sciendo

VOLUME 16, N° 1 2022
Journal of Automation, Mobile Robotics and Intelligent Systems

# Autonomous Anomaly Detection System for Crime Monitoring and Alert Generation

Submitted: 8th December 2021; accepted 6th September 20222

*Jyoti Kukad, Swapnil Soner, Sagar Pandya*

DOI: 10.14313/JAMRIS/1-2022/7

**Abstract**

*Nowadays, violence has a major impact in society. Violence metrics increasing very rapidly reveal a very alarming situation. Many violent events go unnoticed. Over the last few years, autonomous vehicles have been used to observe and recognize abnormalities in human behavior and to classify them as crimes or not. Detecting crime on live streams requires classifying an event as a crime or not a crime and generating alerts to designated authorities, who can in turn take the required actions and assess the security of the city. There is currently a need for this kind of effective techniques for live video stream processing in computer vision. There are many techniques that can be used, but Long Short-Term Memory (LSTM) networks and OpenCV provide the most accurate prediction for this task. OpenCV is used for the task of object detection in computer vision, which will take the input from either a drone or any autonomous vehicle. LSTM is used to classify any event or behavior as a crime or not. This live stream is also encrypted using the Elliptic curve algorithm for more security of data against any manipulation. Through its ability to sense its surroundings, an autonomous vehicle is able to operate itself and execute critical activities without the need for human interaction. Much crowd-based crimes like mob lynching and individual crimes like murder, burglary, and terrorism can be protected against with advanced deep learning-based Anamoly detection techniques. With this proposed system, object detection is possible with approximately 90% accuracy. After analyzing all the data, it is sent to the nearest concern department to provide the remedial approach or protect from any crime. This system helps to enhance surveillance and decrease the crime rate in society.*

**Key words:** *Autonomous vehicle, LSTM, Open CV, ECC, Crime and Streaming*

## 1. Introduction

### 1.1 Motivations

The advancement of technology is a boon to society, but a growing population equipped with the state-of-the-art technology at hand has led to an unprecedented rise of criminality. In the past, hotspots of crime had cameras installed, and manual monitoring of those locations was completed by observing multiple screens. These types of systems work well for less crowded areas. With the introduction of data mining and deep learning techniques, autonomous violence detection techniques are widely used for video surveillance [1]. A lot of work has been carried out in recent years on recognition of human actions using vision and acoustic technologies. These technologies are used to monitor human behavior.

Violence in society is the biggest issue and the most important goal is to detect it. The rapid growth of violence in society is crucial for everyone. As per the above-mentioned report, many cases are not noticed, and the accused are not identified by the authorities. Society needs more digitization of the system wherein video surveillance will be required to detect violence. This system is easy to deploy and not very cost-effective to install. Currently available surveillance systems are sometimes ineffective and insufficient to predict the accurate result. For optimal understanding, a huge amount of video needs to be captured for surveillance. For the correct and accurate measurements, we need a system-trained system [3]. The automation process is a large application to understand when it comes to defining the meaning of crime with different weapons. The proposed kind of system requires the machine-leering method to train the data and predict accurate results. The system works on real-time incidents of violence in any place. Autonomous vehicles are the intelligent key for monitoring and detecting crime.

### 1.2 Contributions

This system's goal is to identify anomalies, which are noticed by a well autonomous vehicle attached with a camera. The system anomaly detection is related to the behavior of the public, especially carriable objects identified in a public crowd. Anomaly detection refers to crime detection, including different kinds of violence or detection of weapons such as a gun, sword, or knife. This type of detection is based on cameras which generate an alarm to the authorities. The entire system provides for a safe city and leads to a safer city over time [5]. Live surveillance with this automobile camera is also made secure using an ECC algorithm so that the live stream will not be able to be altered.

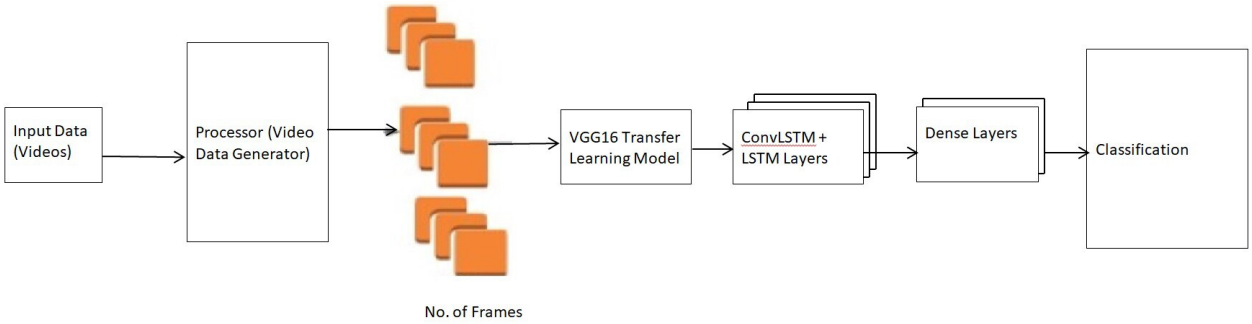


**Fig. 1.** Architecture of System

## 2. Literature Review

Bruno M. Peixoto et al. present a system where violence detection is based on different mechanics based on machine learning [1]. This proposed system breaks into different understood levels of violence (i.e., fights, explosions, blood, and gunshots) and combines all of them for a more detailed understanding of particular scene detection. Systems that explore the depth of different kinds of violence, e.g., gunshots, have more weight than audio features alone, implying the need for a multimodal approach trained on visual and audio features. The machine learning here was based on different DCNNs for detecting violence, focusing on the medieval 2013 VSD dataset. The system deliberated and trained different DCNNs (static and motion-based), each of which was responsible for detecting an object based on a single aspect.

OViF (Oriented Violent Flows) as a reliable feature for violence detection were investigated by Yuan Gao et al [2]. It is based on the concept of motion orientation for detecting changes in motion magnitude. OViF is a better choice for violence detection in non-crowded scenarios. However, for crowded scenarios, this approach fails to generate acceptable results. As such, the paper used a combination of both OViF and ViF (Violent Flows) as feature extraction techniques to detect crowded and uncrowded scenes for better violence detection. The results were generated by using Ada boost and Linear SVM as a multi-classifier way for attaining better classification performance. SVM was selected for its simplicity, effectiveness, and speed. AdaBoost was selected as it was the most efficient boosting algorithm. The two databases used for evaluating the effectiveness were the Hockey Fight Database and the Violent-Flows Database. The performance of OViF was better than ViF on the Hockey Fight Database, but on the ViF database, OViF did not generate satisfying results. The final results on violence detection were generated using ViF+OViF with AdaBoost and SVM. The most challenging task was monitoring crowded events, especially to detect the violence in real time in a timely manner. The proposed system of this model has provided a novel approach for detecting violence in real-time crowded scenes, which is based on flow vector magnitude.

In a paper by Tal Hassner et al, the system collected shot frame sequences and used the violent flow descriptor [3]. After dataset collection, SVM support vector machines based on linear classification classified the out frame as violent or nonviolent. Systems used video surveillance to test the data and provide the accuracy with effectiveness. This kind of model required a high level of motion and shape analysis of the dataset. Despite its limitations and privacy concerns, video surveillance plays an important role in society in instilling a sense of safety, trust, and security. In a paper by Ding et al., violence detection used the 3D ConvNets without previous information, which is used for supervised learning for model training, and used the backpropagation approach for computing gradients [4]. There are several flaws in this system that need to be addressed in order to increase accuracy. On the other hand, it can learn video features automatically. In uncrowded circumstances, this method employs a CNN-based algorithm to recognize categories if a video contains aberrant human behaviors like falling, loitering, or aggression. It is efficient and precise since it operates directly on the picture pixel. It has the ability to learn video features on its own. On a hockey dataset, the system achieved a 91.00% accuracy rate.

G. Mu et al. presented two techniques for using CNNs for classification: one used them end-to-end, taking raw data as input and producing classification results at the last layer; the other used them in a layer-by-layer fashion [5]. For feature representations, CNNs were effective. Video clips with shots, screams, and heavy metal music received high ratings because of their explicit violent relevance, which is an advantage of adopting this model. This model has an Average Precision (AP) of 0.485 and 0.291 on the MediaEval 2015 dataset on the validation and testing sets for Visual- only + End-to-end CNN based on Audio-only. On numerous visual recognition tasks, such as object recognition, the usefulness of CNN models has been demonstrated by Q. Dai et al [13]. Many applications can benefit from these techniques for detecting violent scenes and predicting emotional effects in videos. Longer-term temporal dynamics are incorporated in this model by layering LSTM on top of the two stream CNN features. This approach is quite successful at detecting violence, but it requires more computing complexity. E. Ditsanthia et al analyzed the behavior of crowds, one of the most active study areas in computer vision [25]. The gathering of people in groups at one place is defined as a crowd. The crowd may be different at different locations. The identification of violence is one of the most difficult challenges in crowd behavior analysis. This algorithm was quite successful at detec-

**Table 1. Comparison of papers with key features**

| Sr. No | Papers | Key Features |
|---|---|---|
| 1 | Bruno M. Peixoto [1] | – Breaks into the different understood levels of violence<br>– DCNNs for detecting violence which focuses on the medieval 2013 VSD dataset |
| 2 | Yuan Gao et al. [2] | – Used a combination of both OViF and ViF (Violent Flows) as feature extraction techniques<br>– Ada boost and Linear SVM provided a multi-classifier way for attaining better classification performance |
| 3 | Tal Hassner et al. [3] | – The approach is based on flow vector magnitude to detect the real-time detection of violence in crowded scenes based on flow vector magnitude<br>– SVMs (support vector machines) based on linear classification classify the out frame as violent or nonviolent |
| 4 | C. Ding et al. [4] | – Backpropagation approach for computing gradients<br>– CNN-based algorithm to recognize and categorize |
| 5 | G. Mu et al. [5] | CNNs classification used for |

ting violence, but it requires a massive training dataset of violent films. To calculate accuracy, this model was applied to several datasets, with 75.73% accuracy on the Real-Violent Dataset, 88.74% accuracy on the Movie Dataset, and 83.19% accuracy on the Hockey Fight Dataset. F. U. M. Ullah et al. presents a system where the violence is detected automatically; this prompt response is enormously useful, and it may greatly aid the relevant departments [26]. Violence is an aberrant behavior and activity that often involves the use of physical force to harm or kill, whether the victim be a human or an animal; these acts may be detected on a system baked on smart surveillance, which can be utilized to avoid more catastrophic occurrences. Pre-trained lightweights are used in this approach. Mobile Net CNN aids in the reduction of mass processing of ineffective frames. For feature extraction, this system employs 3D CNN. This device employs closed-circuit television, and if violence is detected, the nearest security agency or police station is notified. Unnecessary processing of worthless frames is eliminated by utilizing this method. It also performed better in terms of accuracy. This system necessitates good hardware, and devices with fewer resources may be unable to implement the suggested concept. This model was tested on a variety of datasets, including a violent crowd dataset with 98% accuracy, a movie dataset with 99% accuracy, and a hockey battle dataset with 96% accuracy.

## 3. An Implementation Map

A. **Computational intelligence:** Computational intelligence is the ability to learn from previous examples and data. It performs in the same way that human beings' intelligence does. Artificial intelligence (AI) is the overarching discipline that covers anything related to making machines intelligent. Whether the machine is in the field of autonomous vehicles, robotics, home appliances, or daily-use system and software applications, if we apply artificial intelligence on it, it will become smarter and provide results that are easier for users [12]. AI-based systems perform high-level, intelligent functions like communicating in languages, learning, reasoning in the same way the human brain does, and so on. Artificial intelligence learning can be completed through neural networks. A neural network is a net of interconnected computing elements known as neurons. Figure 2 represents a single layer feedforward neural net: here, X is a input layer neuron, Y is the output layer, $Y_{in}$ is the net input calculated at output layer neuron, and w is the weight between neurons and $f(Y_{in})$.

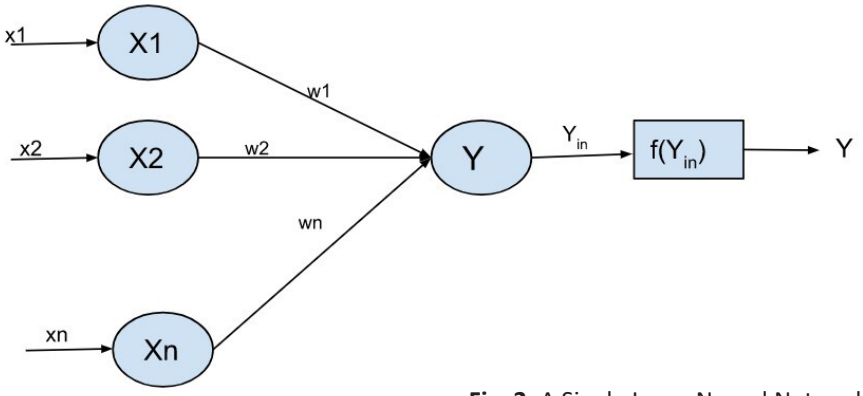


**Fig. 2.** A Single Layer Neural Network Architecture

B. **Machine learning:** Machine learning can be defined as machines learning by themselves and adapting to changes in the environment in which they work. For example, we made our system learn by providing it with datasets for crime classification. However, if there are test data which are provided for validation, such as footage of a knife used to cut vegetables, and they are not labeled in the dataset, then our system must learn to classify this new data as non-crime. As such, it needs to cover its knowledge of previous examples which were provided for study, hence updating its own intelligence. For machine learning, there are many techniques which work the same as animal or human learning. Machines can be made to learn in two ways: either structure-based or parameter-based. Updating either structures or parameters in the desired way leads to the expected target output. But as updating structure for learning is not always possible, parameter learning is preferred. In our system, we have millions of parameters which get updated for expected performance [11]. One of the major flaws of machine learning is the quantity of data machines require. A small amount of data does not provide enough learning, which leads to poor performance of the ML model. Major applications of machine learning are in recognition tasks such as handwriting recognition, face recognition, etc., or predictions of some event or label, forecasting, etc. All of systems require intelligence. More epochs of learning with enough data continue to enhance the performance of ML models.

C. **Deep learning:** A major issue with machine learning models is to select the appropriate set of features which are relevant for the desired or expected output of the model. Selecting these features in the training dataset is up to the user. This requires expertise in the domain of implementation; also, it requires a high processing time for the dataset which includes high dimensions, e.g., signals or videos. Here, we use deep learning architectures which extract features from data that are more likely relevant for decision-making. Here, we have one input layer, one output layer, and multiple intermediate or hidden layers of computational elements, i.e., neurons. In Figure 3, X is the input layer, Y is the output layer, and all others are hidden layers; weights between different sets of layers can be different as activation functions [6]. Here, the processing is nonlinear, because the data here requires nonlinear features like in images, video streams, or sounds. These kinds of data contain many parameters which can easily be handled with deep learning rather than easier machine learning models like a support vector machine, which is a popular net for nonlinear classification. The performance of such models is greatly affected by high numbers of dimensions in data. More features improve performance until a limit, after which performance drops significantly. Deep learning architecture solves these issues better by taking each layer independently and training in a greedy way. First, one layer gets trained completely, and then the next layer starts training with input from the previous layer [16].

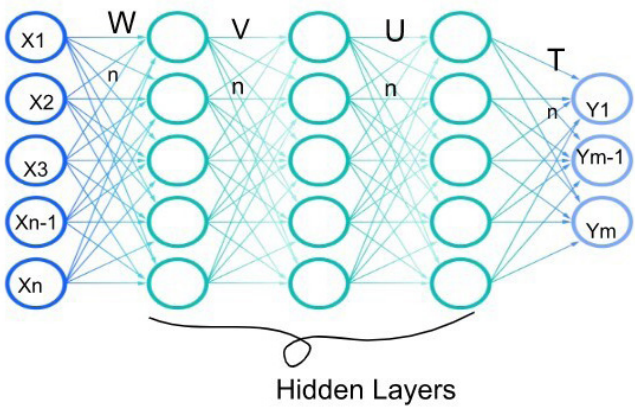


**Fig. 3.** Deep Learning Architecture

D. **CNN:** The concept of convolution networks is to combine local computations, i.e., on weight sharing units, where the convolution of the signal is performed and pooling is completed. The system is invariant due to the convolutions, as parameters such as weights are dependent on spatial separation rather than spatial position [7]. The pooling creates a more suitable collection of features through a nonlinear combination of the preceding level features and consideration of the input data's local topology. By alternately employing convolution layers and pooling layers, the model collects and integrates local features and creates a better representation of the input vectors. The connectivity of the convolution networks, where each neuron of convolution or a pooling layer is fully connected to a small subset of the previous layer, trains the network with multiple layers. Through error gradient backpropagation, the supervised learning is easily accomplished. CNNs need relatively minimal preprocessing compared to other image classification algorithms. The usage of a variant of multilayer perceptions by CNN is intended to need as little preprocessing as possible. In comparison to other image classification algorithms, CNNs require very minimal pre-processing. This implies that the network learns the filters that were previously hand-engineered in traditional algorithms. Convolution layers apply the convolution operation to input before forwarding the output to the next layer. The convolution emulates the response of an individual neuron to visual stimuli [25]. This feature design independence from earlier knowledge and human effort is a significant advantage. The convolution operation solves this concern by limiting the number of free parameters in the network, permitting it to be deeper with fewer parameters [20]. In this way, backpropagation eliminates the issue of diminishing or exploding gradients in training typical multi-layer neural networks with several layers.

E. **RNN**: Recurrent neural networks are neural networks that take the output of previous iterations as input and learn it through all their hidden states. They use previous output as feedback, and also memorize the independent variables as input, thus providing accurate predictions. RNN works by capturing the input

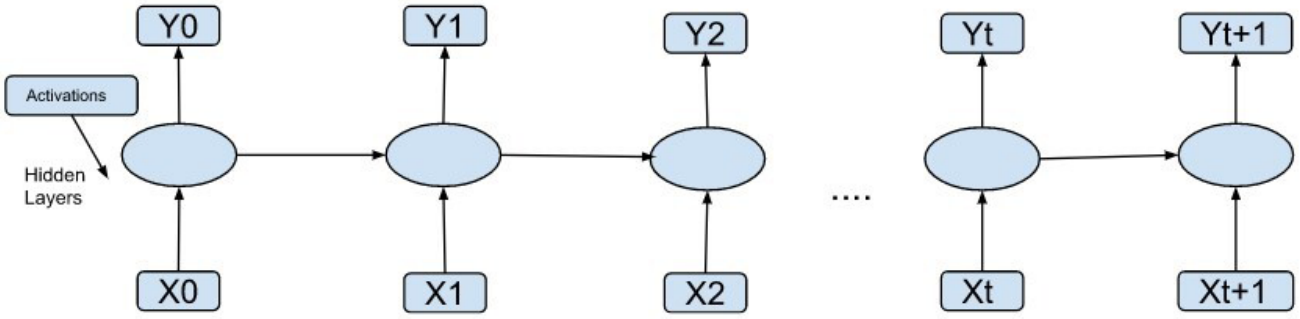


**Fig. 4.** Recurrent Neural Network

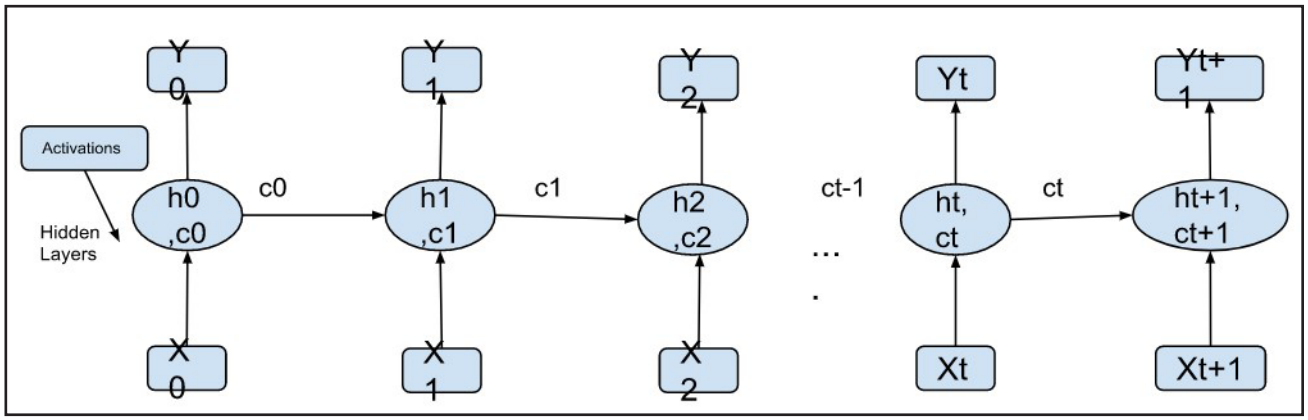


**Fig. 5.** LSTM

data's sequential information. RNNs process the sequence of inputs by using their internal state, i.e., memory. In Figure 4, X0 indicates the first input after applying activations at input, while the hidden and output layers provide output Y0. When we say that it takes the output of previous state as its input, that means that while taking input at state X1, it includes the output of the previous state Y0; similarly, the input for state X2 will combine Y1. This way, it keeps training with outputs generated previously and takes them as learning signals [24]. RNN works with sequential data, so it works well with temporal data. That is why for our system to classify input as crime or not crime, we used an LSTM, which is an RNN. It takes input video sequences and processes them frame by frame. RNNs follow gradient descent for learning in a backpropagation-of-error fashion. When the gradient becomes small and an RNN does not learn well by updating its weights, then it suffers from vanishing gradients.

F. **LSTM:** Long Short Term Memory is a kind of recurrent neural network (RNN). In RNN, it solves the problem of vanishing gradients. It solves this problem by removing the long-term irrelevant dependencies of the current prediction. It finds out which recent or past input is useful for the accurate decision and forgets all others. For this, LSTM architecture uses 3 gates at its hidden states, namely the input, output, and forget gates. These gates keep track of information relevant for its prediction. The input gate checks which input should be used to change the memory of the cell. The forget gate (a neural network with sigmoid) finds details to be discarded. For this, it uses the sigmoid function. The output gate takes the input and the memory of the block and decides the output. If any detail needs to be kept, it outputs 1; otherwise, it outputs 0 [23]. In our application of generating sequences of events in video streams and making decisions, vanishing gradients may lead to inaccurate predictions, but LSTM works in its desired form. For this, LSTM includes cells at each hidden layer neuron. Each cell takes the input of the current state and the output of the previous state and cell; e.g., in Figure 5, the input of cell ct will be input xt and output ht-1 and previous cell state ct-1. Three logistic sigmoid gates and one tanh layer make up an LSTM. These gated layers control the information which flows from the cells.

G. **ECC:** Elliptic curve cryptography is a very important and modern algorithm based on the asymmetric public key cryptography system. This is based on the finite field and difficulty of the elliptic curve discrete logarithm problem. This algorithm uses the key as a small value and smaller signature for the security but is fast to generate the key and signature [9]. The ECC uses a curve that provides very strong security with speed for the whole process of cryptography. Then, mathematically, we can understand the curve and the point (p, q) and describe the equation:

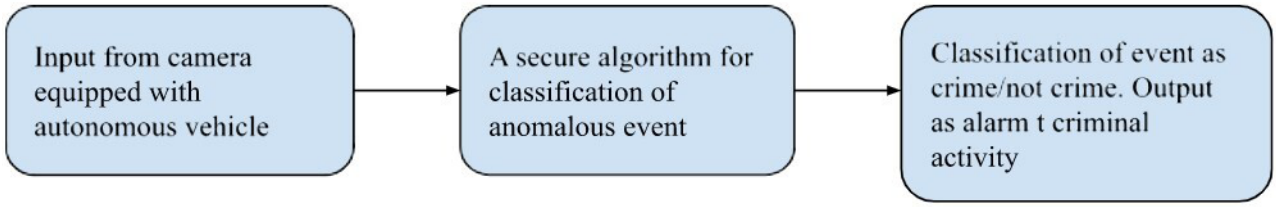


**Fig. 6.** Basis Flow Diagram of Proposed System

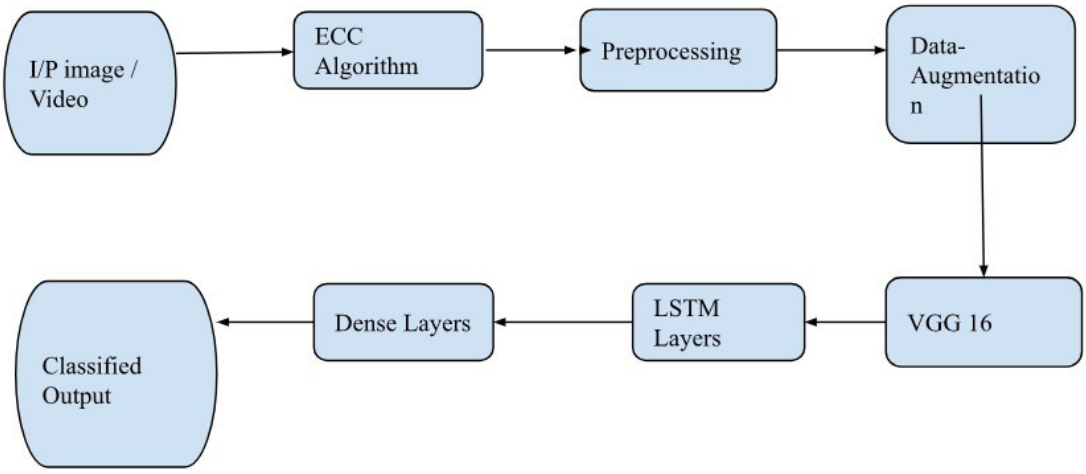


**Fig. 7.** Flow of the Methodology

$$Ap3+Bp2q+C\,pq3+Dq3+Ep2+Fpq+Gq2+Hp+Iq+j=0$$

The cryptography elliptic curve is defined as:

$$q2=p3+\_a\_x+b$$

This encryption algorithm is used in this proposed system to protect the data from unauthorized users. Sometimes, hackers intentionally create a problem by manipulating the data, so we are using a more secure concept..

## 4. Methodlogy

Autonomous Anomaly Detection System for Crime Monitoring and Alert Generation

This is a system for identifying anomalies in any event which is seen by a camera equipped with an autonomous vehicle. Any anomaly related to the behavior of an individual in the public or in a crowd can be detected by this system. Crime which includes physical violence or use of weapons such as knives or guns can be monitored using this mobile camera, which can also generate an alarm to the relevant security system. This whole process can lead to a safer city. Live surveillance with this automobile camera is also made secure using the ECC algorithm, so this live stream will not be able to be altered.

To monitor any event as a crime or not a crime, we have tried multiple methods. We experimented with CONVO-3D, VGG16 + LSTM, VGG16 + CONVO-LSTM, VGG16 + ANN, InceptionV3 + ANN, and InceptionV3 + GRU, but the highest accuracy we got was from GG16 + CONVO-LSTM.

In the proposed method, first, the upload or live stream of an image or video goes through the ECC algorithm for a security check [17]. After passing it, the input is then preprocessed using a video data generator, before presenting it prior to learning. Here, the video is converted to time series images. Then, these RGB images are processed to YUV representation. The Y grayscale information is in the image or component of the luminance, while the U and V components contain the chromatic or color data. With each step, images are subsampled for lower resolutions. We have used the Keras Image Data Generator to convert raw videos into images divided in batch sizes. This Image Data Generator, instead of returning frames, provides a set of frames according to temporal length, temporal size, and batch size. Data augmentation is applied to enhance the size and quality of datasets used for training, which leads to a better model.

The architecture starts with feature extraction using VGG16. Then, there are 2 ConvoLSTM layers, with each one followed by Batch Normalization and a max pooling layer. Then, there is one LSTM layer, and at last, there are two dense layers for performing the task of prediction. This architecture includes the stack of many layers, which includes convolution with Rely activations that are nonlinear, spatial pooling carried out of max pooling layers, and the output layer is a softmax layer [10].

VGG16, a visual geometry group with 16 layers, is a vision architecture in the CNN model which is used

for image classification. Its architecture includes a 224 x 224 RGB image as input, followed by a stack of convolutional layers, where it filters with a very small receptive field: 3 × 3 are present. Further, there are max pooling layers and a fully connected stack of convolution layers. ConvLSTM is also an RNN for spatio-temporal prediction that has convolutional structures in both the input-to-state and state-to-state transitions. ConvLSTM determines the future state of a certain cell in the grid by the inputs and past states of its local neighbors [24]. This is like LSTM, but internal matrix multiplications are exchanged with convolution operations. The model has a single input, and the trailer frames in sequence are generated by Custom Video Data Generator and 2 independent outputs, one for each category. Then it gets chopped into branches where each category has one branch. Here, the processing for each branch is the same. All the branches are the same, as they start with one ConvLSTM layer then a MaxPooling layer. Then, this is connected to a fully-connected Dense network. Finally, the last layer is a Dense with a single cell. The next image illustrates the simplified model with only two categories (crime or not). The model was trained in 10 epochs.

### Convonet configurations

```
Model: "Model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
trailer_input (InputLayer)   [(None, 5, 224, 224, 3)]  0
_________________________________________________________________
time_distributed (TimeDistri (None, 5, 7, 7, 512)      14714688
_________________________________________________________________
conv_lst_m2d (ConvLSTM2D)    (None, 5, 7, 7, 20)       383120
_________________________________________________________________
batch_normalization (BatchNo (None, 5, 7, 7, 20)       80
_________________________________________________________________
max_pooling3d (MaxPooling3D) (None, 5, 4, 4, 20)       0
_________________________________________________________________
conv_lst_m2d_1 (ConvLSTM2D)  (None, 5, 4, 4, 10)       10840
_________________________________________________________________
batch_normalization_1 (Batch (None, 5, 4, 4, 10)       40
_________________________________________________________________
max_pooling3d_1 (MaxPooling3 (None, 5, 2, 2, 10)       0
_________________________________________________________________
time_distributed_1 (TimeDist (None, 5, 40)             0
_________________________________________________________________
lstm (LSTM)                  (None, 128)               86528
_________________________________________________________________
dense (Dense)                (None, 64)                8256
_________________________________________________________________
dense_1 (Dense)              (None, 2)                 130
=================================================================
Total params: 15,203,682
Trainable params: 488,934
Non-trainable params: 14,714,748
_________________________________________________________________
```

**Fig. 8.** Convonet configurations

Here, we have used the OpenCV library that has a huge amount of content of computer vision and is helpful for real-time operations. Using this, we have converted the video streams into frames and also used them for detection of objects like guns and knives.

**Datasets taken for training include:** Gun detection dataset; hockey fight detection dataset; large-scale anomaly detection which includes burglary, explosion, fighting, and shooting videos; a dataset for evaluating blood detection. For example, the below image includes physical violence so this is labeled as "crime" and an alert will be generated. The optimizer used here is ADAM. The loss function used here is Binary Cross entropy [19].


**Fig. 9.** Result: Label - Crime


**Fig. 10.** Result: Label - Not Crime

**Fig. 11.** Result: Label - Crime

## 5. Results

As stated above, we have tried multiple models to find the best performance out of them. Below is the comparison of different models' performances over training and validation datasets.

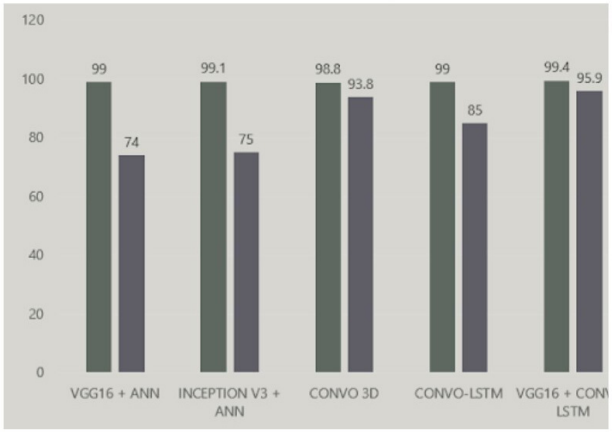


**Fig. 12.** Performance comparison chart

The proposed model, VGG16 + ConvoLSTM + LSTM, provided a training accuracy of 99.4% and validation accuracy of 95.9%. The prediction performed by this model is the most accurate. The model loss at the time of training is 0.017 and at validation it was 0.143.

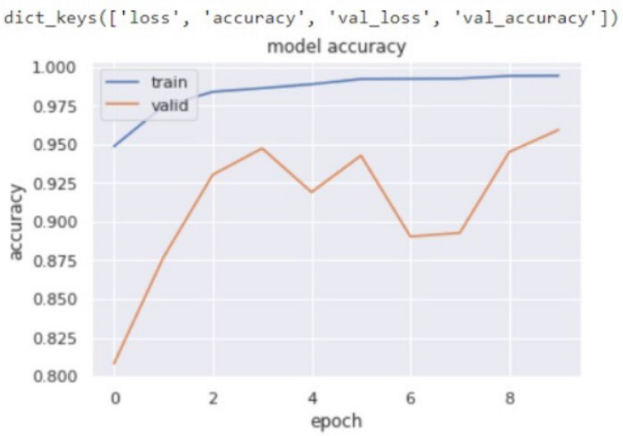


**Fig. 13.** Graph for accuracy at training and validation

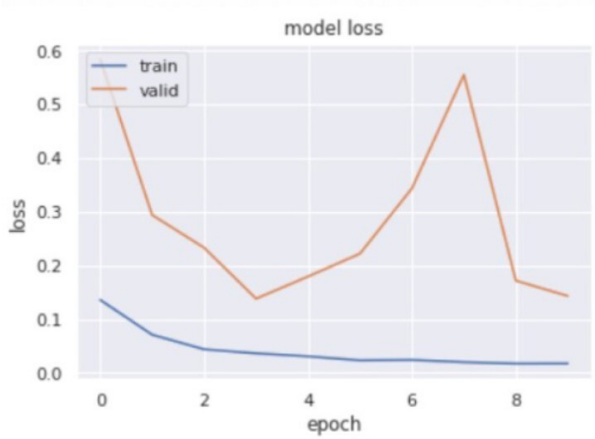


**Fig. 14.** Graph for loss at training and validation

## 6. Findings and Discussion

### 6.1 How Will the Machine Learning Solution Be Used in States?

With the help of machine learning and stored data at different levels, systems can understand and detect different crimes. This detection of crime through the autonomous system process is crucial to maintaining the whole system with efficiency. This system, in particular, provides the accuracy to detect crime as well as associated objects. Our system may prove to create very useful devices for society.

### 6.2 What Kind of Digital Infrastructure Should Be Used for The Individual System?

Drone types of autonomous systems of smart cameras will help us to detect the crime. We use efficient algorithms to understand the learning of given data sets. With this detection, we have also developed the system to send an alarm to the authorities for any given location.

### 6.3 How Can Autonomous Vehicles Be Used for Better Control of the System in Communication and Design?

For better control and more efficiency in communicating with good design, autonomous vehicles need a well-equipped drone system and a high camera quality. In addition, our system's learning methodology will help give an accurate result. When the system gives us the proper result, our performance will increase.

### 6.4 How Are Indian Government Regulations Related to Detection of Crime Through Autonomous Vehicles?

The Indian government still does not make their thoughts regarding new technologies for the detection of crime and creation of reports clear. All the authority related to crime is currently based on manual verification done by the police department. The government should accept, apply, and properly communicate with the citizens regarding this autonomous detection system and evidence collection.

## 7. Conclusion

Timely detection of crime can save lives. A still camera can work, but a camera with an autonomous device can enhance the security. There may be many anomalies in crowd behavior, but recognizing them and then classifying them as crime is a crucial task. While there are other systems designed to perform this task, we propose a novel model to detect criminal activities among people. To check performance, we brought up challenging datasets of crime, violence, and objectionable carried objects. We have implemented 5 models which are com-

binations of techniques and applied these models on a variety of datasets. We have concluded that the VGG16 + CONVOLSTM + LSTM Model built on a custom video data generator works well as per the Accuracy and Loss Comparison charts. To capture live streams, there are many challenges, including camera quality, background noise, lightning conditions, etc. Generating an alarm will help to keep the city safe against crime. This whole system is also secured with the highly secure algorithm of cryptography, but also collectively increases the complexity of the system which requires high end configuration, which can be a limitation to process live streams. In the future, we will work on audio systems that give better results to contribute to the surveillance of anomaly crime detection.

## AUTHORS:

**Jyoti Kukade** – Medi-Caps University, Indore, Email: jyoti.kukade@medicaps.ac.in.

**Swapnil Soner* –** Medi-Caps University, Indore, Email: Swapnil.soner@gmail.com.

**Sagar Pandya** – Medi-Caps University, Indore sagar.pandya@medicaps.ac.in.

## REFERENCE

[1] B.M. Peixoto, B. Lavi, Z. Dias, A. Rocha, "Harnessing high-level concepts, visual, and auditory features for violence detection in videos", *Journal of Visual Communication and Image Representation*, 10.1016/j.jvcir.2021.103174

[2] Y. Gao, H. Liu, X. Sun, C. Wang, Y. Liu, "Violence detection using Violent Flows", 10.1016/j.imavis.2016.01.006

[3] T. Hassner, Y. Itcher & Y. Itcher, *Violent Flows: Real-Time Detection of Violent Crowd Behavior*, IEEE, 2012, www.openu.ac.il/home/hassner/data/violentflows/ 978-1-4673-1612-5/12/$31.00

[4] C. Ding, S. Fan, M. Zhu, W. Feng, and B. Jia, "Violence detection in video by using 3D convolutional neural networks", *Lect. Notes Comput. Sci.* (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8888, 2014, pp. 551–558. 10.3390/app11083523

[5] G. Mu, H. Cao, and Q. Jin, "Violent Scene Detection Using Convolutional Neural Networks and Deep Audio Features," 2008, pp. 645–651. 10.1109/ICCSP48568.2020.9182433

[6] G. Sakthivinayagam, R. Easawarakumar, A. Arunachalam, and M.Pandi, "Violence Detection System using Convolution Neural Network", *SSRG Int. J. Electron. Commun. Eng.*, vol. 6, 2019, pp. 6–9.

[7] B. Peixoto, B. Lavi, and P. Martin, "Toward subjective violence detection in videos", ICASSP, *IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, 2019, pp. 8276–8280.

[8] A. Hanson, K. Pnvr, S. Krishnagopal, and L. Davis, "Bidirectional convolutional LSTM for the detection of violence in videos", *Lect. Notes Comput. Sci.* (including Subser. Lect. Notes Artif. Intell. Lect.Notes Bioinformatics), vol. 11130 LNCS, 2019, pp. 280–295.

[9] S. Soner, R. Litoriya, and P. Pandey, "Exploring Blockchain and Smart Contract Technology for Reliable and Secure Land Registration and Record Management," *Wireless Personal Communications*, Aug. 2021.

[10] C. Dhiman and D. K. Vishwakarma, "A review of state-of-the-art techniques for abnormal human activity recognition," *Eng. Appl. Artif. Intell.*, vol. 77, no. 2018 Aug, 2019, pp. 21–45.

[11] S. Soner, A. Jain, A. Tripathi, R. Litoriya, "A novel approach to calculate the severity and priority of bugs in software projects", *2nd International conference on education technology and computer,* vol. 2, 2010, pp. V2-50-V2-54. 10.1109/ICETC.2010.5529438.

[12] O. Kliper-Gross, T. Hassner, and L. Wolf, "The action similarity labeling challenge", *TPAMI*, vol. 99, 2012.

[13] Q. Dai et al., "Fudan-Huawei at MediaEval 2015: Detecting violent scenes and affective impact in movies with deep learning", *CEUR Workshop Proc.*, vol. 1436, 2015, pp. 5–7.

[14] Y. Pritch, S. Ratovitch, A. Hendel, and S. Peleg. "Clustered synopsis of surveillance video", In *Advanced Video and Signal Based Surveillance*, 2009, pp. 195–200.

[15] R. Retoliya, S. Soner, "RSA Based Threshold Cryptography for Secure Routing and Key Exchange in Group Communication", *International Conference on Advances in Communication, Network, and Computing,* vol. 142, pp. 624-627.

[16] E.Y. Fu, H.V. Leong, G. Ngai, S.C.F. Chan, „Automatic fight detection in surveillance videos", *Int. J. Pervasive Comput. Commun*., vol. 13, no. 2, 2017, pp. 130–156.

[17] T. Senst, V. Eiselein, A. Kuhn, T. Sikora, "Crowd violence detection using global motion-compensated Lagrangian features and scale-sensitive video-level representation", *IEEE Trans. Inf. Forensics Secur.,* vol. 12, no. 12, 2017, pp. 2945–2956.

[18] A. Hanson, K. PNVR, S. Krishnagopal, L. Davis, "Bidirectional convolutional LSTM for the detec-

tion of violence in videos", in: *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018.

[19] X. Zhai, A. Oliver, A. Kolesnikov, L. Beyer, "S4L: Self-supervised semi-supervised learning", in: *Proceedings of the IEEE/CVF International Conference on Computer Vision* (ICCV), 2019.

[20] C. Borrelli, P. Bestagini, F. Antonacci, A. Sarti, S. Tubaro, "Automatic reliability estimation for speech audio surveillance recordings", in: *The IEEE International Workshop on Information Forensics and Security*, WIFS, 2019.

[21] K. Gkountakos, K. Ioannidis, T. Tsikrika, S. Vrochidis, I. Kompatsiaris, "Crowd Violence Detection from Video Footage", *2021 International Conference on Content-Based Multimedia Indexing (CBMI)*, INSPEC Accession Number: 20729035, 10.1109/CBMI50038.2021.9461921

[22] K. Gkountakos, K. Ioannidis, T. Tsikrika, S. Vrochidis, and I. Kompatsiaris, „A crowd analysis framework for detecting violence scenes", *Proceedings of the 2020 International Conference on Multimedia Retrieval*, 2020, pp. 276-280.

[23] S. Soner, A. Jain, D. Saxena, "Metrics calculation for deployment process", *2010 2nd International Conference on Software Technology and Engineering*, 2010. 10.1109/ICSTE.2010.5608760

[24] M. Sharma, R. Baghel, "Video Surveillance for Violence Detection Using Deep Learning", In: *Advances in Data Science and Management*, Springer: Berlin/Heidelberg, Germany, 2020, pp. 411–420.

[25] E. Ditsanthia, L. Pipanmaekaporn, and S. Kamonsantiroj, "Video Representation Learning for CCTV-Based Violence Detection", *TIMES-iCON 2018 - 3rd Technol. Innov. Manag. Eng. Sci. Int. Conf.,* 2019, pp. 1–5.

[26] F. U. M. Ullah, A. Ullah, K. Muhammad, I. U. Haq, and S. W. Baik, "Violence detection using spatiotemporal features with 3D convolutional neural network", *Sensors* (Switzerland), vol. 19, no. 11, 2019, pp. 1–15.