCATION
C-20/1 SECTOR-62, NOIDA, UTTAR PRADESH.

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Prateek Kumar Rajput	***Manjit Kumar Gautam***	***Yashaswani Srivastava***
(2000910310114)	*(2000910310094)*	*(2000910310197)*

.

CERTIFICATE

This is to certify that Project Report entitled **"REAL TIME VIOLENCE DETECTION AND ALERT SYSTEM"** which is submitted by **PRATEEK KUMAR RAJPUT, MANJIT KUMAR GAUTAM** and **YASHASWANI SRIVASTAVA** for partial fulfillment of the requirement for the award of B. Tech degree in Electronics and Communication Engineering of **Dr. A.P.J. Abdul Kalam Technical University, Lucknow** is a record of the candidate own work carried out by him under my supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other degree.

Dr. Anuranjan Kansal
Assistant Professor

Date:

PLAGIARISM REPORT

CHAPTER Prateek Rajput

ORIGINALITY REPORT

10%	6%	4%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.arxiv-vanity.com Internet Source	1%
2	ejournal.unkhair.ac.id Internet Source	1%
3	Submitted to Saudi Electronic University Student Paper	1%
4	Submitted to Bournemouth University Student Paper	1%
5	www.ijraset.com Internet Source	<1%
6	www.coursehero.com Internet Source	<1%
7	Submitted to University of Bolton Student Paper	<1%
8	etheses.whiterose.ac.uk Internet Source	<1%
9	Submitted to Polytechnic Institute Australia Student Paper	<1%

ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to all those who have contributed to the successful completion of this project.

First and foremost, I extend my sincere thanks to my supervisor, Dr. Anuranjan Kansal, for his invaluable guidance, support, and encouragement throughout the duration of this project. His expertise, patience, and insightful feedback have been instrumental in shaping this endeavor.

I am also deeply indebted to the Head of the Department of Electronics and Communication Engineering, Dr. Arun Kumar G, for his constant encouragement and support. belief in my abilities and his willingness to provide resources and assistance have been indispensable.

I am grateful to the Principal, Vice-Principal & Registrar of JSS Academy of Technical Education, Noida, for providing an environment conducive to learning and innovation. Their vision and leadership have created opportunities for students to excel and grow.

Last but not least, I extend my thanks to the entire faculty and staff of the Electronics and Communication Engineering Department for their collective efforts in fostering an enriching academic environment.

I am also thankful to my family and friends for their unwavering support and encouragement throughout this journey.

Prateek Kumar Rajput	**Manjit Kumar Gautam**	**Yashaswani Srivastava**
(2000910310114)	(2000910310094)	(2000910310197)

ABSTRACT

Real-time violence detection and alert system (RT-VDAS) is a critical tool for preventing and responding to violence in a timely manner. As we can see, in our day-to-day life we hear news of violence taking place but actions are not taken on time to prevent the incidents. This leads to injuries of people involved in fights and people present in crowds may also get affected. Old methods of detecting violence through surveillance may not detect violence and crime in a timely manner. Therefore, there is a high need to revolutionize the way of surveillance. This research deals with this gap by introducing a *Real-Time Violent Detection and Alert System*, an advanced fusion of machine learning algorithms and smart alert algorithms formulated to strengthen public safety. The system uses computer vision to analyse video feeds from surveillance cameras to identify suspicious behaviour and potential threats. Once a threat is detected, the system can generate an alert to security personnel or law enforcement so that they can intervene quickly and effectively.

The system has the potential to revolutionize the way we prevent and respond to violence in a number of ways. Places such as transportation areas, grocery stores, shopping complexes, parks, entertainment areas. schools, colleges, parking areas, roads, etc, are always a target for violence. In schools, it can be used to detect violence. In public places, it can be used to detect suspicious individuals or groups, weapons, and other potential threats. And in homes it can be used to detect elder abuse, domestic violence, and other forms of violence against vulnerable populations. Therefore, the enforcement of effective real time violence detection will not only detect violence but also generate instant and speedy alerts for prompt action. Therefore, this system provides the significance of enhanced public safety, proactive risk reduction, reduced response time, integration with existing security infrastructure.

TABLE OF CONTENTS

LIST OF FIGURES

Page No.

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION OF PROJECT

In today's world, ensuring public safety has become a paramount concern, with the rising instances of violence and crime in various communities. In response to these challenges, innovative technological solutions have emerged, one of the most promising being real-time violence detection and alert systems. These systems leverage advanced technologies such as artificial intelligence (AI), machine learning (ML), computer vision, and sensor networks to detect and respond to violent incidents swiftly and effectively[1]. Real-time violence detection and alert systems operate on the principle of proactive surveillance, continuously monitoring public spaces, workplaces, educational institutions, and other areas prone to violence. These systems utilize a network of cameras, microphones, and other sensors to capture audiovisual data in real-time. Advanced algorithms analyse this data to identify patterns, anomalies, and potential threats indicative of violent behaviour.

One of the key components of these systems is computer vision[2], which enables the automated analysis of video feeds to detect violent actions such as physical altercations, assaults, and weapon use. Deep learning models trained on vast datasets learn to recognize specific gestures,movements, and interactions associated with violent behaviour, allowing for accurate and reliable detection. Additionally, audio analysis techniques[3] can be employed to identify sounds characteristic of violence, such as gunshots, screams, or breaking glass, further enhancing the system's capabilities. Upon detecting a potential threat or violent incident, the system triggers an immediate response, which may include activating alarms, notifying security personnel or law enforcement, and initiating emergency protocols. By providing real-time alerts, these systems enable rapid intervention, potentially preventing escalation and minimizing harm to individuals and property.

Moreover, real-time violence detection and alert systems can be integrated with existing security infrastructure, such as surveillance cameras, access control systems, and emergency response protocols, creating a comprehensive security ecosystem. This integration allows for seamless coordination between different security measures, enhancing overall effectiveness and

responsiveness. Beyond their role in incident detection and response, these systems also serve as valuable tools for post-incident analysis and forensic investigation. By recording and analyzing data before, during, and after a violent event, they provide valuable insights into the circumstances and dynamics of the incident, aiding law enforcement agencies in their investigations and legal proceeding[4].

1.1.1 Components of Real-Time Violence Detection and Alert Systems:

- Surveillance Infrastructure: Real-time violence detection and alert systems rely on a network of surveillance cameras, microphones, and other sensors strategically deployed in key locations. These sensors capture audiovisual data in real-time, providing a comprehensive view of the environment.
- Advanced Algorithms: Central to these systems are sophisticated algorithms capable of analyzing vast amounts of audiovisual data. Machine learning models, particularly deep learning algorithms, are trained on extensive datasets to recognize patterns, anomalies, and indicators of violent behavior.
- Computer Vision: Computer vision techniques enable automated analysis of video feeds, allowing the system to detect and interpret visual cues associated with violent actions. By recognizing gestures, movements, and interactions, computer vision algorithms can identify potential threats with high accuracy.
- Audio Analysis: In addition to visual cues, real-time violence detection systems also analyze audio data to detect sounds indicative of violence, such as gunshots, screams, or aggressive speech. Audio analysis algorithms can differentiate between normal background noise and potential threats, providing an additional layer of detection[5].
- Integration with Security Infrastructure: These systems can be seamlessly integrated with existing security infrastructure, including surveillance cameras, access control systems, and emergency response protocols. Integration enables coordinated responses and enhances overall security effectiveness.

1.1.2 Operation and Response Mechanisms:

- Continuous Monitoring: Real-time violence detection systems operate 24/7, continuously monitoring the environment for signs of potential threats or violent behavior.
- Immediate Alerting: Upon detecting a potential threat or violent incident, the system triggers immediate alerts, notifying security personnel, law enforcement, and other relevant parties.
- Emergency Protocols: The system may initiate emergency protocols, such as activating alarms, locking doors, or broadcasting emergency messages, to mitigate the risk and ensure the safety of individuals in the vicinity[6].
- Rapid Intervention: By providing real-time alerts and actionable intelligence, these systems enable rapid intervention, potentially preventing escalation and minimizing harm.

1.1.3 Benefits and Applications:

- Enhanced Public Safety: Real-time violence detection and alert systems contribute to enhanced public safety by enabling swift detection and response to violent incidents, reducing the risk of harm to individuals and property.
- Prevention and Deterrence: The proactive surveillance capabilities of these systems act as a deterrent against violent behavior, discouraging potential perpetrators and preventing incidents before they occur.
- Post-Incident Analysis: These systems facilitate post-incident analysis and forensic investigation by providing detailed audiovisual records of violent events, aiding law enforcement agencies in their investigations and legal proceedings.
- Comprehensive Security Ecosystem: By integrating with existing security infrastructure, real-time violence detection systems create a comprehensive security ecosystem that enhances overall security effectiveness and responsiveness[7].

1.1.4 Ethical and Privacy Considerations:

- Privacy Protection: Deployment of surveillance technologies raises concerns about individual privacy and surveillance. It is essential to implement robust privacy

protections, transparency measures, and legal safeguards to ensure responsible use and protect individual rights.

- Ethical Deployment: Responsible deployment of real-time violence detection systems requires careful consideration of ethical implications, including potential biases, discrimination, and misuse of data. Ethical guidelines and oversight mechanisms should be established to ensure ethical and accountable use.

CHAPTER 2
LITERATURE SURVEY

2.1 EFFICIENT SPATIO-TEMPORAL MODELING METHODS FOR REAL-TIME VIOLENCE RECOGNITION

The research paper delves into the realm of real-time violence recognition in videos, exploring various deep learning techniques and methodologies to enhance the efficiency and accuracy of violence detection systems. The authors discuss a wide array of methods, including non-local neural networks, global temporal attention mechanisms, lightweight network architectures, bidirectional convolutional LSTM, and 3D convolutional neural networks, among others, to address the challenges associated with violence recognition in video data. They introduce specialized datasets tailored for violence detection and evaluate the efficacy of different spatiotemporal modeling approaches, shedding light on the advancements and achievements in this domain. One of the key contributions of the paper is the proposed method that leverages channel averaging and 2D CNNs for real-time classification of violent actions in videos. Through a series of experiments, which encompass videos captured by moving cameras, the effectiveness of this method is demonstrated[8].

The paper also delves into the optimal time interval for frame grouping and the utilization of efficient 2D CNN architectures to enhance violence recognition accuracy. Visualizations using Grad-CAM are provided to underscore the significance of spatial locations in violence detection, offering insights into the inner workings of the proposed approach. Furthermore, the paper explores a range of techniques for efficient spatiotemporal modeling in real-time violence recognition, including strategies like pruning convolutional neural networks, low-rank approximations, and knowledge distillation. It also delves into the practical applications of these methods, such as face recognition in surveillance videos and crime prediction, showcasing the versatility and potential impact of the proposed approaches[9].

The research introduces spatiotemporal attention modules and frame-grouping as pivotal components for real-time violence detection in videos, emphasizing the use of lightweight CNN backbones to enhance efficiency and accuracy. Through a series of experiments, the proposed system showcases superior performance compared to existing methods across various violence

datasets, highlighting its efficacy in real-time surveillance applications with low latency requirements. The stability and effectiveness of the proposed approach are further validated through experiments on long-term violence videos, underscoring its robustness and reliability in practical scenarios. The paper also outlines an efficient violence detection pipeline tailored for real-time and on-device operations, with a primary focus on spatio-temporal modeling techniques. The pipeline incorporates frame-grouping for 2D CNNs, spatial and temporal attention modules, and achieves state-of-the-art performance on violence datasets. Notably, the proposed approach reduces computational complexity in comparison to traditional 3D-CNN methods, making it well-suited for surveillance.[10]

Research Gap - One research gap identified in the paper is the need for further exploration of data augmentation techniques to enhance the robustness of the model for real-time violence recognition in videos. This gap suggests an opportunity for future research to investigate and implement augmentation strategies that can improve the model's performance and generalization capabilities.

2.2 VIOLENCE DETECTION USING SPATIOTEMPORAL FEATURES WITH 3D CONVOLUTIONAL NEURAL NETWORK.

The paper introduces a novel three-staged framework for violence detection in surveillance videos utilizing Convolutional Neural Network (CNN) models. This method demonstrates high accuracy on benchmark datasets and surpasses existing techniques in violence detection tasks. The study also outlines future directions, including the implementation of the system on resource-constrained devices and the exploration of edge intelligence for violence recognition in the Internet of Things (IoT). Notably, the research received support from a grant provided by the Korean government, highlighting its significance and potential impact. Furthermore, the paper is part of a comprehensive collection of research papers focusing on violence detection in videos through computer vision techniques, particularly emphasizing deep learning models like CNNs and Long Short-Term Memory (LSTM) networks. Various approaches and algorithms are discussed within this context, showcasing the evolution and advancements in the field of violence detection in surveillance videos.

The study specifically leveraged 3D CNN to identify instances of violence in video clips sourced from diverse datasets. While the model exhibited exceptional accuracy on the violent

crowd dataset, it showed comparatively lower accuracy on datasets containing violence in movies and hockey fights. To enhance performance, the researchers conducted model optimization using the OPENVINO toolkit, aiming to improve the efficiency and effectiveness of the violence detection system. The findings suggest the necessity for further experiments and optimizations to enhance accuracy across all datasets, indicating a continuous effort towards refining the proposed method.

The proposed method for violence detection using spatiotemporal features with a 3D Convolutional Neural Network involves a multi-stage process. It begins with detecting individuals in a video stream using a MobileNet CNN model, followed by the extraction of spatiotemporal features from a sequence of frames using a 3D CNN model. The final stage involves classifying the activity using a Softmax classifier, ultimately aiming to enhance both accuracy and efficiency in violence detection within surveillance videos. The approach not only outperforms existing methods but also includes optimization for deployment using the OPENVINO toolkit, showcasing a comprehensive and advanced approach to violence detection in surveillance videos.[11]

Research Gap - The research paper on violence detection using spatiotemporal features with a 3D Convolutional Neural Network addresses the gap in the existing literature by proposing an advanced method that combines CNN models for accurate violence detection in surveillance videos. Future work could focus on enhancing the model's performance on datasets where it showed lower accuracy, such as violence in movies and hockey fights, through further optimization and fine-tuning of the 3D CNN architecture. Additionally, exploring real-time implementation of the method on edge devices for efficient surveillance systems could be a promising direction for future research.

2.3 VIOLENT VIDEO DETECTION BASED ON SEMANTIC CORRESPONDENCE

Automatic detection of violent videos has broad application prospects in many fields such as video surveillance and movie grading. However, most existing violent video detection models based on multimodal feature fusion ignore the fact that the audio-visual data in the same violent video may not semantically correspond. Blindly fusing non-corresponding features is not beneficial even potentially harmful to models. In this paper, a novel violent video detection model based on semantic correspondence between audio-visual data is proposed from the same

video. Deep neural networks are used to extract features of three different modalities: appearance, motion, and audio. After that, the feature-level fusion strategy to fuse these multimodal features via shared subspace learning is chosen. Semantic correspondence is used to guide this process through multitask learning and semantic embedding learning.

In the field of violent video detection, appearance and motion features are crucial and complementary. As most videos contain information of both visual and auditory modalities, more and more researchers adapt audio features to assist detection of violent videos. Early works mainly focus on hand-crafted features. The most commonly used appearance descriptors include: scale invariant feature transform (SIFT), histograms of oriented gradients (HOG), etc. Space-time interest points (STIP) and improved dense trajectories(iDT) are commonly used motion features. As for auditory feature, Mel-scale frequency cepstral coefficient (MFCC) is the most used descriptor. After feature extraction, a classifier is attached to get the final score. In recent years, some researchers apply deep neural networks to violent video detection, which is used for not only feature extraction but also feature fusion and classification. There are two most common structures in visual feature extraction: two-stream networks based on 2D ConvNet and networks based on 3D ConvNet. Besides, long short-term memory network (LSTM) is used to capture long-term dependency in the time domain sometimes. The main contribution of this paper is the proposal of a violent video detection model based on semantic correspondence. The proposed scheme fuses multimodal features through shared subspace learning. Semantic correspondence information is added to further improve the performance.

First, three typical video features: appearance, motion, and audio, are extracted and fused based on deep learning methods. Semantic correspondence information in feature fusion stage to eliminate interference of non-corresponding data is introduced further. The correspondence information is exploited in two ways to help optimize shared subspace learning: (i) multitask learning, (ii) semantic embedding learning. After semantic correspondence information is added, the performance of our model has been improved. Its effectiveness has been proved both on the public Violent Scene Detection 2015 dataset (VSD2015) and our self-built Violence Correspondence Detection dataset (VCD). The results show that the model achieves quite competitive results on both. The research paper offers a comprehensive exploration of violent video detection methodologies, highlighting the significance of semantic correspondence, shared subspace learning, and multimodal feature fusion in enhancing detection accuracy[12].

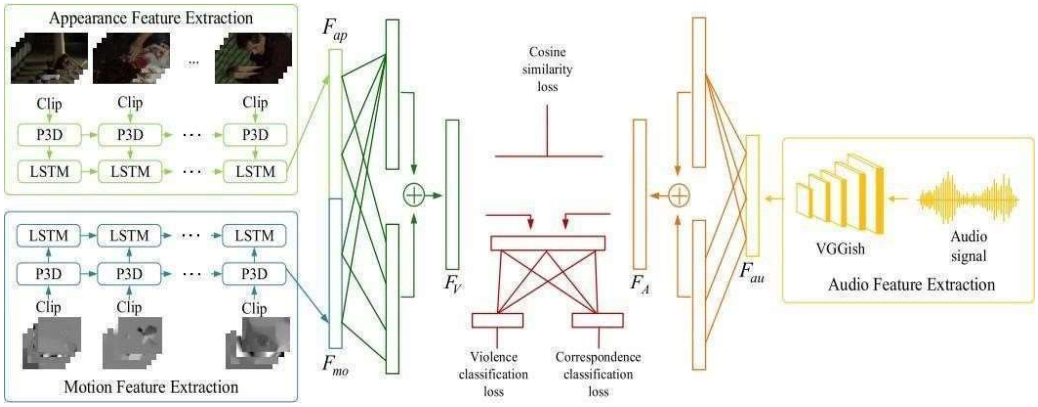

Figure 2.1: The structure of this violent video detection model

Research Gap - In consideration that most violent videos contain obvious emotional information, emotion information in videos to assist violent video detection should be utilized in future work which should focus on scalability, efficiency, robustness, and ethical considerations for further improvements.

2.4 VISION-BASED FIGHT DETECTION FROM SURVEILLANCE CAMERAS

Vision-based action recognition is one of the most challenging research topics of computer vision and pattern recognition. A specific application of it, namely, detecting fights from surveillance cameras in public areas, prisons, etc., is desired to quickly get under control these violent incidents. In this study, we focus on the fight activity. A fight event is defined as two or more people, who are fighting to a degree that must be interfered. Related approaches consist of two parts as feature extraction and classification. Mainly two different approaches are applied for feature extraction: computing optical flow information of the videos and computing deep convolutional neural networks-based representations. Due to the proven success of convolutional neural networks (CNN) in various computer vision applications, CNN based approaches are highly preferred in recent works. Long Short-term Memories (LSTM) are used

for modeling the temporal information, as they find out relationships between the consecutive frames through their memory ability. In summary, CNN + LSTM network is commonly used in action recognition due its high performance[13].

In this study, in order to enhance the CNN + LSTM based approach for the fight detection task, a modified Xception CNN is trained using the fight scenes. Thus, it is expected that this CNN is more familiar with the input sequences and extracts more relevant features from them. In the classification layer, a novel approach is developed by using Bidirectional LSTM (Bi-LSTM) along with a self-attention layer to improve the performance. Furthermore, a new surveillance camera fight dataset is collected.

The experiments are conducted for each three datasets: Hockey, Peliculas, and surveillance camera dataset. For feature extraction part, VGG16 and Xception architectures are tested. In addition, a modified Xception architecture is trained using the fight scenes from Hockey dataset and named as Fight-CNN. For the classification part, regular LSTMs and Bi-LSTMs are tested along with VGG16 and Xception models. Also, the network is augmented by attention layer, which are tested by Xception and Fight-CNN. For each CNN, two classifiers which are Bi-LSTM with attention or Bi-LSTM without attention, are considered. In CNN and LSTM experiments, to observe the effect of number of frames to the accuracy, frame numbers are changed between 5 and 10. Number of epochs is 20, batch size is 10 for Fight-CNN experiments and 100 for VGG16 and Xception experiments. Datasets are split as 80% for training and 20% for testing[14].

The proposed method has benefited from the CNNs for feature extraction from frames. Two-way learning of bi-directional LSTMs and the attention layers that can also determine the amount of given attention to each part of the sequence are found to improve the accuracy. As a result, proposed method has surpassed the state-of-the-art performance. Additionally, a new model is tested by using Fight-CNN, a modified version of Xception model. Bi-LSTMs show better performance than regular LSTMs in action recognition. The attention layer improves the performance of sequence learning. This study validates this finding and shows that using Bi-LSTM together with attention is a promising solution to classify fight scenes.

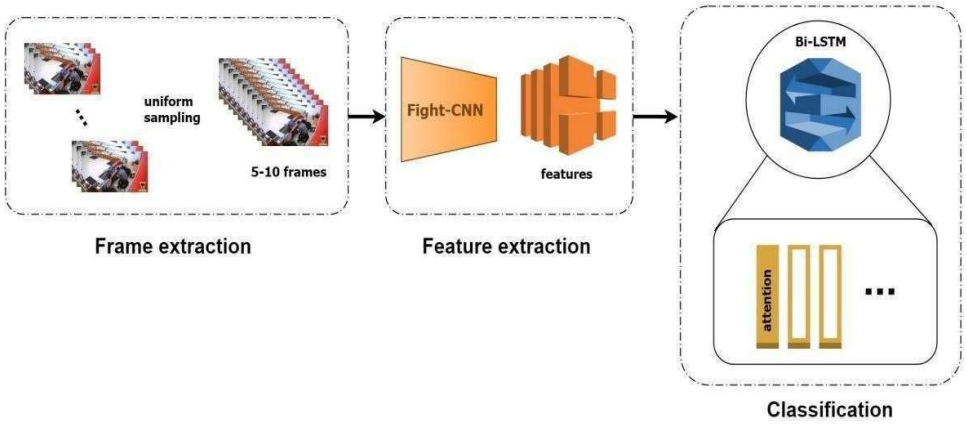

Figure 2.2: Overview of the proposed system

Research Gap - Surveillance camera fight dataset is the important contribution of the study which presents further challenges for automatic fight detection. This surveillance camera dataset can be extended by adding new samples from security camera footages on streets or underground stations.

2.5 CHANNEL-WISE ATTENTION IN 3D CONVOLUTIONAL NETWORKS FOR VIOLENCE DETECTION

With the widespread application of video surveillance (Skynet system) and public safety issues, a simple and efficient method for detecting violent behavior of video streams is necessary. The high latitude of the video data, the noise and the intertwined operation of various events make it difficult to represent and model, and the violence is also highly relevant to the background. . In the pre-deep learning era, most jobs detect violent behavior based on the frame difference between two frames of video streams. The principle is that general violent behaviors are accompanied by strong movements, and frame difference can describe such movements well. Using this feature to list different formulas can represent normal or violent behavior. For example, optical flow method, trajectory method and so on. The accuracy of these methods can be guaranteed to be high, but their method has a limitation on speed. Because of too many image data, it is a time-consuming process to calculate the frame difference, which makes the model too slow and cannot achieve real-time detection. Moreover, these methods do not have good portability and usually work only on a single dataset[15].

When computer vision enters the era of deep learning, the way to extract features is automatically transferred from manual design to network structure. 2D Convolutional Neural Networks (2DCNNs) are very successful in image classification, such as AlexNet, VGGNet , and GoogLeNet . However, when the 2DCNNs are adopted for the violent detection, the accuracy is degraded because the time feature cannot be extracted. Recent efforts find out that, C3D can be adopted to identify the temporal and spatial characteristics of video, and the speed is extremely fast. However, the accuracy adopted in violent behavior recognition is still not as good as the optical flow method.

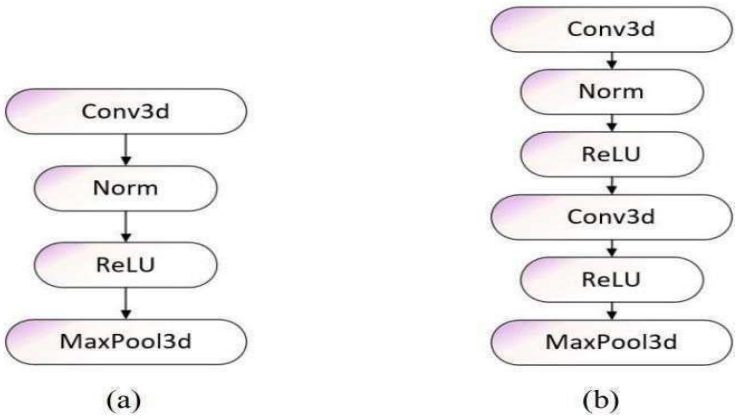

Figure 2.3: a) is first type of C3D structure and b) is second type of C3D structure

It is well known that behaviors in video stream (normal or violent) occur mostly in local regions, but it takes too much time to extract frame differences for all video streams. Based on this, we improved the accuracy of the C3D model from the Attention mechanism. We present a SELayer-C3D violence detection model in video streams, which mainly consists of SELayer and two types C3D model. It aims at ensuring computational efficiency without losing detection accuracy. The model obtains the accuracy of 99.0% and 98.08% on the Hockey dataset and the Crowd dataset. The key contributions of this paper are two-fold:

- Channel attention mechanism into the field of violence detection is introduced.
- In the model, 2D-SELayer is adapted to 3D domain in the process of detecting violence. The channel features obtained by C3D are weighted according to their importance by SELayer. Compared with other models, our model improves the calculation efficiency and satisfies the actual demands with higher accuracy.

Squeeze-and-Excitation Networks (SENet) is a Channel-wise attention. The core idea of SENet is to learn the feature weights according to the loss through the network, so that the effective feature map weight is significant, invalid or the effect of the small feature map weight training mode to achieve better results[16].

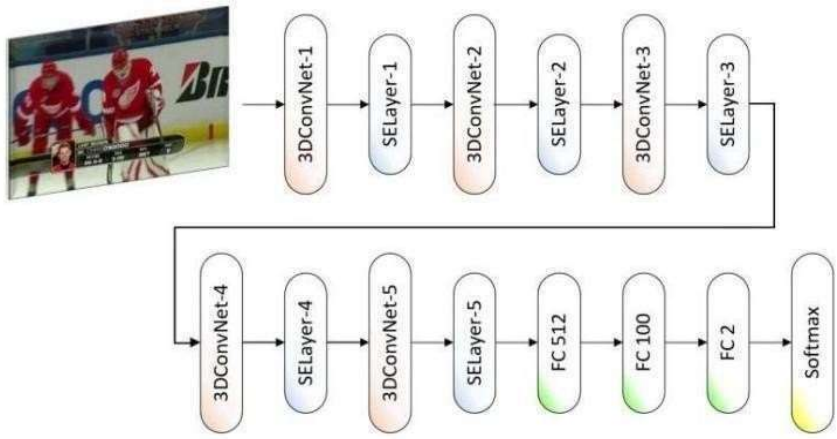

Figure 2.4: SELayer-C3D model, where 3dConvNet1-3 is the first type of convolution structure, and 3DConvNet4-5 is the second type of volume and structure.

Research Gap - In consideration of the diversity of violent behaviors and the complexity of simultaneous identification, Residual Neural Networks can be adopted to construct the characteristics of deep network structures, and to extend the model to simultaneously identify multiple violent behaviors.

CHAPTER 3

PROPOSED ALGORITHMS AND METHODOLOGY

3.1 PROBLEM STATEMENT

Design and develop a technological solution based on live CCTV feeds, that can automatically detect incidents related to street crime, violence, burglary, unauthorized access etc. and generate alerts to the concerned Authority.

The solution should also be able to generate a report and maintain a database that includes the nature of incident/crime, location, time, level of alert (i.e., low, medium, high-risk alert).

3.2 OBJECTIVES OF PROJECT

- To develop a Machine learning model which can detect any type of violence using raw CCTV footages, and achieve the best possible accuracy.
- To implement an alert system which can generate a warning of violence.
- To make a dashboard where the Alerts and Violence Records can be accessed.

3.3 USE CASE DIAGRAM

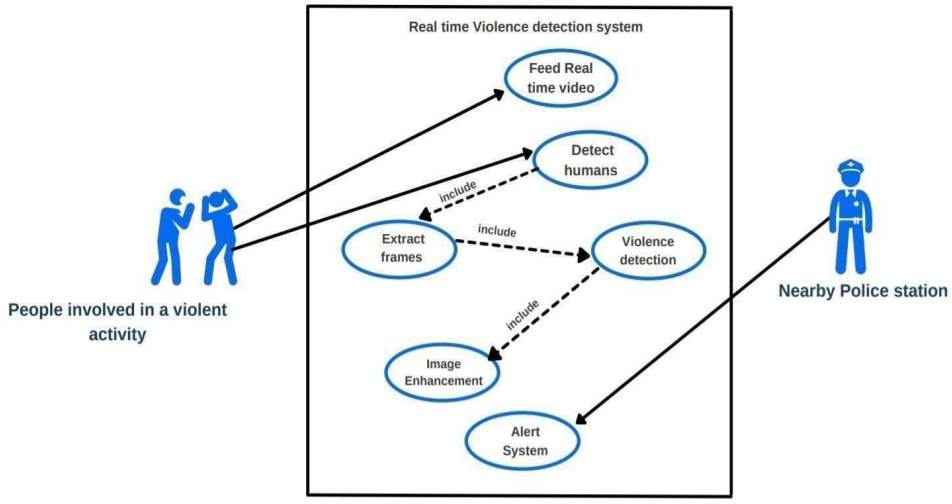

Figure 3.1 Use Case diagram

The rectangle box helps to define the scope of the proposed architecture. Anything happens within the rectangle happens within the system. There are 2 actors in this scenario, People involved in a violent activity and a nearby police station. People involved in a violent activity are primary actors as their actions will be detected by a real-time system. Use cases are represented by oval shape and they represent an action that accomplishes some sort of task within the system. Whenever a real-time video is given as input, the system will try to detect humans. Every time humans are detected the given included use cases are executed as well. They are Frames Extraction, Violence Detection, and Image Enhancement. After these three steps alert system will send an alert to a nearby police station[17].

3.4 DATASET USED

A diverse dataset of video footage encompassing various types of violence and non-violent activities was taken. Violence videos in the dataset contain many real fights situations in several environments and conditions. Non-violence videos from our dataset are collected from many different human actions like sports, eating, walking etc. Many of these videos are of CCTV footage. In original data set, there were 2000 videos given, 1000 for violence, and 1000 for non-violence and data was balanced. Due to storage issue, we have used 1000 videos for the model, which contains equally violent and non-violent videos. Raw surveillance video dataset is collected and sliced into clips. The average duration of the video clips is 5 sec. For training, 350 videos each from the violent and non-violent classes are taken at each epoch.

Also, we have used a self-made dataset of videos according to Indian conditions. These videos are tested against the model and the results are found to be true. Initially it contains 50 videos of fight and non-fight scenarios. Each video clip is 5 second long. Further, we may include more videos in this dataset, including various conditions.

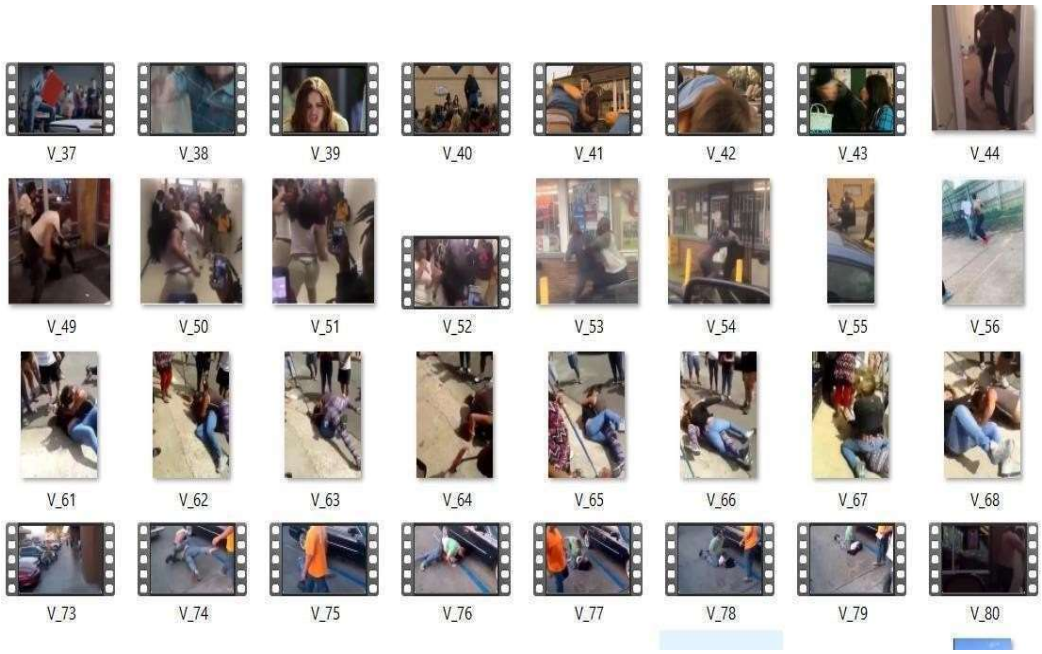

Figure 3.2 Video Clips from Violence Dataset

3.5 ARCHITECTURE - MobileNetV2

MobileNetV2 is a significant advancement in the field of convolutional neural networks (CNNs), specifically designed to address the challenges of deploying deep learning models on resource-constrained devices such as mobile phones, embedded systems, and IoT devices. Introduced by Google researchers in 2018, MobileNetV2 builds upon the success of its predecessor, MobileNet, by introducing novel architectural enhancements and optimization techniques aimed at improving both performance and efficiency[1].

3.5.1 Understanding MobileNetV2 Architecture:

At the core of MobileNetV2[1] design philosophy lies the concept of depth-wise separable convolutions, which enable significant reductions in computational complexity without sacrificing model accuracy. This architectural innovation decomposes standard convolutions into two separate operations: depth-wise convolutions and pointwise convolutions[7]. By doing so, MobileNetV2 effectively reduces the number of parameters and computations required, making it highly efficient for deployment on mobile and embedded devices.

In addition to depth-wise separable convolutions, MobileNetV2 introduces inverted residual blocks with linear bottlenecks, a key design element aimed at capturing rich features while minimizing computational cost. Inverted residuals consist of lightweight linear bottleneck layers followed by non-linear expansion layers, providing a compact yet expressive representation of the input data. Linear bottlenecks further enhance information flow through the network, ensuring efficient utilization of computational resources.

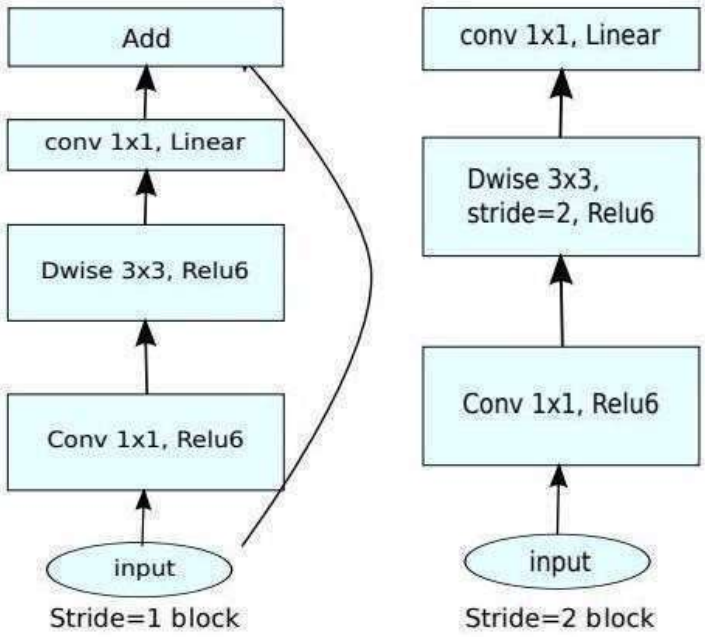

Figure 3.3: MobileNetV2 Architecture

The MobileNet architecture is primarily based on depth wise separable convolution, in which factors a traditional convolution into a depth wise convolution followed by a pointwise convolution.

The module presents a residual cell (has a residual/identity connection) with strideof 1, and a resizing cell with a stride of 2. From Figure 3.3, 'conv' is a normal con- volution, 'Dwise' is a depth wise separable convolution, 'Relu6' is a ReLu activation function with a magnitude limitation, and 'Linear' is the use of the linear function. Some of its features are:

- In MobileNetV2, there are two types of blocks. One is residual block with stride of 1. Another one is block with stride of 2 for downsizing.
- There are 3 layers for both types of blocks.
- The first layer is 1×1 convolution with ReLU6.
- The second layer is the depth-wise convolution.

- The third layer is another 1×1 convolution but without any non-linearity. It is claimed that if ReLU is used again, the deep networks only have the power of a linear classifier on the non-zero volume part of the output domain.

3.5.2 Efficiency and Performance Trade-offs:

MobileNetV2 achieves a delicate balance between efficiency and performance, making it a versatile choice for a wide range of computer vision tasks. By leveraging lightweight architectural components and optimization techniques, MobileNetV2 offers state-of-the-art performance on tasks such as image classification, object detection, and semantic segmentation, while maintaining low inference latency and minimal memory footprint.

Moreover, MobileNetV2 provides various model variants and customization options to suit different application requirements and constraints. For instance, depth multiplier and width multiplier parameters allow users to adjust the model's depth and width, trading off between accuracy and computational efficiency based on specific deployment scenarios.

3.5.3 Applications and Impact:

The impact of MobileNetV2 extends beyond academic research, with widespread adoption in real-world applications across diverse domains. In the field of mobile and embedded vision, MobileNetV2 has empowered developers to build intelligent applications and services that were previously inaccessible due to computational constraints. From mobile image classification apps to on-device object detection systems and smart IoT devices, MobileNetV2 has enabled a new wave of innovation, democratizing access to advanced AI capabilities.

Furthermore, MobileNetV2 has contributed to the democratization of AI by enabling on-device inference, reducing reliance on cloud-based services and addressing privacy and security concerns associated with data transmission and storage. By bringing AI capabilities directly to edge devices, MobileNetV2 opens up new possibilities for personalized experiences, enhanced privacy, and real-time decision-making in diverse settings, from smart homes and wearable devices to autonomous vehicles and industrial IoT applications.

3.5.4 Challenges and Future Directions:

While MobileNetV2 represents a significant advancement in lightweight CNN architectures[2], several challenges remain to be addressed. One key challenge is the optimization of model inference speed and energy efficiency for real-time applications with strict latency and power constraints. Additionally, ensuring robustness and generalization across diverse datasets and environments remains a critical research area.

Looking ahead, future directions for MobileNetV2 and similar architectures include exploring techniques for network compression, quantization, and model distillation to further reduce model size and computational complexity. Additionally, continued advancements in hardware acceleration technologies, such as specialized neural processing units (NPUs) and efficient parallel computing architectures, will play a crucial role in unlocking the full potential of MobileNetV2 for a wide range of edge computing applications.

3.6 WORKING/METHODOLOGY

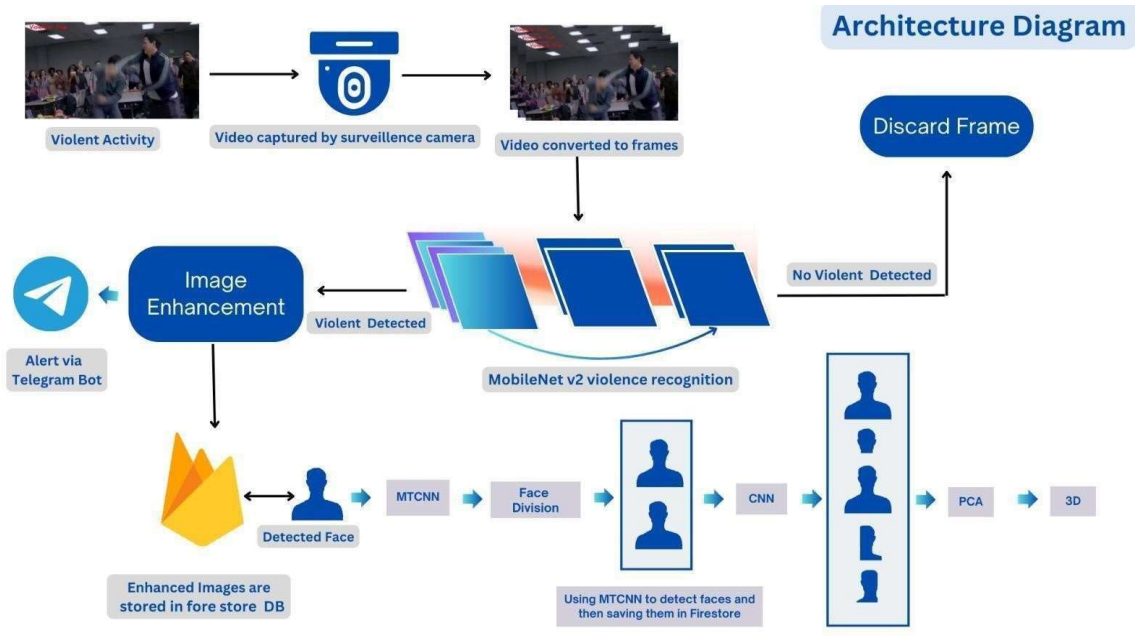

Figure 3.4: Project Flow Diagram

Initially, dataset containing 1000 videos is chosen. 350 videos of violence and 350 videos of non-violence is taken for training. Raw surveillance video dataset is collected and sliced into clips. The average duration of the video clips is 5 sec. For each epoch 350 videos from the

violence class and 350 videos from the non-violence are trained. The model is trained using MobileNetV2 using this dataset.

Now as per the flow diagram real-time video footage is given as input to CCTV. The footage is captured by the surveillance camera. After this footage is converted into frames and given to the model. Since our model is trained before using MobileNetV2 for violence recognition, the model detects the frames which contain violence and the frames which do not contain violence. The frame in which violence is not detected is discarded. The frames in which violence is detected goes for image enhancement.

Image Enhancement is performed on the frames that are obtained as output. This is performed using the inbuilt functions provided by the Python Imaging Library (PIL). PIL offers extensive file format support, efficient presentation and fairly powerful image processing capabilities. The Core Image Library is designed to provide quick access to data stored in several major pixel formats. It provides a solid foundation for common image processing tools. The brightness andcolor of the obtained output frames is increased by a factor of 2.

The **Alert Module** sends alert message to the specified authority. Figure 3.5 describes the architecture of the implemented alert system. When a frame is detected as true for violence, the system initializes a counter variable to one. Then it checks the subsequent 30 frames, whether if they too have violence detected true. The counter is incremented at each consecutive frame that is true for violence. If a frame is false for violence, the counter variable is set to 0 and starts checking the consecutive frame respectively checking whether violence is recognized. On the other hand, if the violence is detected true for the 30 consecutive frames, the current time is obtained using an inbuilt python function and an alert is sent to a Telegram group that consists officials of higher authorities. The Alert message comprises of an image of the detected violent activity, current timestamp, and the location where the camera is placed.

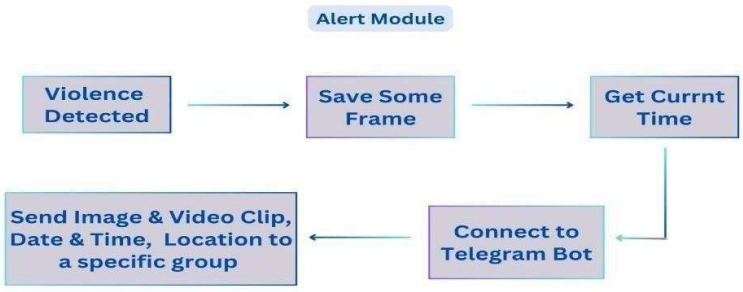

Figure 3.5: Architecture Diagram of Alert System

Figure 3.6 shows the alert message that is sent to the telegram group by the telegram bot. The concerned authorities can view the alert and take necessary actions.

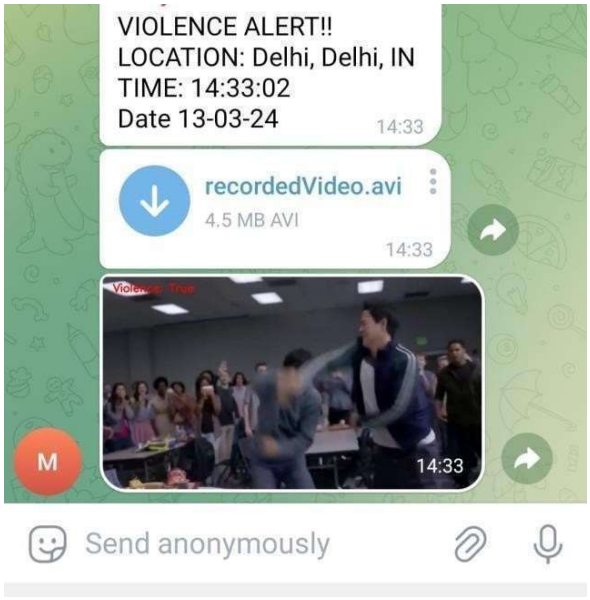

Figure 3.6: Screenshot of alert message

After image enhancement, enhanced images are stored in Firestore database. Face detection is performed using MTCNN and detected faces are stored back in the firestore database. MTCNN and Pyplot are used for face detection. MTCNN[4] consists of 3 stages of CNN for face detectionand face alignment. Pyplot is a submodule of the matplot library.

3.7 OPERATING ENVIRONMENT

3.7.1 Jupyter:

Jupyter is an open-source web application designed to facilitate interactive computing and data exploration. It offers users a versatile environment where they can write and execute code in various programming languages such as Python, R, and Julia. The platform's interactive nature allows for iterative development and experimentation, with code organized into cells for easy execution and modification. Additionally, Jupyter provides support

for Markdown syntax, enabling users to create formatted text, headings, and lists within their documents. This feature, coupled with Jupyter's rich output capabilities, allows users togenerate interactive visualizations, plots, and tables, enhancing their ability to communicate insights effectively. Furthermore, Jupyter seamlessly integrates with external libraries and tools, enabling users to access additional functionality for data analysis and computation tasks.Overall, Jupyter serves as a powerful tool for researchers, data scientists, educators, and professionals across various domains, empowering them to explore, analyze, and visualize datainteractively.

3.7.2 Language used (Python):

Python is a versatile and widely-used programming language known for its simplicity, readability, and flexibility. Developed by Guido van Rossum and first released in 1991, Python has since grown in popularity and become one of the most popular programming languages worldwide. Its straightforward syntax and dynamic typing make it accessible to beginners while its extensive standard library and robust ecosystem of third-party packages cater to advanced users across diverse domains. Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming, making it suitable for a wide range of applications—from web development and data analysis to machine learning and scientific computing. Its interpreted nature allows for rapid prototyping and development, while its cross-platform compatibility ensures seamless deployment across various operating systems. Python's vibrant community, comprehensive documentation, and active development ensure its continued relevance and adaptability to evolving technological trends and challenges. Overall, Python remains a powerful and versatile programming language, valued for its simplicity, productivity, and broad applicability in the ever-expanding world of technology.

3.7.3 Firestore:

Firestore is a fully managed NoSQL document database provided by Google Cloud Platform (GCP), designed to store, sync, and query data for web and mobile applications. It offers a flexible and scalable solution for managing structured data, enabling developers to build powerful and interactive applications with ease. At its core, Firestore utilizes a document-oriented data model, where data is organized into collections and documents. Each document is a JSON-like object that contains key-value pairs, known as fields, representing the data.

Documents within a collection can have different structures, allowing for versatile data storage without the need for a fixed schema. This flexibility makes Firestore well-suited for managing diverse datasets, ranging from simple user profiles to complex product catalogs and messaging systems.

One of the key features of Firestore is its real-time data synchronization capability. Changes made to data in the database are automatically propagated to all connected clients in real-time, allowing applications to instantly reflect updates without the need for manual refreshes. This real-time sync enables developers to build collaborative applications, such as chat platforms, collaborative editing tools, and real-time analytics dashboards, where multiple users can interact with the same data concurrently. In addition to real-time sync, Firestore offers built-in offline support, allowing client applications to access and modify data even when offline. Changes made offline are stored locally on the device and automatically synchronized with the server once connectivity is restored. This feature ensures that applications remain functional and responsive, regardless of network availability, and is particularly valuable for mobile applications that may experience intermittent connectivity.

Firestore is designed to scale effortlessly to handle large volumes of data and high traffic loads. It automatically handles data sharding, replication, and load balancing behind the scenes, ensuring high availability, low latency, and consistent performance. This scalability makes Firestore suitable for both small-scale applications and large enterprise systems, providing developers with the flexibility to start small and scale as their needs grow. Furthermore, Firestore integrates seamlessly with other Google Cloud Platform services, such as Firebase Authentication, Cloud Functions, and Cloud Storage, enabling developers to build end-to-end solutions using a unified set of tools and services. This integration simplifies development and deployment processes, allowing developers to focus on building innovative features and delivering exceptional user experiences.

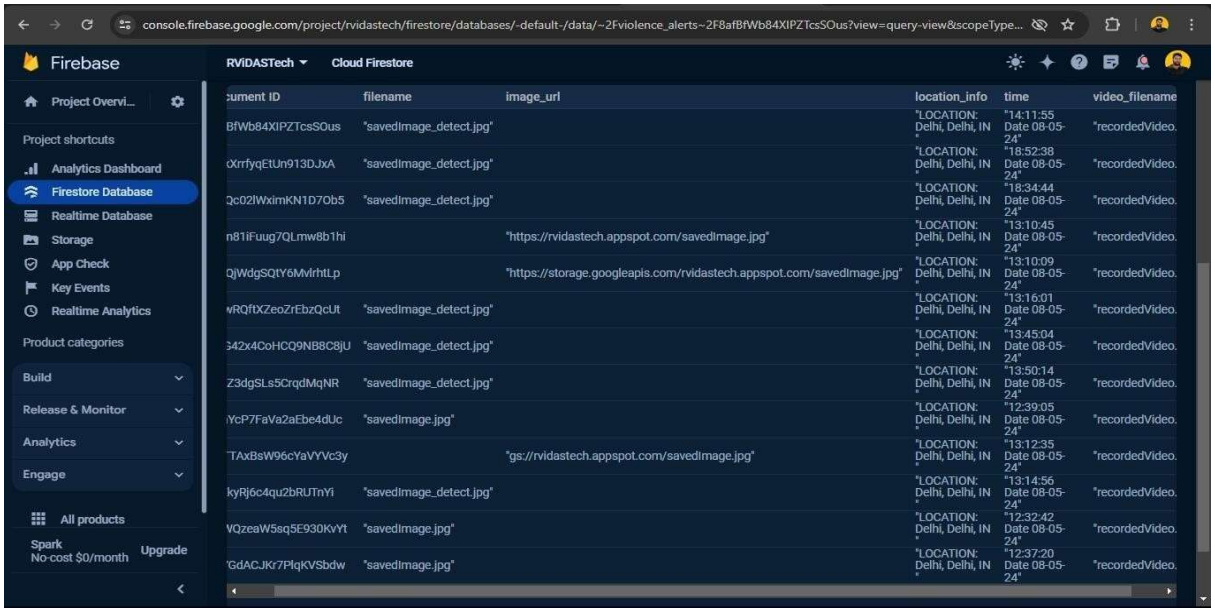

Figure 3.7: Real-time violence activity database

3.7.4 Web Monitoring:

Creating a real-time web monitoring dashboard involves developing an interactive interface for displaying monitoring data and ensuring that it updates instantly with the latest information. By using Firestore's real-time synchronization features along with Bootstrap for the front-end, developers can build a visually appealing and responsive dashboard without the need for more complex JavaScript frameworks like React.

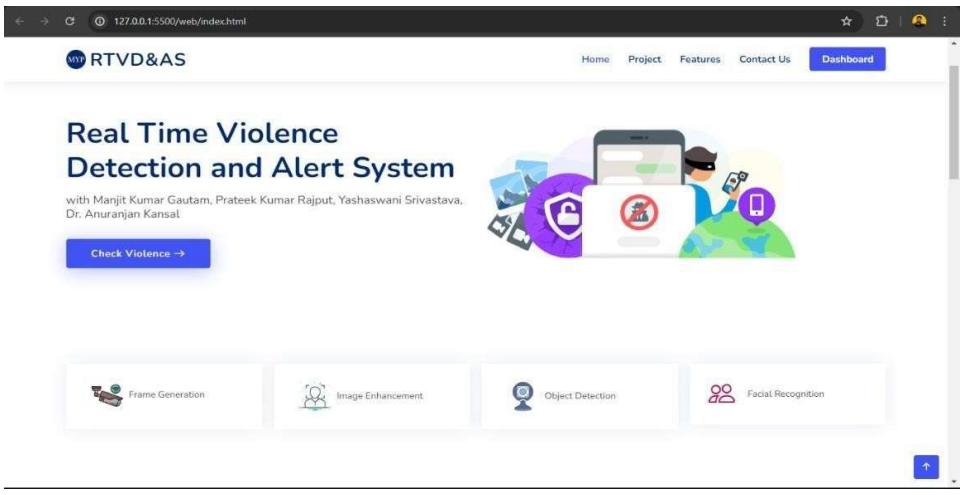

Figure 3.8: Activity Tracking Web Interface

CHAPTER 4
RESULTS AND DISCUSSION

In this section testing and training accuracy are displayed in the below given graphical representation. Figure 4.1 displays the training and testing accuracy and loss for the MobileNetV2 model when a dataset containing 1000 videos of average duration 5 seconds is given as input. For each epoch 350 videos from the violence class and 350 videos from the non-violence are trained. 95.2% accuracy was obtained on training and a respective accuracy of 94.7% was obtained when a CCTV footage that was not included in the dataset was given for testing.

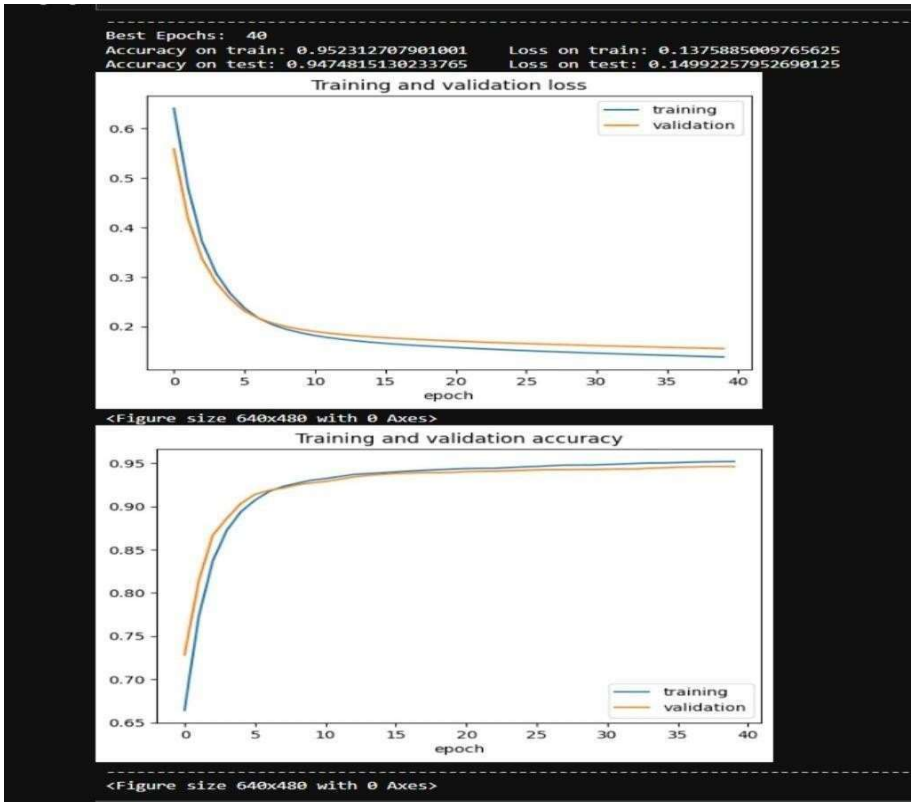

Figure 4.1: Accuracy and Loss of Training Set

Figure 4.2 shows confusion matrix for our model. It contains the obtained confusion matrix and other evaluation parameters. A confusion matrix is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. It allows visualization of the performance of an algorithm by displaying the number

of true positives, false positives, true negatives, and false negatives.

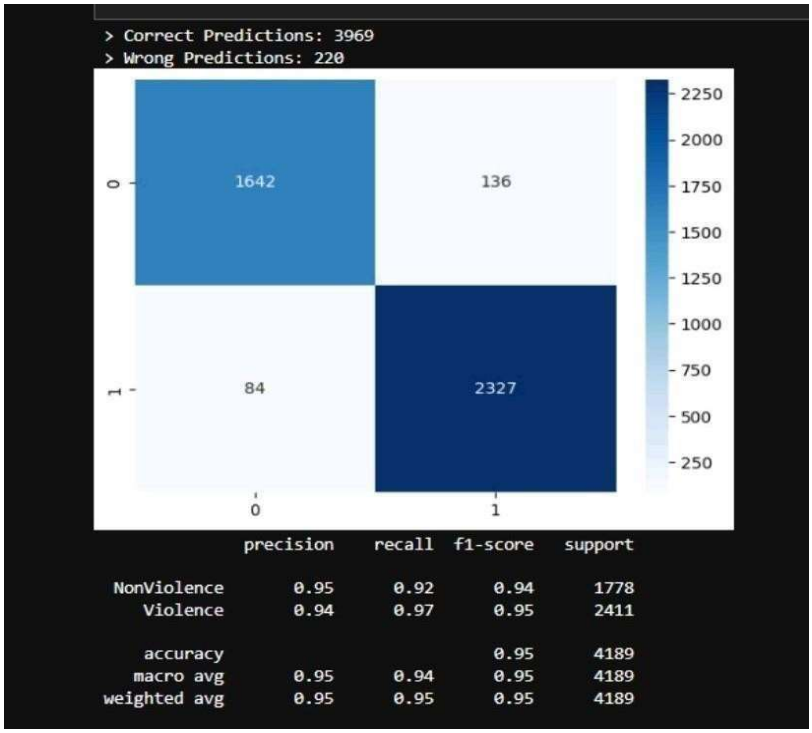

Figure 4.2: Confusion Matrix of Trained Model

In the graph in figure 4.3 shows the ROC curve. The ROC curve shows how good a model is at distinguishing between different groups. It plots how often the model correctly identifies positive cases without wrongly labeling negative cases as positive. A higher curve means a better model. The accuracy and loss come to a constant level of increment and decrement after approximately 5 epochs.

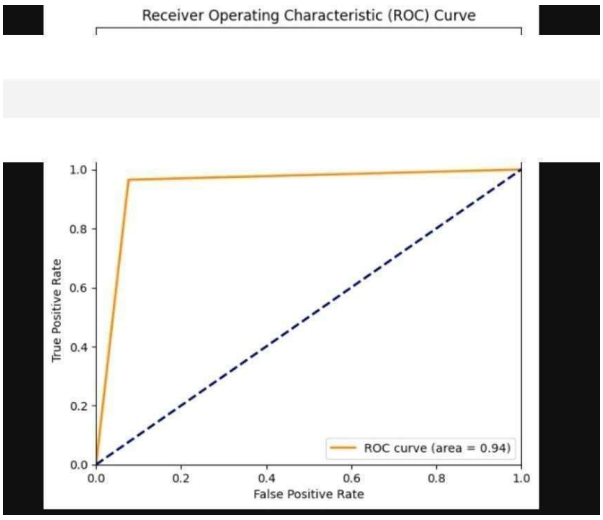

Figure 4.3 Receiver Operating Characteristic (ROC) Curve

A video with violence is given as input to the system. Figure 4.4 and figure 4.5 shows one frame in the videothat was labeled to have violent activity. Another video clip without violent activity was given as input.

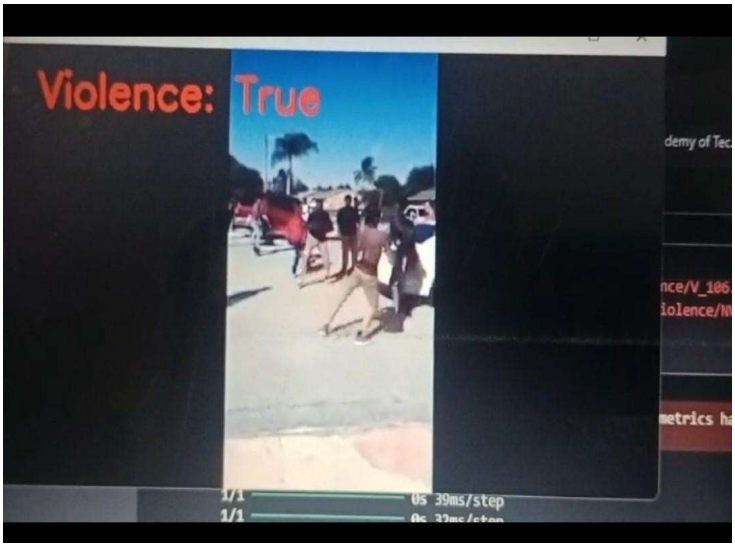

Figure 4.4: Output frame that recognized violence

Figure 4.5 Output frame that recognized violence

Figure 4.6 shows one frame of that video which is rightly labelled as false for violence. When an input containing non-violent activity was given to the model then it detected it correctly and displayed false for violence in output.

Figure 4.6: Output frame that did not recognize violence

CHAPTER 5
CONCLUSION AND FUTURE WORK

Violence scene detection in real-time is a challenging problem due to the diverse con-tent and large variations quality. In this research, we use the MobileNetV2 model to offer an innovative and efficient technique for identifying violent events in real-time surveillance footage. The proposed network has a good recognition accuracy in typical benchmark datasets, indicating thatit can learn discriminative motion saliency maps successfully. It is also computationally efficient, making it ideal for use in time-critical applications and low-end devices. Here, we had also shown the working of an alert system that is integrated with the pretrained model. In comparison to other state-of- the-art approaches, this methodology will give a far superior option.

Future Work - This system presents a significant advancement in real-time violence detection, contributing to enhanced security measures. Its proactive approach allows for timely intervention in potential threats, safeguarding public safety and reducing response times. In future work this model could be upgraded to work in multiple cameras connected to a single network simultaneously. Also, integration of audio and other sensor data to further improve detection accuracy and context awareness can be done. As mentioned before, distinguishing between mirthful conduct and genuine threats is also a challenge. Utilization of emotion information in videos to assist violent video detection is important. Additionally, it is important to ensure that the detection model is used in a responsible and ethical manner, and that privacy concerns are adequately addressed. Enhanced feature extraction for exploring and incorporating additional features into the deep learning model for more comprehensive violence detection can be added.

APPENDIX
MACHINE LEARNING

Machine learning is a transformative field within artificial intelligence (AI) that empowers computers to learn from data and improve their performance over time without explicit programming. At its core, machine learning algorithms analyze vast amounts of data to identify patterns, extract insights, and make predictions or decisions. This process mimics the way humans learn, allowing machines to adapt and evolve based on experience. Datasets serve as the foundation upon which machine learning models are trained, validated, and tested. These datasets may consist of structured or unstructured data, including text, images, audio, and numerical values. Through the process of training, machine learning algorithms iteratively adjust their parameters to minimize errors and optimize performance on the given task.

Machine learning techniques can be broadly categorized into three main types: supervised learning, unsupervised learning, and reinforcement learning. In supervised learning, algorithms learn from labeled data, where each example is paired with a corresponding target or output. This type of learning is commonly used for tasks such as classification and regression. Unsupervised learning involves learning from unlabeled data to uncover hidden patterns or structures, often used for clustering and dimensionality reduction. Reinforcement learning focuses on training agents to make sequential decisions through trial and error, guided by feedback from the environment. The applications of machine learning span across various industries and domains, including healthcare, finance, marketing, e-commerce, transportation, and more. In healthcare, machine learning models assist in disease diagnosis, personalized treatment planning, and drug discovery. In finance, algorithms drive algorithmic trading, fraud detection, and risk assessment. In marketing, recommendation systems leverage machine learning to deliver personalized product recommendations, optimize advertising campaigns, and enhance customer experiences.

SOME IMPORTANT TERMS:

Deep Learning - Deep Learning (DL) is a subset of machine learning focused on training artificial neural networks with many layers (deep architectures) to automatically learn hierarchical representations of data directly from raw input.

It has gained importance for its ability to solve complex problems such as image and speech recognition, better than traditional machine learning algorithms. DL requires large amounts of labeled data and significant computational resources for training, but it excels at tasks involving high-dimensional data like images, text, or audio.

Model - In machine learning, a model is a mathematical representation or algorithm that is trained on data to make predictions, decisions, or identify patterns. Essentially, it's a way of capturing the underlying relationships between input data and output predictions.

Vector - A vector is a fundamental data structure that represents a list of numerical values arranged in a specific order. Vectors are commonly used to represent features, observations, or target variables in datasets.

Training - Training refers to the process of teaching a machine learning model to make predictions or decisions based on input data. It involves exposing the model to a dataset containing examples of input-output pairs, known as the training data, and adjusting the model's parameters or internal representations to minimize the difference between its predictions and the actual outputs.

Validation - Validation refers to the process of assessing the performance of a trained model on a separate dataset, known as the validation dataset, to ensure that it generalizes well to unseen data. Validation is an essential step in the machine learning workflow as it helps prevent overfitting and provides insights into the model's ability to generalize to new, unseen examples.

Overfitting - Overfitting is a common problem in machine learning where a model learns to capture noise or random fluctuations in the training data instead of the underlying patterns or relationships. This results in a model that performs well on the training data but fails to generalize to new, unseen examples.

Underfitting - Underfitting is the opposite of overfitting and occurs when a machine learning model is too simplistic to capture the underlying patterns in the data. It results in a model that performs poorly on both the training data and new, unseen examples because it fails to learn the relationships between the input features and the target variable adequately.

Epochs - Epochs refer to the number of times the entire dataset is passed forward and backward through the neural network during the training process. Each epoch consists of one forward pass (where the input data is fed into the network and predictions are made) followed by one backward pass (where the model's parameters are updated based on the calculated loss).

CNN - Convolutional Neural Network, is a type of deep learning model commonly used for analyzing visual imagery. It's inspired by how the human visual system works and consists of layers that automatically learn to recognize patterns in images. CNNs are great for tasks like image classification, object detection, and facial recognition because they can effectively capture spatial hierarchies and local patterns in images.

MTCNN - Multi-Task Cascaded Convolutional Neural Network. It's a specific type of deep learning model used for face detection tasks. MTCNN is designed to detect faces in images with high accuracy by utilizing a cascade of three neural networks: one for finding potential face regions, another for refining those regions, and a third for accurately determining facial landmarks.

ReLU - Rectified Linear Unit is an activation function used in neural networks. ReLU sets negative values to zero and leaves positive values unchanged. This helps the network learn faster and reduces the likelihood of the vanishing gradient problem. ReLU is widely used in deep learning models because of its simplicity, efficiency, and ability to handle complex data effectively.

Image Enhancements - It includes preprocessing techniques to improve quality or utility for tasks like classification or object detection. Examples include contrast-enhanced, noise-reduced, or sharpened images. These datasets aid in training models to better handle diverse image conditions.

Convolution - A mathematical operation that combines two sets of data to produce a third set. In the context of image processing or deep learning, convolution involves sliding a small matrix (called a kernel or filter) over an input image. At each position, the kernel multiplies with the overlapping pixel values in the image, and the results are summed to produce a single output value. This operation helps extract features from the image, like edges or textures, which are crucial for tasks such as image recognition or object detection.

Dataset - A dataset is a collection of data points used to train, validate, or test a model. It typically

consists of multiple instances, both positive and negative, where each instance represents an observation or example. Depending on the problem domain and the type of learning task, they can range from structured data like numerical or categorical features in tabular format to unstructured data like images, text, or audio.

LSTM (Long Short-Term Memory) - A type of neural network that's great for understanding and making predictions based on sequences of data, like words in a sentence or stock prices over time. It's good at remembering important things from the past while learning new stuff, which makes it useful in tasks like predicting future trends or understanding language.

CCTV (Closed Circuit Tele-Vision) - The output format of a CCTV system typically consists of video footage captured by the cameras. This footage is usually stored in digital format, commonly using video compression techniques like H.264 or H.265 to reduce file size while maintaining quality. The footage is often stored on digital video recorders (DVRs) or network video recorders (NVRs) for later retrieval and analysis.

LIBRARIES:

Tensorflow - TensorFlow is a powerful open-source machine learning framework developed by Google, widely used for building, training, and deploying machine learning models, especially deep learning models. It offers both high-level APIs, like Keras, for easy model development, and low-level APIs for more advanced customization.

Keras - Keras is a user-friendly neural networks API that runs on top of TensorFlow It simplifies model building, provides a user-friendly interface for creating neural networks with minimal code. Keras allows developers to quickly prototype and experiment with different model architectures.

Matplotlib - Matplotlib is a comprehensive library for creating static, interactive, and animated visualizations in Python. It is widely used for generating plots, charts, histograms, and other types of graphical representations of data.

Pyplot - Pyplot is a module within the Matplotlib library that provides a MATLAB-like interface for creating static, interactive, and animated visualizations in Python. It is commonly used for generating plots, charts, histograms, and other types of graphical representations of data.

PIL (Python Imaging Library) - PIL (Python Imaging Library) is a library in Python that provides extensive capabilities for opening, manipulating, and saving different images file formats. It allows for a wide range of image processing tasks, including resizing, cropping, rotating, filtering, enhancing, and converting between different image formats.

OpenCV – OpenCV (Open Source Computer Vision Library) is an open-source computer vision and machine learning software library. It provides a wide range of functionalities for image and video processing, including object detection, recognition, tracking, segmentation, and more.

TOOLS:

Jupyter - Jupyter is an open-source web-based interactive computing platform that allows users to create and share documents containing live code, equations, visualizations, and narrative text. It supports various programming languages, including Python, R, Julia, and Scala, but is primarily used with Python through the IPython kernel.

Telegram Bot - A Telegram bot is a special type of account on the Telegram messaging platform that is operated by software, rather than by a human user. Activated using a group chat ID, these bots can interact with users, send messages, provide information, perform tasks, and respond to commands automatically. When a frame is flagged as indicating violence, the counter is activated. If 30 consecutive frames are identified as violent, the event is classified as such, triggering an alert. The alert is transmitted via a bot using the messaging app - Telegram, including a warning, location, time, date, as well as a captured image and short video clip of the incident.

Firebase - Firebase is a platform developed by Google that provides a wide range of services for building and managing mobile and web applications. It is a NoSQL cloud database, which offers features like authentication, real-time database, cloud storage, hosting, machine learning, and analytics, among others. Firebase enables developers to focus on building high-quality apps without worrying about managing infrastructure, making it a popular choice for startups and mobile app developers.

RESULT PARAMETERS:

Confusion Matrix - A confusion matrix is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. It allows visualization of the performance of an algorithm by displaying the number of true positives, false positives, true

negatives, and false negatives.

Training loss - It measures how well the model is performing on the training data during the training process.
It is calculated as the difference between the model's predictions and the actual target values (labels) for the training examples. The goal during training is to minimize the training loss by adjusting the model's parameters (weights and biases) using optimization algorithms like gradient descent.
A decreasing training loss indicates that the model is learning and improving its ability to fit the training data.

Validation loss - Like the training loss, the validation loss measures the difference between the model's predictions and the actual target values for the validation examples. Monitoring the validation loss helps detect overfitting, where the model performs well on the training data but poorly on new, unseen data.
Ideally, both the training loss and validation loss should decrease during training. However, if the validation loss starts to increase while the training loss continues to decrease, it may indicate that the model is overfitting to the training data.

Training accuracy - Training accuracy is a metric used to evaluate the performance of a machine learning model during the training process. It measures the percentage of correctly predicted examples (or instances) in the training dataset out of the total number of examples in the dataset.

Validation accuracy - Validation accuracy is a crucial metric for assessing a model's generalization ability and selecting the best-performing model. It complements training accuracy by providing insights into how well the model performs on new, unseen data.

ROC Curve - The ROC curve shows how good a model is at distinguishing between different groups. It plots how often the model correctly identifies positive cases without wrongly labeling negative cases as positive. A higher curve means a better model. The area under the curve (AUC) quantifies overall performance, with higher AUC indicating better discrimination. Optimal models have ROC curves that hug the upper left corner, indicating high sensitivity and low false positive rate.

PAPER PUBLISHED

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue III Mar 2024- Available at www.ijraset.com

Real Time Violence Detection and Alert System

Manjit Kumar Gautam[1], Prateek Kumar Rajput[2], Yashaswani Srivastava[3], Dr. Anuranjan Kansal[4]
Electronics and Communication Engineering JSS Academy of Technical Education, Noida India

Abstract: ***This paper talks about a précised safety model incorporating elements of Artificial Intelligence for real-time violence detection along with Alert System. This model utilizes the advancements in technology to rapidly respond to any potential violent assault incident. When these technologies are further developed, it also increases the possibilities of how they could be used in future to safeguard schools, public area safety, personal safety, and social stability. Numerous researches and trials have been conducted to counter violence with the passage of time that includes the installations of surveillance systems for warning or alerting to violent activities. Its main objective is to get surveillance systems to automatically annotate the violence activities and issue any Warning/Alerts. For this purpose, first of all the system proceeds with the process of foregrounding a person in each frame, then relevant frames are extracted and irrelevant are ignored, after this violent pattern is identified by the trained model is detected, and end up saving these frames as images. The image is then enhanced and the corresponding details like time and location are transmitted as an alert via Telegram app. The proposed technique is essentially based on deep learning for automatic violence detection using Convolutional Neural networks (CNN). For this diagnosis, a light weight pre-trained model, MobileNetV2, is used to ensure better accuracy than an independent CNN which requires massive computation time and reduced precision.***

Keywords: Real-time violence detection, Alert systems, Surveillance, Convolutional Neural Network (CNN), Image enhancement, MobileNetV2.

I. INTRODUCTION

In our lives we often witness incidents of violence occurring without preventive measures being taken. This leads to injuries, for those involved in conflicts and even bystanders in crowds can be impacted. Traditional surveillance methods may not always catch criminal activities in a manner. That's why it's crucial to update surveillance techniques to keep up with the times. This research project aims to fill this gap by introducing a Time Violent Detection and Alert System, which combines machine learning algorithms and smart alert systems to enhance safety. By using computer vision technology the system analyzes video feeds from security cameras to identify behaviors and potential threats. When a threat is detected the system can quickly alert security personnel or law enforcement for action.

Recent advancements in learning have proven effective in extracting temporal features from videos capturing both movement and detailed spatial information across frames. This study specifically focuses on implementing a Real Time violence alert system utilizing MobileNetv2. After processing the model outputs the enhanced frames are stored in the Firebase database. Face detection is performed using MTCNN and Pyplot on these images. The frames along with details such as the time and location of an incident are then sent as an alert to the police station through the systems module.

In this paper's context, an in-depth into real-time violence detection systems is showcased, covering technologies, uses, challenges, and moral considerations. Through an examination of existing research and emerging patterns in this field, the goal of this publication is to enhance the understanding of the capabilities of real-time violence detection systems in enhancing public safety and security. Furthermore, it aims to address the complex ethical and social issues associated with their deployment.

II. MOTIVATION

The main motivation behind this project is to develop a system that can address the issue of Violent behavior in public places.

Violence is a social problem that has devastating effects over society. Real-time detection of violence is challenging due to the need to go through huge amounts of surveillance video data from various locations and different camera devices, for which violence activity may last for just a few seconds. So, Intelligent Video Surveillance methods are required to detect violence and prevent criminal activities accurately as well as in Real-time. This will aid in quick decision making. Recent researches and studies have also highlighted the accuracy of deep learning approaches to violence detection.

REFERENCES

[1] Zhang Y, Li Y, Guo S (2022) Lightweight mobile network for real-time violence recognition. PLoS ONE 17(10): e0276939. doi: 10.1371/journal.pone.0276939. PMID: 36315496; PMCID: PMC9621415.

[2] M. -S. Kang, R. -H. Park and H. -M. Park, "Efficient Spatio-Temporal Model- ing Methods for Real-Time Violence Recognition," in IEEE Access, vol. 9, pp. 76270-76285, 2021, doi: 10.1109/ACCESS.2021.3083273.

[3] Ullah FUM, Ullah A, Muhammad K, Haq IU, Baik SW. Violence Detection Using Spatiotemporal Features with 3D Convolutional Neural Network. Sensors (Basel). 2019 May 30;19(11):2472. doi: 10.3390/s19112472.

[4] J. C. Vieira, A. Sartori, S. F. Stefenon, F. L. Perez, G. S. de Jesus and V. R. Q. Leithart, "Low-Cost CNN for Automatic Violence Recognition on Embedded System," in IEEE Access, vol. 10, pp. 25190-25202, 2022, doi: 10.1109/AC- CESS.2022.3155123.

[4] P. Sernani, N. Falcinelli, S. Tomassini, P. Contardo and A. F. Dragoni, "Deep Learning for Automatic Violence Detection: Tests on the AIRTLab Dataset," in IEEE Access, vol. 9, pp. 160580-160595, 2021, doi: 10.1109/AC- CESS.2021.3131315.

[5] C. Gu, X. Wu and S. Wang, "Violent Video Detection Based on Semantic Correspondence," in IEEE Access, vol. 8,pp.85958-85967,2020,doi:10.1109/ACCESS.2020.2992617. -May 2020.

[6] Ş. Aktı, G. A. Tataroğlu and H. K. Ekenel, "Vision-based Fight Detection from Surveillance Cameras," 2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA), Istanbul, Turkey, 2019, pp. 1-6, doi: 10.1109/IPTA.2019.8936070. - February 2020.

[7] B. Jiang, F. Xu, W. Tu and C. Yang, "Channel-wise Attention in 3D Convolutional Networks for Violence Detection," 2019 International Conference on Intelligent Computing and its Emerging Applications (ICEA), Tainan, Taiwan, 2019, pp. 59-64, doi: 10.1109/ICEA.2019.8858306. -August 2019.

[8] K. Singh, K. Yamini Preethi, K. Vineeth Sai and C. N. Modi, "Designing an Efficient Framework for Violence Detection in Sensitive Areas using Computer Vision and Machine Learning Techniques," 2018 Tenth International Conference on Advanced Computing (ICoAC), Chennai, India, 2018, pp. 74-79, doi: 10.1109/ICoAC44903.2018.8939110.

[9] Hassner T, Itcher Y, Kliper-Gross O. Violent flows: Real-time detection of violent crowd behavior. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. IEEE; 2012. p. 1–6.

[10] Kim HD, Ahn SS, Kim KH, Choi JS. Single-channel particular voice activity detection for monitoring the violence situations. In: 2013 IEEE RO-MAN. IEEE; 2013. p. 412–417.

[11] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, T. Darrell, and K. Saenko, ‘‘Long-term recurrent convolutional networks for visual recognition and description,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 2625–2634.

[12] J. Li, X. Liu, W. Zhang, M. Zhang, J. Song, and N. Sebe, ‘‘Spatio-temporal attention networks for action recognition and detection,’’ IEEE Trans. Multimedia, vol. 22, no. 11, pp. 2990–3001, Nov. 2020.

[13] Nievas, E.B.; Suarez, O.D.; García, G.B.; Sukthankar, R. Violence detection in video using computer vision techniques. In Proceedings of the International Conference on Computer Analysis of Images and Patterns, Seville, Spain, 29–31 August 2011; pp. 332–339.

[14] Hassner, T.; Itcher, Y.; Kliper-Gross, O. Violent flows: Real-time detection of violent crowd behavior. In Proceedings of the 2012 IEEE Computer Society Conference on Computer

Vision and Pattern Recognition Workshops (CVPRW), Providence, RI, USA, 16–21 June 2012; pp. 1–6.

[15] De Souza, F.D.; Chavez, G.C.; do Valle Jr, E.A.; Araújo, A.d.A. Violence detection in video using spatio-temporal features. In Proceedings of the 2010 23rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Gramado, Brazil, 30 August–3 September 2010; pp. 224–230.

[16] S. Ji, W. Xu, M. Yang and K. Yu, "3D convolutional neural networks for human action recognition", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221-231, Jan. 2013.

[17] A. Datta, M. Shah and N. Da. Vitoria Lobo, "Person-on-person violence detection in video data" in Object recognition supported by user interaction for service robots, Quebec, pp. 433-438, 2002