CrimeGuard: Real-time Violence Detection and Predictive Analysis System

1st Shreya Goswami
Computer Science Engineering (AIML)
KIET Group of Institutions, Delhi-NCR
Ghaziabad, UP, India
shreya.2125csme1036@kiet.edu

2nd Khushi Bansal
Computer Science Engineering (AIML)
KIET Group of Institutions, Delhi-NCR
Ghaziabad, UP, India
khushi.2125csme1013@kiet.edu

3rd Mahi Tyagi
Computer Science Engineering (AIML)
KIET Group of Institutions, Delhi-NCR
Ghaziabad, UP, India
mahityagi1222@gmail.com

4th Rajeev Kumar Singh
Associate Professor
Computer Science Engineering (AIML)
KIET Group of Institutions, Delhi-NCR
Ghaziabad, UP, India
rajeev.kumar.csai@kiet.edu

5th Dr Puneet Garg
Associate Professor
Computer Science Engineering (AI)
KIET Group of Institutions, Delhi-NCR
Ghaziabad, UP, India
puneetgarg.er@gmail.com

***Abstract*—This research presents CrimeGuard, a real-time violence detection system designed to enhance public safety by automating video surveillance analysis. The system processes live and recorded video feeds to classify activities as either violent or non-violent using deep learning architectures, including CNN+LSTM, MobileNetV2+LSTM, VGG16+LSTM, VGG19+LSTM, and ResNet50+LSTM. Among these models, CNN+LSTM achieved the highest accuracy of 96%. The backend, built using Flask, facilitates video processing and automated email alerts, while a React-based frontend ensures seamless user interaction. Additionally, the system integrates predictive modeling and visualization techniques to assist law enforcement in data-driven decision-making. The results demonstrate CrimeGuard's effectiveness in real-time crime monitoring, providing proactive alerts to aid law enforcement and security agencies.**

***Index Terms*—Violence Detection, Crime Analysis, Deep Learning, Predictive Modeling, Video Surveillance, Automated Alerts, Crime Prevention**

I. INTRODUCTION

Crime and violence remain persistent challenges, demanding real-time monitoring and intelligent analysis to enhance public safety. Traditional surveillance systems rely heavily on manual intervention, which is inefficient in preventing violent incidents. Similarly, crime data is often underutilized, limiting law enforcement agencies' ability to detect trends and prevent future crimes.

To address these issues, **CrimeGuard** integrates real-time violence detection from video feeds and crime pattern analysis using machine learning and deep learning techniques. The system processes live and recorded videos to identify violent activities using **CNN**, **MobileNetV2**, **VGG16**, **VGG19**, and **ResNet50** along with **LSTM** models. Upon detecting violence, it sends automated email alerts to authorities for quick intervention. Additionally, it analyzes historical crime data (**9,840** records) to identify trends, enabling better law enforcement strategies.

The violence detection model processes frames using feature extraction and sequence modeling, distinguishing between violent and non-violent activities with high accuracy. Meanwhile, crime analysis leverages statistical techniques and forecasting models to uncover regional crime patterns. The system is designed for scalability, ensuring adaptability to diverse surveillance environments and crime datasets.

By combining **computer vision**, **deep learning**, and **statistical analysis**, CrimeGuard offers an advanced framework for proactive crime detection and prevention, assisting law enforcement in maintaining public safety.

II. RELATED WORK

Crime detection and analysis have been the focus of numerous studies, leveraging machine learning, deep learning, and statistical models to improve accuracy and efficiency in identifying criminal activities. Traditional methods relied on manual data collection and statistical techniques, which often lacked scalability and real-time applicability [1]–[3].

Several studies [4]–[7] have utilized deep learning-based models for violence detection, employing **Convolutional Neural Networks (CNNs)** and **Recurrent Neural Networks (RNNs)** for video classification. **CNN+LSTM** architectures have demonstrated improved performance by capturing both spatial and temporal features in violent activity recognition. **MobileNetV2** and **ResNet50** have also been explored for their lightweight architectures and efficient feature extraction capabilities [8], [9]. Despite promising results, many models face challenges such as high computational costs and difficulty in generalizing across different environments.

Other works [10], [11] focus on crime data analysis using predictive models. Some studies employ time-series forecasting models like **ARIMA** and **LSTMs** to predict future crime occurrences based on historical data [12], [17]. However, challenges such as data imbalance and regional biases remain prevalent in crime prediction research.

Real-time alerting systems have also been explored, integrating cloud-based solutions with machine learning models to provide instant notifications upon detecting criminal activities [13], [14]. Implementing automated reporting systems and email notifications in specific frameworks has improved response efficiency in law enforcement applications.

Various limitations such as limited data availability, misinterpretation of video, and other real-world challenges are still present. It is necessary to address these challenges by improving current models and developing solutions to enhance their efficiency [15], [16].

III. Proposed Model

The model integrates crime analysis, prediction and real-time violence detection. It utilizes data analytics, statistical methods and deep learning technologies. This helps in better decision-making and in implementing preventive measures towards improving public safety against crimes.

A. Data Collection and Pre-processing

The proposed system involves collecting data related to crime including historical documents. The data is pre-processed and normalized to improve accuracy. Duplicate values are eliminated and missing values are handled to ensure consistency.

- **Feature Selection**: The purpose of feature selection is to improve the accuracy of prediction. The major features for this study are relevant crime indicators such as type of crime, state, district, and year.
- **Data Cleaning and Transformation**: This step is very critical in data preprocessing. In this step, we use procedures for data files. Normalization of numerical properties, handling of missing values, and encoding of categorical functions directly impact the result. We need data in a suitable format, so data transformation is required.
- **Data Visualization**: To simplify complex data, we use heatmaps, bar charts, and line graphs. Generally, these tools provide graphic representations in this study and are used to examine crime trends over time and in various places.
- **Time-Series Forecasting**: To predict future values, historical patterns are required. ARIMA model is used to forecast future crime incidents for this study. By using ARIMA, we analyze the crime data on the basis of past history. This is helpful in strategic law enforcement, like identifying crime trends and hotspots for the upcoming years.

B. Feature Extraction

Deep learning models automatically extract the features from data. Pre-trained models like CNN, VGG16, VGG19, ResNet50, and MobileNetV2 in association with LSTM are used to extract significant features from video frames. These models can be fine-tuned for real-time monitoring. LSTM extracts temporal dependency from data and concentrates on important trends associated with violent acts.

1) Key Feature Extraction Techniques:

- **Frame-Wise Feature Extraction:** To identify aggressive behavior or unusual activity, CNN-based models like VGG16, VGG19, ResNet50, and MobileNetV2 detect edges, textures, and objects in each video frame. These models can identify important features across multiple frames.
- **Temporal Relationship Modeling by LSTM:** It is used to capture relationship between events over time. The LSTM layer tracks temporal changes across frames and handles sequential data by preserving long-term dependencies. It captures patterns that help indicate violence and allow to understand change of motion over time.
- **Spatio-Temporal Fusion:** This integration combines spatial features and temporal dependencies in data. Characteristics such as shapes, textures, and objects are extracted from frames while also tracking changes in motion over time. Using CNN and LSTM together allows for better differentiation between violent and non-violent activities, improving detection accuracy.
- **Key Indicators of Violence:** The model helps to identify patterns that have distinct movement patterns(aggressive behaviours) like rapid fluctuations or posture changes that are different from normal behaviour.

C. Model Training and Inference

Real-time violence detection and crime trend forecasting are the two main functions that the system is trained to accomplish.

1) Violence Detection (Deep Learning): Deep learning-based models, like CNNs, and temporal dependency models like LSTM can analyze spatial and temporal patterns in video sequences. To improve accuracy for the detection of violence or crime prediction, CNN extracts visual details like textures, shapes, and object structures from each frame. The current model will help to identify key spatial features as well as detect motion patterns. LSTM processes the sequence of frames to detect movement changes, dependencies, and motion flow over time. By working together, these models provide a more accurate understanding of actions and behaviors in videos.

a) Mathematical Formulation for Violence Detection: Given an input video sequence represented as frames F_t, where $t = 1, 2, \ldots, T$, the spatial feature extraction using a CNN is defined as:

$$S_t = \text{CNN}(F_t), \tag{1}$$

where S_t represents the extracted spatial features at time t.

To capture temporal dependencies, the LSTM processes sequential feature vectors:

$$h_t = \sigma(W_h \cdot S_t + U_h \cdot h_{t-1} + b_h), \tag{2}$$

where h_t is the hidden state at time t, W_h, U_h, and b_h are trainable parameters, and σ is the activation function.

The final classification probability is obtained using a softmax function:

$$P(y \mid F) = \text{Softmax}(W_o \cdot h_T + b_o), \tag{3}$$

where W_o and b_o are output layer parameters.

2) Crime Trend Forecasting (Time-Series Analysis): Crime trends over time are modeled using statistical forecasting techniques. Given a time series of crime incidents C_t, the future crime count is estimated:

$$C_t = \phi_1 C_{t-1} + \phi_2 C_{t-2} + \cdots + \phi_p C_{t-p} + \epsilon_t, \tag{4}$$

where ϕ_i are autoregressive coefficients and ϵ_t represents random noise.

To model seasonality and trend components, an **ARIMA** model is employed:

$$(1 - \sum_{i=1}^{p} \phi_i L^i)(1 - L)^d C_t = (1 + \sum_{j=1}^{q} \theta_j L^j)\epsilon_t, \tag{5}$$

where L is the lag operator, d is the differencing order, and θ_j are moving average coefficients.

D. Data visualization

Crime data, including historical records and predictive forecasts, are visualized through various graphical representations:

- **Heatmaps**: Show geospatial crime distribution.
- **Bar and Pie Charts**: Provide statistical breakdowns of crime types and occurrences.

E. Interactive Dashboard

An interactive dashboard is developed to allow users, including law enforcement and policymakers, to dynamically analyze crime trends. The dashboard provides:

- **Customizable Filters**: Users can filter data based on state, district, year, and crime category.
- **Real-Time Violence Detection**: Integrates a real-time surveillance system that supports both **live camera feeds** and **file uploads** for detecting violent activities.
- **Graphical Insights**: Interactive charts and visualizations enable an intuitive exploration of crime patterns.

F. Algorithm Flow

- **Crime Data Preprocessing**: Load and clean crime data, normalize features, and split data for training and testing.
- **Crime Trend Prediction (ARIMA)**: Use historical crime data for forecasting trends using the ARIMA model.
- **Video Preprocessing**: Extract frames from videos, preprocess frames, and resize for CNN processing.
- **Violence Detection (CNN-LSTM)**: Use CNN to extract spatial features and LSTM to analyze temporal dependencies for violence classification.
- **Alert System**: Trigger alerts if violence is detected via email notification.

The following pseudo code provides a high-level overview:

```
# Data Preprocessing (Crime Analysis)
LOAD crime_data.csv
HANDLE missing values
SPLIT dataset (80% train, 20% test)

# Model Training (Crime Trend Forecasting)
FOR each state:
    FIT ARIMA on historical crime data
    FORECAST crime trends for next 5 years

# Video Preprocessing (Violence Detection)
LOAD video
EXTRACT frames, preprocess (resize,
normalize)

# Model Inference (Violence Detection)
LOAD CNN model (MobileNetV2/VGG16/VGG19/
CNN/ResNet50)
FOR each frame:
    EXTRACT features with CNN
    INPUT features to LSTM for violence
    prediction
    IF violence detected, trigger alert

# Backend API (Flask)
DEFINE '/upload_video' METHOD POST:
    RECEIVE video, preprocess frames,
    apply violence detection, return status

DEFINE '/send_alert' METHOD POST:
    SEND email notification if violence detected
```

IV. Evaluation and Results

This section presents the performance evaluation of the proposed violence detection models. Various deep learning architectures were tested on the dataset, and their accuracy, precision, recall, and F1-score were recorded.

A. Evaluation Metrics

To assess the effectiveness of the models, the following evaluation metrics were used:

- **Accuracy:** Measures the overall correctness of the model and is given by:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

- **Precision:** Represents the proportion of correctly predicted positive instances among all predicted positives:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7}$$

- **Recall:** Measures the proportion of actual positive instances correctly identified:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{8}$$

- **F1-score:** Provides a harmonic mean of precision and recall:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

Here, TP, TN, FP, and FN denote the true positives, true negatives, false positives, and false negatives, respectively.

B. Model Performance

To compare different deep learning architectures, multiple models were trained and tested. The table below presents the results:

TABLE I
PERFORMANCE METRICS OF DIFFERENT MODELS

Model	Accuracy (%)	Precision	Recall	F1-score
CNN+LSTM	**96.0**	0.95	0.96	0.96
MobileNetV2+LSTM	93.0	0.93	0.93	0.93
ResNet50+LSTM	60.0	0.61	0.60	0.60
VGG16+LSTM	89.0	0.89	0.89	0.89
VGG19+LSTM	85.0	0.85	0.85	0.85

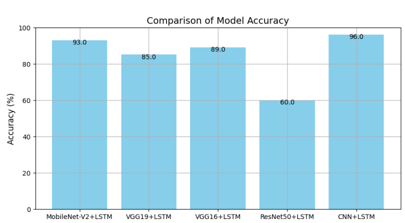

Fig. 1. Comparison of Model Accuracy for Violence Detection

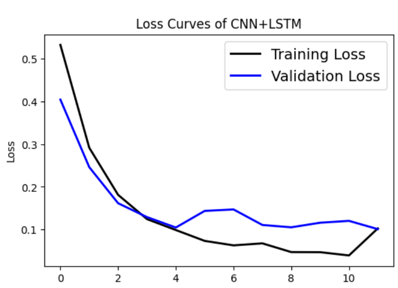

Fig. 2. Loss Curve of CNN+LSTM model

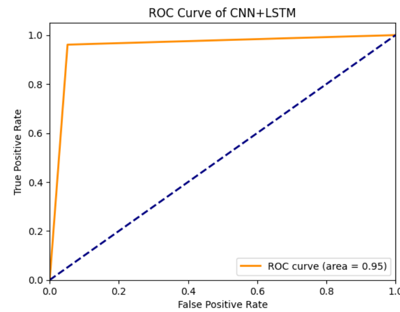

Fig. 3. Receiver Operating Characteristic (ROC) Curve of CNN+LSTM model

C. Computational Complexity

The computational efficiency of each model was assessed in terms of training time and inference time. The results are summarized in Table II.

TABLE II
COMPUTATIONAL COMPLEXITY (IN SECONDS)

Model	Training Time (s)	Inference Time (s)
CNN+LSTM	2520	0.014084
MobileNetV2+LSTM	1025	0.028255
ResNet50+LSTM	604	0.143210
VGG16+LSTM	383	0.035911
VGG19+LSTM	2058	0.043785

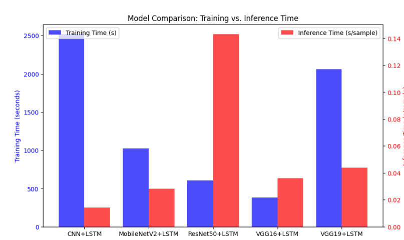

Fig. 4. Computational Complexity (Training and Inference Time)

D. Observations

From the results, the following observations can be made:

- The **CNN+LSTM** model achieved the highest accuracy (96.0%), outperforming all other models in precision, recall, and F1-score. Figure 1 highlights CNN+LSTM as the best-performing model.
- The **MobileNetV2+LSTM** model performed exceptionally well with 93.0% accuracy, making it a strong lightweight alternative for real-time applications.
- Hybrid CNN-LSTM architectures like **VGG16+LSTM** and **VGG19+LSTM** performed reasonably well, achieving 89.0% and 85.0% accuracy, respectively.
- **ResNet50+LSTM** underperformed with an accuracy of 60.0%, indicating potential overfitting or inefficiencies in feature extraction for violence detection.
- The loss curve (Figure 2) depicts smooth reduction, validating the optimization process.

- The ROC curve (Figure 3) indicates a high area under the curve (AUC) for CNN+LSTM, further proving its superior predictive performance.
- **Computational Complexity:** As shown in Table II and Figure 4, CNN+LSTM has the highest training time (2520s) but the lowest inference time (0.014s), making it ideal for real-time violence detection.
- MobileNetV2+LSTM provides a balance between computational efficiency and accuracy, achieving 93.0% accuracy while maintaining a relatively low training time of 1025s.
- ResNet50+LSTM had a very high inference time (0.143s), making it less suitable for real-time applications.
- Future optimizations may focus on reducing training complexity while maintaining high detection accuracy.

E. Result

The evaluation results demonstrate that deep learning models, particularly CNN+LSTM, significantly enhance violence detection accuracy. These findings support the use of hybrid architectures in real-world surveillance systems, improving public safety through automated violence detection.

V. Conclusion and Future Scope

The proposed system enhances public safety by combining predictive crime analysis with real-time violence detection using spatial feature extraction with motion analysis. Violent incidents can be detected in real-time generating alerts so that faster response can be taken to prevent them. This allows for better crime analysis and in taking preventive measures in advance. Real-time violent detection, future crime prediction and data visualizations allow concerned authorities to make policies and strategies in order to reduce crime rates and improve public security. Analyzing trends in data can help in law enforcement and enhancing decision-making.

The results obtained show that deep learning-based violence detection increases accuracy of the model. The CNN+LSTM approach performed the best among other models, highlighting the effectiveness of hybrid models for real-world surveillance. Additionally, time-series forecasting using ARIMA model for predictive crime analysis provides useful analysis into crime trends, helping authorities optimize resource allocation and improve their strategic planning.

A. Limitations

Several limitations are present in spite of promising results obtained through the proposed system:

- **Dataset Constraints:** The efficiency of the model depends on data which has limited availability. Efficiency of the model depends on quality and diversity of data. Privacy and various restrictions are also the concern. The system may not be able to perform well in practical scenario if dataset size is not sufficient. Restrictions and privacy are also the challenges in crime-related data availability.
- **Environmental Factors:** Environmental actors such as changes in camera angles, objects blocking the view, and poor lighting conditions can affect the accuracy of video-based violence detection. Crowded places also create challenges in identifying violent activities accurately.
- **False Positives and Negatives:** The model can sometimes misinterpret body gestures, making it possible for misclassification of events despite showing high accuracy. There is a need for continuous improvements and refinements in training data to avoid challenges like bias.

B. Future Scope

To improve the effectiveness of the system, future work can explore the following domains:

- **Multimodal Analysis:** Inclusion of audio data for analysis can enhance the accuracy of violence detection. For understanding of various complex environments like crowded place where visuals are not alone enough to take better decision, audio data can be beneficial.
- **Edge Computing:** It is highly useful in real time detection. Deployment of models on edge devices causes faster processing in real-time, reduced latency, lower bandwidth consumption, and improved privacy.
- **Enhanced Predictive Modeling:** Using advanced machine learning and deep learning, like deep reinforcement learning, geo spatial crime analysis, vision transformers, can help make crime predictions more accurate.
- **Partnerships and Collaborations:** Support to government, private sector, civil organisations, smart city developers, and surveillance technology providers which expand its real-world adoption for public safety.

This system creates a framework for AI-driven crime prevention. Improving its accuracy and using advanced analysis, it can help make communities safer and assist law enforcement in providing responding more effectively.

References

[1] M. Sabokrou, M. Fathy, M. Hoseini, and R. Klette, "Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes," *Computer Vision and Image Understanding*, vol. 172, pp. 88-97, 2018.

[2] B. Sathyadevan and S. Gangadharan, "Crime analysis and prediction using data mining techniques," *2014 First International Conference on Networks & Soft Computing (ICNSC)*, 2014, pp. 406-412.

[3] M. Ullah, M. A. Muhammad, G. H. Lee, and S. W. Baik, "Violence detection using spatiotemporal features with 3D convolutional neural networks," *Sensors*, vol. 19, no. 11, p. 2472, 2019.

[4] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138-52160, 2018.

[5] X. Xu, L. Yang, H. Wang, and F. Zhao, "Edge computing: A new computing model for Internet of Things," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 1778-1786, 2020.

[6] N. K. Sharma, M. Pandey, and R. K. Gupta, "IoT-based smart surveillance systems for crime detection and prevention," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 8473-8490, 2021.

[7] G. Gorr and R. Harries, "Introduction to crime forecasting," *International Journal of Forecasting*, vol. 19, no. 4, pp. 551-555, 2003.
[8] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles, "ActivityNet: A large-scale video benchmark for human activity understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 961-970.
[9] T. Hassner, M. Itcher, and R. Kliper-Gross, "Violent flows: Real-time detection of violent crowd behavior," in *Proceedings of IEEE Workshop on Socially Intelligent Surveillance and Monitoring (SISM)*, 2012, pp. 1-6.
[10] Y. Zhang, Z. He, and S. Zhang, "A survey on violence detection from videos," *Multimedia Tools and Applications*, vol. 79, no. 39, pp. 29321-29350, 2020.
[11] P. Sudhakaran and O. Lanz, "Learning to detect violent videos using convolutional long short-term memory," in *Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2017, pp. 1-6.
[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations (ICLR)*, 2015.
[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770-778.
[14] A. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.
[16] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 1993, pp. 207-216.
[17] G. Box, G. Jenkins, and G. Reinsel, *Time Series Analysis: Forecasting and Control*, 5th ed., Wiley, 2015.