Violence Detection using Posture Analysis

Bipin Gowda[1], Abhishek J[2], Jairaj P[3], Nikhil H Y[4], Akshay Pai[5], Jebah Jaykumar[6]

[1-6]Department of Computer Science & Engineering, BNM Institute of Technology, Bengaluru - 560070

---***---

Abstract -- God's eye is a video surveillance system that primarily uses machine learning and deep learning algorithms to identify violent people among a group of people, primarily based on posture. Machine learning is an application of Artificial Intelligence. It is a technology that focuses on the development of computer programs that access data and uses it to learn for themselves. This system makes use of supervised learning to classify individuals as violent based on machine learning and deep learning algorithms.

CCTV cameras have been around for a long time and the usual protocol for any crime that is committed is to review the footage and identify the perpetrator. God's eye automates this process to cut down the human intervention required and to detect any violent activity happening in real-time using algorithms built into the surveillance system using computer vision libraries. This system can also be deployed on drones and serve a diverse range of functionalities. Drone systems have been deployed by various law enforcement agencies to monitor hostiles, spy on foreign drug cartels, conduct border control operations, etc. The violence detection system first identifies humans from images taken from the video feed. The image region with the human is then analyzed and an outline of the posture is generated. The orientations between the limbs of the estimated pose are next used to identify violent individuals. The system can detect violent individuals from real-time drone footage using cloud processing technology.

I. INTRODUCTION

Surveillance cameras are increasingly being used in public places like streets, intersections, banks, and shopping malls to increase public safety. However, the monitoring capability of law enforcement agencies has not kept pace. The result is that there is a glaring deficiency in the utilization of surveillance cameras and an unworkable ratio of cameras to human monitors. One critical task in video surveillance is detecting anomalous events such as traffic accidents, crimes, violence, or illegal activities. Generally, anomalous events rarely occur as compared to normal activities. But when they do occur it is not immediately identified. The current practice is to review the video footage and identify the criminal at a later point in time. Therefore, to alleviate the wastage of labor and time, developing intelligent machine learning algorithms for automating video surveillance is a pressing need. The goal of a practical violence detection system is to accurately identify violent individuals among groups of people primarily based on their posture. The posture of any individual can be easily captured. It is impossible to mask your posture and hence it is a sure-shot way to produce more accurate results compared to the existing emotion detection systems. Therefore, violence detection can be considered as a coarse level of video understanding which filters out violent individuals from normal individuals. Once a violent person is detected, the necessary action can be taken to suppress them.

Real-world events are complicated and diverse. A small step towards addressing violence is to develop algorithms that cleverly identify posture and then further classify it as violent or not based on supervised machine learning algorithms. Supervised machine learning is a technique where we teach the program what postures are to be identified as violent and what postures to be identified as non-violent using training data which consists of classified images of people. As we train the program with many images it will learn and develop a hypothesis or a remembrance net and then use this to correctly classify images. It is difficult to list all of the possible violent postures. Therefore, it is desirable that the violence detection algorithm takes into account all the violent postures and clearly distinguishes between violent and non-violent. Sports activities and dance moves will present challenges and have to be cleverly handled so that the system isn't tricked into perceiving these individuals as violent ones.

II. LITERATURE CITED

A. *Eye in the Sky: Real-time Drone Surveillance System (DSS) for Violent Individuals Identification using ScatterNet Hybrid Deep Learning Network*

Amarjot Singh et al. (2018), have proposed a real-time Drone Surveillance System (DSS) framework that can detect one or more individuals engaged in violent activities from aerial images. The framework first uses the Feature Pyramid Network (FPN) network to detect humans after which the proposed Scatternet Hybrid Deep Learning (SHDL) network is used to estimate the pose of humans. The estimated poses are used by the Support Vector Machine (SVM) to identify violent individuals. The utilization of fewer labeled examples for pose estimation is beneficial for this application as it is expensive to collect annotated examples. This paper also introduced the Aerial Violent Individual (AVI) Dataset which can benefit other researchers aiming to use deep learning for aerial surveillance applications. The proposed DSS framework outperforms the state-of-the-art technology on the AVI dataset. This framework will be instrumental in detecting individuals engaged in violent activities in public areas or large gatherings.

B. Human Posture Recognition Using Skeleton and Depth Information

Bo Cao et al. (2018), have presented an approach to efficiently recognize human posture with a multi-classified support vector machine (SVM). This paper presents an approach to efficiently recognize human posture which is widely applied to virtual reality, video surveillance, human-computer interaction, health care, and so on. Features of human posture are extracted from skeleton and depth information using the interpolation algorithm. A multi-classified SVM is used to recognize posture using the features. The results of experiments they conducted show that the approach meets the application requirements with high accuracy of 97.9% and a high average operating speed of 0.483 ms.

C. Violence Detection in Video Using Spatio-Temporal Features

Fillipe D. M de Souza et al. (2017), have built a violence detector on the concept of visual codebooks using linear support vector machines. It differs from the existing works of violence detection in the concern of data representation, as none have considered local Spatio-temporal features with bags of visual words. An evaluation of the importance of local Spatio-temporal features for characterizing the multimedia content is conducted through the cross-validation method. The results obtained confirm that motion patterns are crucial to distinguish violence from regular activities in comparison with visual descriptors that rely solely on the space domain.

III. PROPOSED SYSTEM

Step 1: A Web User Interface is provided where the user can Upload, Preview, and Generate Output for a selected video.

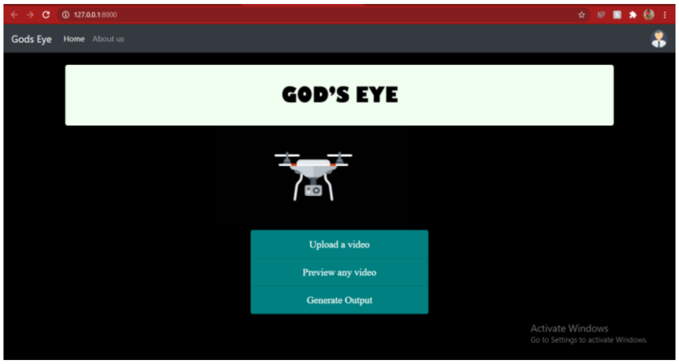


Figure 3.1: UI Landing Page

Step 2: The python code finds out the video fps and extracts a specified number of frames in a randomized order and saves them. Then the frames are ordered and are modified to be of a specified shape and size.

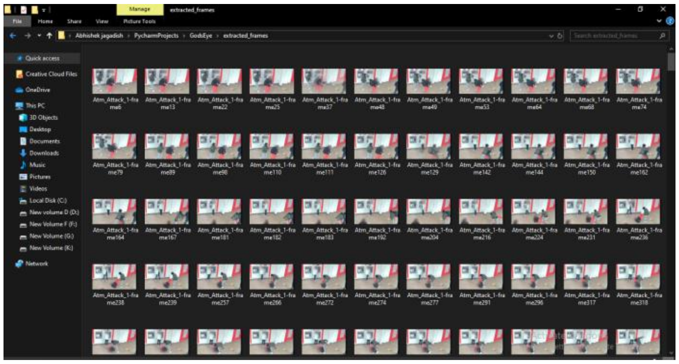


Figure 3.2: Video Segmentation

Step 3: Individual key point features such as right-shoulder, right-knee, etc. are identified along with the corresponding human that these key points belong to using the COCO model. Specified Key point features of each human are connected by lines of a different color for each unique identified human.

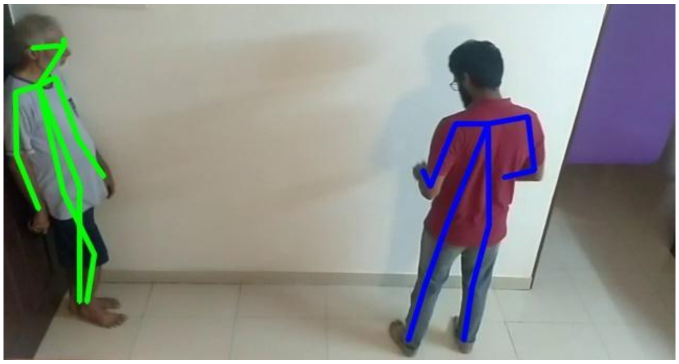
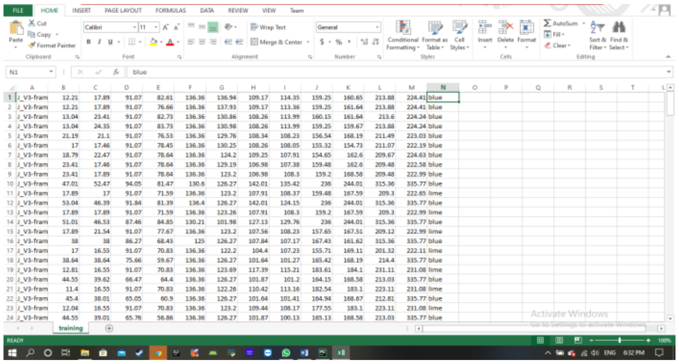


Figure 3.3: Stick Figurines

Step 4: The distance of the coordinates of each identified human's specified key points from the selected neck key point along with other parameters such as frame name, the color of lines, Non-violent, or Violent frames are converted to a vector.



N1: blue

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	J_V3-fram	12.21	17.89	91.07	82.61	136.36	136.94	109.17	114.35	159.25	160.65	213.88	224.41	blue
2	J_V3-fram	12.21	17.89	91.07	76.66	136.36	137.93	109.17	113.36	159.25	161.64	213.88	224.41	blue
3	J_V3-fram	13.04	23.41	91.07	82.73	136.36	130.86	108.26	113.99	160.15	161.64	213.6	224.24	blue
4	J_V3-fram	13.04	24.35	91.07	83.73	136.36	130.98	108.26	113.99	159.25	159.67	213.88	224.24	blue
5	J_V3-fram	21.19	21.1	91.07	76.53	136.36	129.76	108.34	108.23	156.54	168.19	211.49	223.03	blue
6	J_V3-fram	17	17.46	91.07	78.45	136.36	130.25	108.26	108.05	155.32	154.73	211.07	222.19	blue
7	J_V3-fram	18.79	22.47	91.07	78.64	136.36	124.2	109.25	107.91	154.65	162.6	209.67	224.63	blue
8	J_V3-fram	23.41	17.46	91.07	78.64	136.36	129.19	106.98	107.38	159.48	162.6	209.48	222.58	blue
9	J_V3-fram	23.41	17.89	91.07	78.64	136.36	123.2	106.98	108.3	159.2	168.58	209.48	222.99	blue
10	J_V3-fram	47.01	52.47	94.05	81.47	130.6	126.27	142.01	135.42	236	244.01	315.36	335.77	blue
11	J_V3-fram	17.89	17	91.07	71.59	136.36	123.2	107.91	108.37	159.48	167.59	209.3	222.65	lime
12	J_V3-fram	53.04	46.39	91.84	81.39	136.4	126.27	142.01	124.15	236	244.01	315.36	335.77	blue
13	J_V3-fram	17.89	17.89	91.07	71.59	136.36	123.26	107.91	108.3	159.2	167.59	209.3	222.99	lime
14	J_V3-fram	51.01	46.53	87.46	84.85	130.21	101.98	127.13	129.76	236	244.01	315.36	335.77	blue
15	J_V3-fram	17.89	21.54	91.07	77.67	136.36	123.2	107.56	108.23	157.65	167.51	209.12	222.99	lime
16	J_V3-fram	38	38	86.27	68.43	125	126.27	107.84	107.17	167.43	161.62	315.36	335.77	blue
17	J_V3-fram	17	16.55	91.07	70.83	136.36	122.2	104.4	107.23	155.71	169.11	201.32	222.11	lime
18	J_V3-fram	38.64	38.64	75.66	59.67	136.36	126.27	101.64	101.27	165.42	168.19	214.4	335.77	blue
19	J_V3-fram	12.81	16.55	91.07	70.83	136.36	123.69	117.39	115.21	183.61	184.1	231.11	231.08	lime
20	J_V3-fram	44.55	39.62	66.47	64.4	136.36	126.27	101.87	101.2	164.15	168.58	213.03	335.77	blue
21	J_V3-fram	11.4	16.55	91.07	70.83	136.36	122.26	110.42	113.16	182.54	183.1	223.11	231.08	lime
22	J_V3-fram	45.4	38.01	65.05	60.9	136.36	126.27	101.64	101.41	164.94	168.67	212.81	335.77	blue
23	J_V3-fram	12.04	16.55	91.07	70.83	136.36	123.2	109.44	108.17	177.55	183.1	223.11	231.08	lime
24	J_V3-fram	44.55	39.01	65.76	56.86	136.36	126.27	101.87	100.13	165.13	168.58	213.03	335.77	blue

Figure 3.4: CSV file containing vectors for training

Step 5: This vector is used to train the RFC model which is used to classify the humans in the frames as either a violent or a non-violent person.

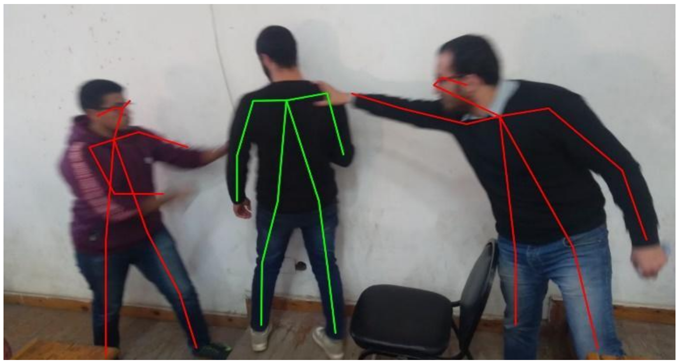


Figure 3.5: Classified Stick Figurines

Step 6: However, if the identifiable key points are lesser than a specified number for more than a specified number of identified humans, then that entire frame is sent to image classification for classification as violent or non-violent.

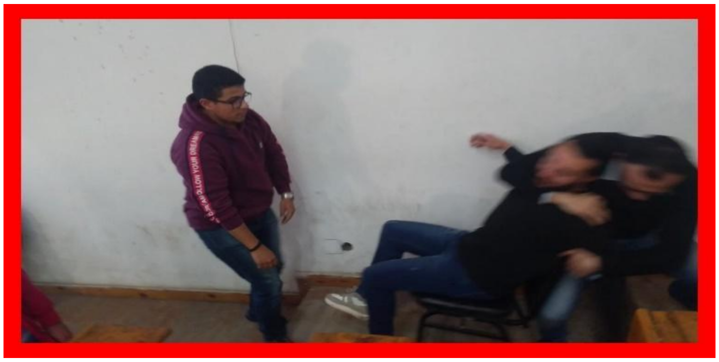
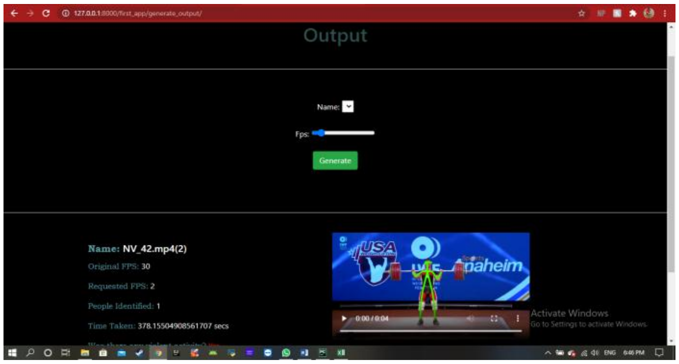


Figure 3.6: Frame classified from image classification

Step 7: These classified frames from both models are merged in chronological order into a video and displayed to the User on the Web User Interface.



127.0.0.1:8000/first_app/generate_output/
Output
Name:
Fps:
Generate
0:00 / 0:04
Activate Windows
Go to Settings to activate Windows.

Name: NV_42.mp4(2)

Original FPS: 30

Requested FPS: 2

People Identified: 1

Time Taken: 378.15504908561707 secs

Figure 3.7: Classified Output Video

A. System Architecture

The proposed system has four components, web application, ML engine, SQL database, and the video processor. The web application exposes three main features to the end-users, uploading a video, previewing any video, and most importantly the violence detection module. The violence detection module acts as the connection point to the ML engine. Once a video is chosen and the various parameters are selected the orchestrator uses the models generated at the training phase and processes each frame of the video and performs necessary action on the frames and stores them at a particular location. Once this is complete the video processor takes all these frames and generates a new video. The video processor used is FFMPEG. SQL database is used to store the videos and the other relevant information.

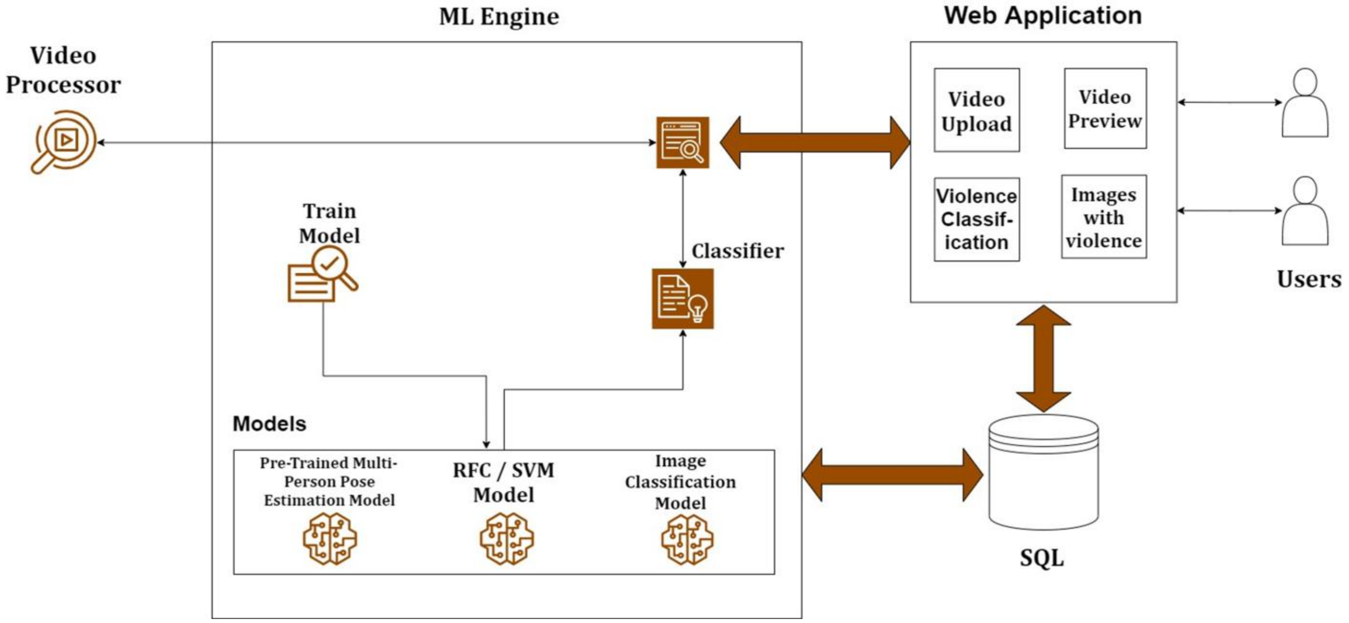


Figure 3.7: System Architecture

IV. METHODOLOGY

A. *Algorithms/methods used:*

i. Multi-person Pose Estimation Algorithm

When there are multiple people in a photo, multi-person pose estimation produces multiple independent key points. The system needs to figure out which set of key points belong to the same person. It uses an 18-point model trained on the COCO (Common Objects in Context) dataset for the functioning of this model. COCO model will identify the following key-points: Nose – 0, Neck – 1, Right Shoulder – 2, Right Elbow – 3, Right Wrist – 4, Left Shoulder – 5, Left Elbow – 6, Left Wrist – 7, Right Hip – 8, Right Knee – 9, Right Ankle – 10, Left Hip – 11, Left Knee – 12, Left Ankle – 13, Right Eye – 14, Left Eye – 15, Right Ear – 16, Left Ear – 17, Background – 18.

Input: Individual random frames from video segmentation step will be ordered and input to the multi-person pose estimation.

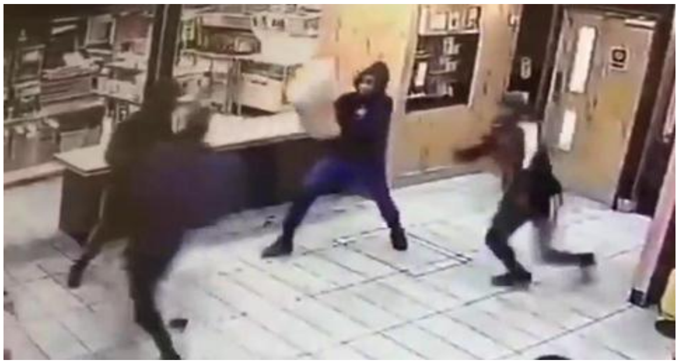


Figure 4.1: Input Frame

Output: Individual frames with Human Pose detected with stick figures drawn over them.

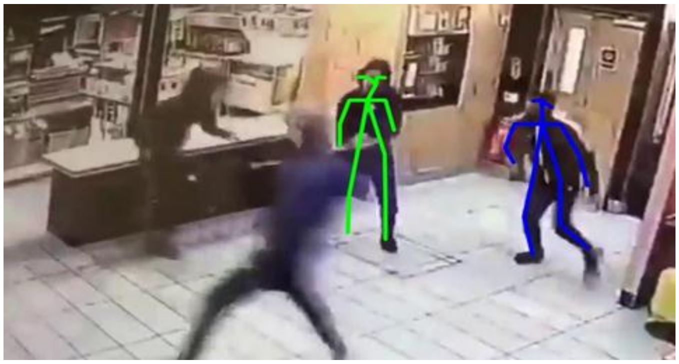
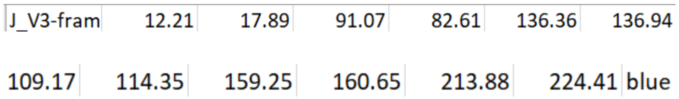


Figure 4.2: Output Frame

ii. Random Forest Classifier

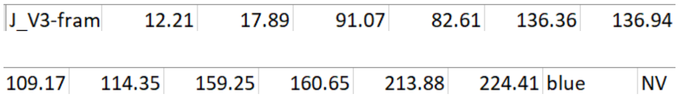
Random forest classifier creates a set of decision trees from a randomly selected subset of the training set. It then aggregates the votes from different decision trees to decide the final class of the test object.

Input: Vector of individual humans.



J_V3-fram	12.21	17.89	91.07	82.61	136.36	136.94
109.17	114.35	159.25	160.65	213.88	224.41	blue

Output: Classified vector of humans consisting of their key points based on their related vector after comparison with the trained model.



J_V3-fram	12.21	17.89	91.07	82.61	136.36	136.94	
109.17	114.35	159.25	160.65	213.88	224.41	blue	NV

iii. CNN Image Classification

A model is trained to identify images as either violent or non-violent. Later this model is called to predict images with insufficient key points as either violent or non-violent. Once images are classified as violent and non-violent, then a green border is drawn around non-violent images and a red border is drawn around violent frames.

Input: Only images that have insufficient key points.

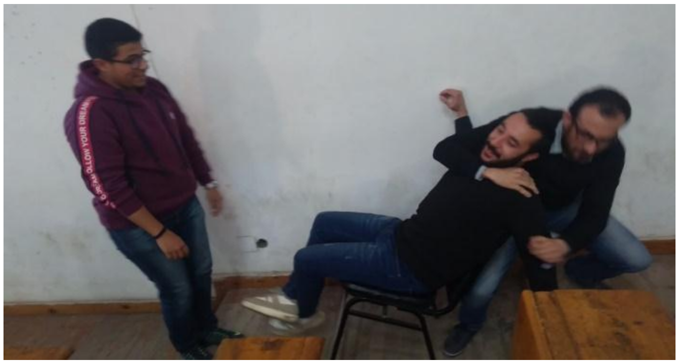


Figure 4.3: Input Frame with unidentifiable key points

Output: Classified Images as Violent or Non-Violent with a red border drawn for violent frames and a green border drawn for non-violent frames.

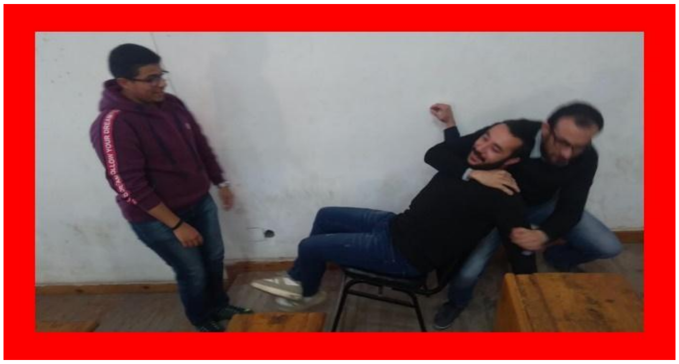


Figure 4.4: CNN Classified image

V. TEST CASES

- Test 1: Non-violent Greeting

Description: This is a video that involves two people who greet each other with a handshake and a hug which are considered to be non-violent activities.

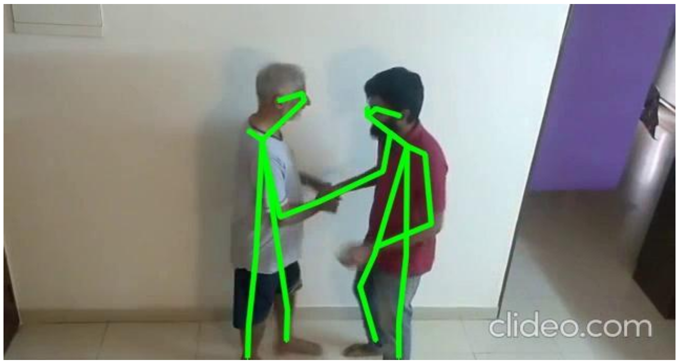


Figure 5.1: Greeting each other

- Test 2: Non-violent Weightlifting

Description: This is a video of a professional weightlifting event where an athlete lifts free weights over his head.

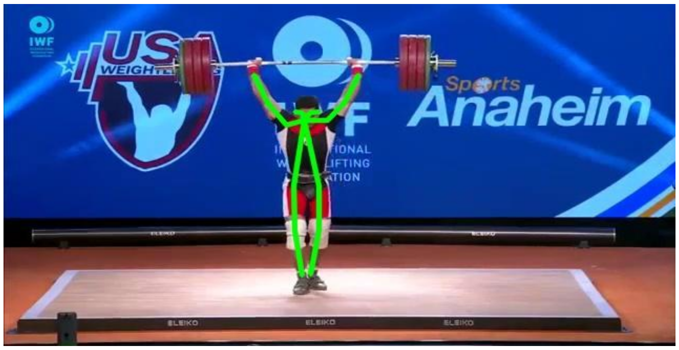


Figure 5.2: Weight lifting

- Test 3: Atm-attack-1

Description: This is a video of an ATM scenario where a thief assaults and steals money from an individual who had just withdrawn some money. The kicking action is identified as a violent posture as seen in the below frame.

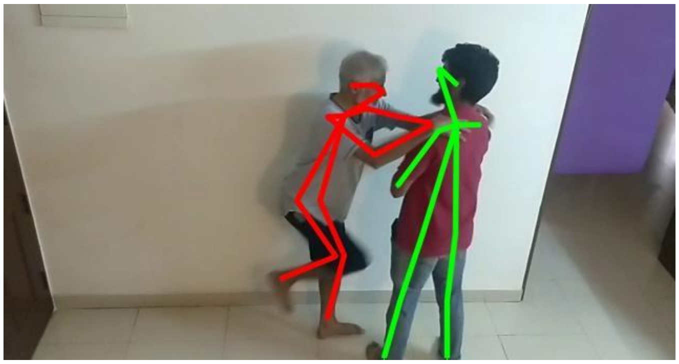


Figure 5.3: Assaulting a person

- Test 4: Atm-attack-2

Description: This is a video of an ATM scenario where a thief points a gun and then strikes the victim with the same gun and steals money from him. The proximity and the posture of the person with the gun helps classify the frame as a violent one. Along with this, other frames with the person holding the gun have also been classified as violent.

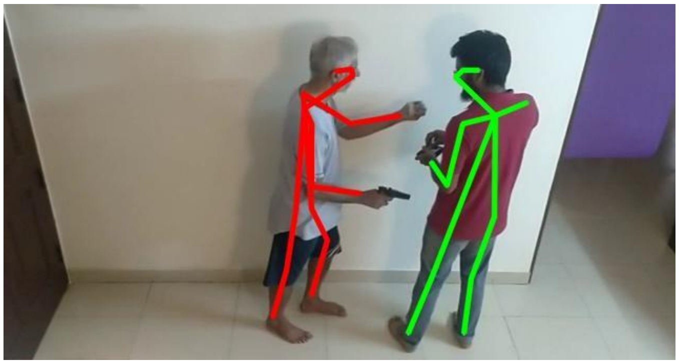


Figure 5.4: Threatening a person

- Test 5: Store-mob-fight

Description: This is a video taken from a CCTV camera of a mobile store where a fight breaks out between a customer and the owner. We have included this test case to demonstrate the efficiency and accuracy of our system in situations where there are groups of people involved in violent activities in close proximity to each other.

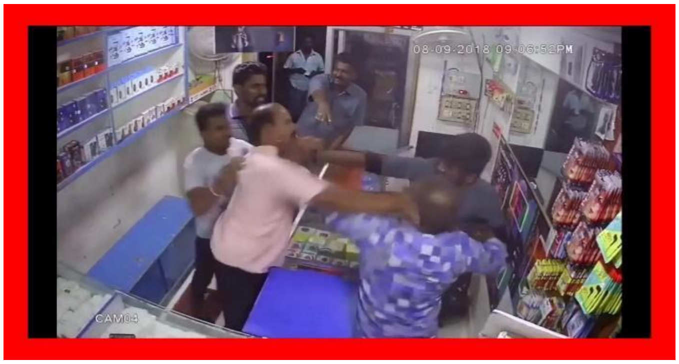


Figure 5.5: Shop Fight caught on camera

VI. RESULT AND ANALYSIS

From the above test cases, it can be observed that the system has more recognitions of true positives and true negatives as compared to recognitions of false positives and false negatives. The system provides an accurate recognition of up to 95.3% on average.

Test Cases	Cumulative no. of people in all frames	No. of incorrect classifications	Accuracy
1.Non-violent Greeting	60	1	98.33%
2. Non-violent Weightlifting	11	0	100%
3. Atm-attack 1	108	1	99.07%
4. Atm-attack 2	76	4	94.73%
5. Store-mob fight	77	12	84.41%

Table 6.1: Performance Evaluation

CONCLUSION

Human safety is imperative and is of the highest priority. With the God's Eye System in effect, the detection of violence in real-time allows for swift action and immediate justice through governed intervention.

High-security locations like ATMs can go completely man less on the security front with surveillance being automated. The surveillance can be remotely processed in real-time in the ATM itself or it can be sent to a cloud for processing. With the processor sizes decreasing and the processing power increasing, real-time processing at the edge node is becoming more viable in terms of server space occupancy and financial affordability. Whereas the cloud processing alternative provides financial affordability at the cost of speed of processing as a network latency exists for transporting the input feed to the Cloud and bringing back the results.

Initial research and investment costs may be steep compared to having human monitoring system, but in the span of time, the God's Eye system will prove to be more viable in all aspects. It requires almost no rest period and can work continuously around the clock to achieve maximum security. The scope of this paper and its areas of application are plenty.

REFERENCES

[1] Fillipe D M Desouza et al. "Violence Detection in Video Using Spatio-Temporal Features", the international conference on computer vision (ICCV), 2014.

[2] Bruno Peixoto et al. "Toward Subjective Violence Detection in Videos", IEEE ICASSP 2019, UK.

[3] Shakil Ahmed Sumon et al. "Violent Crowd Flow Detection Using Deep Learning", Asian Conference on Intelligent Information and Database Systems 2019, Indonesia.

[4] Oscar Deniz et al. "Fast Violence Detection in Video", International Conference on Computer Vision Theory and Applications (VISAPP) 2017.

[5] Yilun Chen et al. "Cascaded Pyramid Network for Multi-Person Pose Estimation", the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2018.

[6] Amarjot Singh et al. "Eye in the Sky: Real-time Drone Surveillance System (DSS) for Violent Individuals Identification using ScatterNet Hybrid Deep Learning Network", Efficient Deep Learning for Computer Vision (ECV) workshop at IEEE Computer Vision and Pattern Recognition (CVPR) 2018.