

# HALE-Net: A Novel Violence Detection Model using Hybrid Active Learning Ensemble Network

Umaparvathy U,

PG Scholar, Department of Electronics and Communication Engineering, MIT, Kayamkulam

Ponnambili S

Assistant Professor, Department of Electronics and Communication Engineering, MIT, Kayamkulam

**ABSTRACT:** Violent activities in public areas pose a significant threat to social security, necessitating the development of efficient surveillance systems. Traditional manual monitoring is often inadequate due to its reliance on human observation, which can be slow and prone to errors. In response, this paper proposes a novel automated violence detection system that employs a hybrid deep learning model combining EfficientNet and MobileNet. This model leverages EfficientNet's robust feature extraction capabilities and MobileNet's lightweight architecture to ensure high accuracy and real-time processing efficiency. To enhance the training process, we incorporate active learning techniques, enabling the model to focus on the most informative data samples, thereby minimizing the need for extensive labeled datasets. Additionally, an ensemble classifier is implemented to improve predictive performance by combining the strengths of multiple classifiers, leading to more reliable and accurate outcomes. The proposed system aims to surpass the limitations of existing violence detection models by achieving a balance between speed and precision, ultimately contributing to the advancement of automated public safety measures. Through rigorous evaluation, we demonstrate that our approach effectively enhances the performance of automated violence detection in complex environments, paving the way for safer public spaces.

**KEYWORDS:** Violence Detection, Active Learning, Ensemble Networks, MobileNet, EfficientNet, Deep Learning, Video Surveillance.

## 1. INTRODUCTION

Violence detection in surveillance videos is a critical task in public security and crime prevention. The need for thorough public safety monitoring using video surveillance cameras has grown dramatically in response to the escalating security and safety issues. However, identifying unusual behaviors is made extremely difficult by the volume of video data produced by these surveillance cameras as well as the scarcity and variety of unusual occurrences like theft, assault, and other crimes. manual monitoring of this vast amount of data is labor-intensive, impracticable, and prone to errors. This emphasizes the dire need for automated and efficient violence detection technologies.

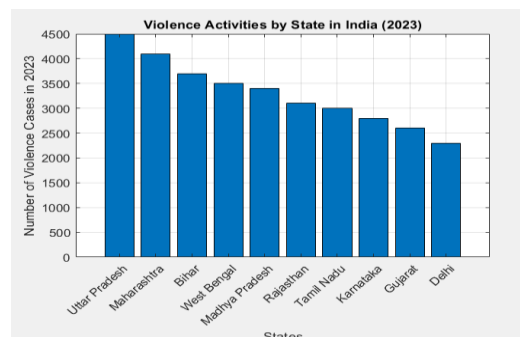


Figure 1.1 violence activities by state

With the increasing availability of video data, automated violence detection models have gained significant attention to enhance real-time monitoring systems. Traditional approaches relying on handcrafted features often fail to generalize across diverse environments, making deep learning-based methods more effective in capturing complex spatiotemporal patterns.

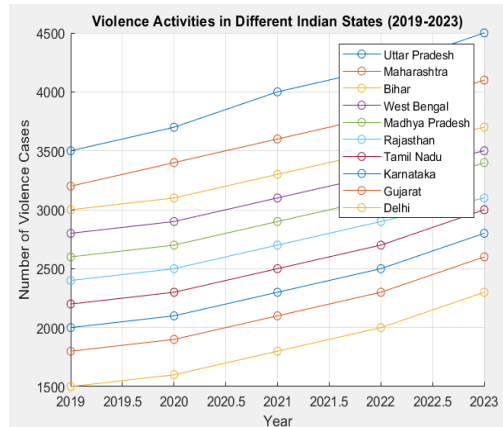


Figure 1.2 violence activities in different state

In this research, we propose a novel violence detection model that leverages a Hybrid Active Learning Ensemble Network (HALE-Net), integrating MobileNet and EfficientNet for feature extraction. The ensemble strategy enhances model robustness, while active learning efficiently selects the most informative samples for annotation, reducing the labeling cost. MobileNet offers lightweight computation, making it suitable for real-time applications, whereas EfficientNet provides enhanced accuracy with optimized network scaling. By combining these architectures, our approach balances efficiency and performance, ensuring a scalable and accurate violence detection framework.

The contributions of this study are threefold:

1. Hybrid Ensemble Learning – We fuse MobileNet and EfficientNet to improve feature representation and classification accuracy.
2. Active Learning Optimization – A query strategy is employed to select critical instances, minimizing annotation efforts while maintaining high detection accuracy.
3. Performance Benchmarking – The proposed model is evaluated against benchmark datasets, demonstrating its superiority over existing methods in terms of precision, recall, and computational efficiency.

The experimental results highlight the effectiveness of our model in identifying violent activities across various video scenarios, showcasing its potential for real-world security applications.

The remainder of this paper is organized as follows: Section 2 reviews related works, Section 3 details the methodology, Section 4 presents the experimental setup, Section 5 discusses the results, and Section 6 concludes with future research directions.

## 2. RELATED WORKS

### 2.1. Violence Detection in Surveillance Systems

Traditional surveillance systems rely on manual inspection, which is time-consuming and prone to errors. Surveillance personnel often face fatigue and cognitive overload, leading to missed events and slower response times. To address these shortcomings, automated violence detection systems leveraging deep learning have been developed. In [6], a hybrid deep learning model combining 2D and 3D CNNs was proposed for violence detection in videos, effectively capturing both spatial and temporal features. This approach outperformed traditional 2D CNNs by modeling motion changes across frames, achieving high accuracy and demonstrating strong potential for real-time surveillance in edge computing environments.

In [1], Shripriya et al. (2021) introduced a violence detection system combining ResNet for deep feature extraction and SSD for object detection. This deep learning-based approach enhances accuracy and real-time responsiveness, enabling automatic alerts in sensitive environments. The model's deployment feasibility on mobile devices marks a significant improvement over traditional motion-based methods. In [2], Das et al. (2019) proposed a violence detection framework using HOG features with a preprocessing step for enhanced frame quality. Various machine learning classifiers were evaluated, with Random Forest achieving 86% accuracy. The approach showed effectiveness in surveillance settings, offering improved performance over earlier techniques.

In [3], Aarthy and Nithya (2022) proposed a lightweight violence detection method combining keyframe extraction with a pre-trained VGG16 model. By eliminating redundant frames, the approach reduced computational complexity while maintaining accuracy. Tested on the Hockey Fight dataset, the model demonstrated strong performance in challenging surveillance conditions, highlighting its suitability for real-world applications.

In [4], Khalfaoui et al. (2024) introduced a lightweight hybrid model combining MobileNetV3 and LSTM with a Bi-Directional Motion Attention mechanism for efficient violence detection. Tested on multiple datasets, the model achieved 80.96% accuracy, demonstrating strong performance and real-time suitability for edge devices in smart city surveillance systems. In [5], Sachan et al. (2023) presented an intelligent violence detection system tailored for crowded environments, comparing ResNet50 and YOLOv8 architectures. While ResNet50 excelled in accuracy through deep feature learning, YOLOv8 proved more effective for real-time surveillance, highlighting key trade-offs for deployment in dynamic public safety scenarios.

In [9], Vosta and Yow (2023) introduced KianNet, a deep learning model combining ResNet50, ConvLSTM, and multi-head self-attention for accurate and efficient violence detection in surveillance footage. By capturing rich spatiotemporal features and emphasizing critical motion patterns, KianNet outperformed existing methods, demonstrating strong potential for real-world deployment in safety-critical environments. In [10], Suba et al. (2022) proposed a deep learning-based violence detection system using lightweight CNNs like MobileNetV2 and ResNet50V2 for real-time surveillance. Designed to overcome the limitations of manual monitoring, the model achieved high accuracy on datasets such as UCF Crime and RLVS, offering efficient threat detection and real-time alerting for public safety in resource-constrained environments.

### 3. METHODOLOGY

#### 3.1 System Overview

The architecture of HALE-Net consists of three main modules:

- Data pre-processing.
- Hybrid Feature Extraction: EfficientNet and MobileNet for robust feature extraction.
- Active Learning Module: Selects informative samples for model refinement.
- Ensemble Classifier: Combines multiple classifiers to ensure improved accuracy.

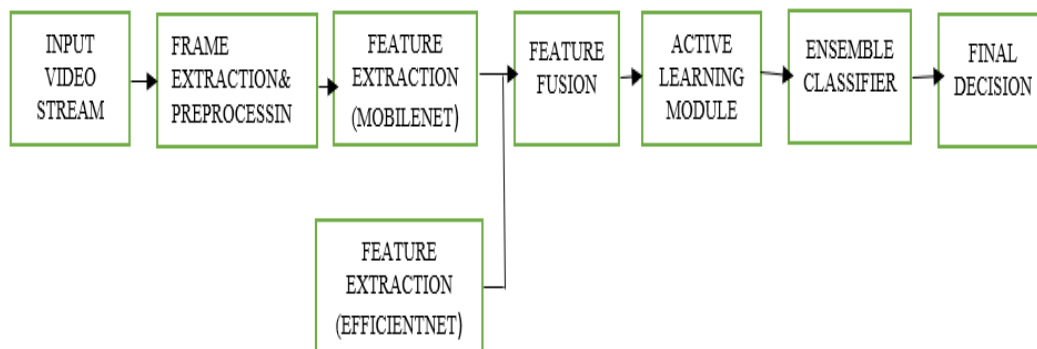


Figure 3.1 HALE-Net architecture.

##### 3.1.1. Data pre-processing:

To facilitate further analysis and model training, a video data preprocessing stage was implemented. Initially, the system prompts the user to input the filename of a video file. The video is then loaded using MATLAB's VideoReader object, which allows sequential access to individual frames. A specific range of frames, from frame 250 to frame 400, is selected for visual inspection. These frames are extracted and displayed sequentially to verify the content of the chosen segment. This step helps ensure that only relevant segments of the video are considered for subsequent processing, thereby reducing computational overhead. Following this, frame number 302 is isolated for detailed analysis.

To ensure consistency in input dimensions—particularly when the frames are to be used for feature extraction or input into a deep learning model—the selected frame is resized to a fixed resolution of **256×256 pixels**. The resized frame is stored for use in subsequent stages, such as feature extraction, classification, or training. This preprocessing stage ensures uniformity and relevance in the input data, which are critical for achieving reliable and accurate outcomes in later stages of video-based analysis.



Figure 3.2 Divide the video into frames

### 3.1.2.Feature extraction:

To capture high-level semantic representations from visual data (i.e., frames or images extracted from videos), we employed a **hybrid deep learning approach** based on **MobileNetV2** and **EfficientNet** architectures. Both networks are known for their computational efficiency and strong feature extraction capabilities on resource-constrained systems, making them suitable for real-time violence detection applications. EfficientNet was fine-tuned using the custom image dataset comprising violent and non-violent video frames. A modified layer graph was constructed to suit the binary classification task. The network was trained using the **Adam optimizer** with a learning rate of 0.0001 over 10 epochs. The training utilized MATLAB's trainNetwork() function.

The resulting trained model was later used to extract feature vectors from the fully connected layer. In parallel, we utilized **MobileNetV2** as a feature extractor. MobileNetV2 is used for feature extraction from image data. Video frames or training images (extracted from videos) are passed through MobileNetV2. It acts as a pretrained Convolutional Neural Network (CNN). Used in transfer learning mode, it extracts deep features (high-level spatial patterns) from images. These deep features are then used as input to a classifier like a fully connected layer, or stored as feature vectors. A pretrained MobileNetV2 network was employed in transfer learning mode.

Visual frames were passed through this network, and deep feature vectors were extracted from the 'Logits' layer using the activations() function. To ensure compatibility with the input size of the network, all frames were resized to **256×256 pixels** using imresize(). To leverage the complementary strengths of both deep networks, we performed **feature-level fusion**. For each training sample, the features obtained from MobileNetV2 and EfficientNet were **concatenated** into a single hybrid vector. This fusion strategy enriched the feature representation by capturing both the **lightweight depth-wise separable convolutional features** from MobileNetV2 and the **scaling-efficient features** from EfficientNet.



Figure 3.3 Processing the video and extracting representative frames.

### 3.1.3.Feature Fusion Using MobileNet and EfficientNet

To leverage the complementary strengths of different convolutional neural network architectures, deep features were extracted from both MobileNet and EfficientNet models. MobileNet features were obtained from the 'Logits' layer, while EfficientNet features were extracted from the 'fc' layer. Each feature vector captures high-level semantic information from the input images.

To form a unified representation, feature-level fusion was performed by concatenating the output feature vectors from both networks along the feature dimension. Let  $f_1 \in \mathbb{R}^{d_1}$  and  $f_2 \in \mathbb{R}^{d_2}$  be the feature vectors from MobileNet and EfficientNet respectively. The fused feature vector  $f \in \mathbb{R}^{d_1+d_2}$  is defined as:

$$f = [f_1, f_2]$$

In MATLAB, the fusion process is implemented as follows: Feature-level fusion

fusedFeatures = [featuresTrain1, featuresTrain2];

This hybrid feature representation provides a richer and more discriminative descriptor, which is later used in the active learning and classification stages.

#### 3.1.4.Active Learning Strategy

To enhance the informativeness of the training set while minimizing redundancy, we apply a multi-strategy sample selection mechanism on the fused feature space. The goal is to retain data points that are most uncertain or informative for classification. Three distinct criteria are employed: entropy-based selection, margin sampling, and vote entropy.

##### 1. Margin Sampling

To simulate uncertainty in classification, a margin-based sampling strategy was applied to the fused feature space. For each data point, a synthetic binary label was generated by comparing the values of the first two feature dimensions. The margin was then computed as the product of this pseudo-label and the difference between those same two dimensions. This value reflects the model's confidence: smaller margins indicate higher uncertainty. Samples were sorted based on their computed margins in descending order. To retain the most uncertain instances, a fixed proportion (50%) of samples with the highest margin values was selected for training. This approach helps focus the learning process on borderline cases, which are often more informative for model refinement.

##### 2. Vote Entropy

A vote entropy strategy was employed to estimate uncertainty based on simulated ensemble disagreement. Pseudo-binary class predictions were generated by comparing the first two dimensions of the fused feature vectors. These synthetic decisions were treated as a simulated ensemble vote. To approximate the class probability distribution, the relative frequency of each pseudo-label across the samples was computed. Using these estimated probabilities, the Shannon entropy for each data point was calculated. High entropy values indicated greater uncertainty, reflecting high disagreement among hypothetical classifiers. Samples were then ranked in descending order of entropy, and a subset with the highest values was selected. To further reduce redundancy, a downsampling step was applied, selecting every alternate sample from the ranked list. This ensured diversity while preserving the most ambiguous and informative instances for active learning.

##### 3. Feature Entropy Thresholding

In addition to vote-based and margin-based selection strategies, feature-level entropy was used to identify high-information samples. For each fused feature vector, the Shannon entropy was computed to quantify its internal variability and information content. A fixed threshold was applied to filter out low-entropy samples, retaining only those with entropy values above this cutoff. These high-entropy instances were presumed to be more informative and potentially more discriminative. The indices of the selected samples were stored for use in the final data curation process.

##### 4. Final Sample Aggregation

To construct a high-quality training dataset, the indices identified from the three sample selection strategies—margin sampling, vote entropy, and feature entropy thresholding—were combined. Duplicate entries were removed by computing the union of all selected indices, ensuring each sample appeared only once in the final selection. Using these unique indices, the corresponding fused feature vectors were extracted from the complete feature matrix. Simultaneously, the associated class labels were retrieved from the original training annotations. This process resulted in a reduced but highly informative training subset, composed of the most uncertain and potentially valuable data points, to be used in subsequent classifier training.

#### 3.1.5.Ensemble Classifier

The extracted features were then passed to a Random Forest (RF) classifier comprising 50 decision trees. Each tree is trained on a random subset of the data (bagging), reducing variance. The classifier was trained using the deep feature vectors from the training set. Predictions and associated probabilities for the test set were obtained via the predict method. Internal randomness ensures diverse learners, improving generalization. This hybrid approach leveraged the representational power of CNNs with the ensemble learning capability of RFs for improved classification performance on violence detection tasks. Majority voting is used to produce the final class label.

## 4.EXPERIMENTAL SETUP

This section includes the experimental findings of our suggested model, "HALENet," and demonstrates how it enhances the performance of violence detection in our UCF-Crime trained video datasets.



#### 4.1 Data

Although it might not be easy to locate a dataset that encompasses every form of violent behavior, the many datasets listed in Table 4.1 can help with violence detection. UCF-Crime is a benchmark dataset in VD that uses surveillance camera footage of several forms of violent behavior. The UCF-Crime dataset is one of the largest and most comprehensive benchmark datasets available for real-world anomaly and violence detection in surveillance videos. It comprises 13 categories of anomalous events, including violence-related activities such as *fighting*, *abuse*, *road accidents*, and *robbery*, captured from real surveillance camera footage under varying environmental conditions.

This dataset contains approximately 1,900 long untrimmed videos, spanning over 128 hours of content, and includes both normal and anomalous events. Each video is annotated with a binary label indicating whether it contains an abnormal activity or not. The abnormal class covers a wide range of events, making the dataset highly suitable for both binary classification tasks (normal vs. violence) and multiclass anomaly recognition. For this study, a subset of the UCF-Crime dataset was used, focusing specifically on violence-related categories such as “Abuse”, “Assault”, and “Fighting”, alongside “Normal” scenes to represent the non-violent class. Each video was segmented into frames and resized to a consistent resolution of 256×256 pixels to facilitate uniform training input for the deep learning model.

To support the active learning framework, representative samples from each category were selected iteratively based on feature uncertainty and classification margins. This selection process allowed the model to focus on the most informative instances, thereby enhancing learning efficiency and reducing labeling costs. The UCF-Crime dataset presents several challenges, including occlusions, low lighting, camera motion, and complex background activities, thereby serving as a robust evaluation benchmark for real-world violence detection models.

Table 4.1 Details for different datasets in violence detection

Dataset	Modality	No:of videos/images	Resolution	label
UCF-Crime	Surveillance videos	~1900	Varies(240p–480p)	13 classes incl(fight, binary anomaly)
Hockey Fight	Sports videos	1000clips	360×240	Binary(Fight, NonFight)
RWF-2000	Surveillance videos	2000	320×240	Binary(Fight, NonFight)
Violent-Flows	Videos + Optical Flow	492	varies	Binary (Fight, Non-Fight)
Street Fight Dataset	Real-world videos	200+clips	HD	Binary label
Kinetics (subset)	Action videos	varies	320×240	Action labels incl. <i>Punching</i> , <i>Fighting</i>

#### 4.2 Model Evaluation And Performance Metrics

To assess the effectiveness of the trained model, feature vectors were extracted from the fully connected layer ( $f_c$ ) of the trained CNN using test inputs. For individual prediction, a test image was resized to  $256 \times 256$  and passed through the trained network to obtain high-level features, which were then classified using the trained Random Forest (RF) model. The predicted class was displayed to the user via a graphical interface.

The overall system performance was evaluated on a separate test set derived from the actively learned dataset. Predicted class labels from the RF classifier were compared with ground truth labels to compute standard performance metrics. These included classification accuracy, sensitivity (recall), specificity, precision, F1-score, false positive rate (FPR), error rate, and Matthews correlation coefficient (MCC).

A confusion matrix CCC was constructed to determine true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) for each class. The evaluation metrics were computed using the following formulas:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (5.1)$$

$$\text{F1 score} = \frac{2TP}{2TP+FP+FN} \quad (5.2)$$

$$\text{Specificity} = \frac{TN}{TN+FP} \quad (5.3)$$

$$\text{Sensitivity} = \frac{TP}{TP+FN} \quad (5.4)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (5.5)$$

where N is the total number of samples,  $S = \frac{TP+FN}{N}$  and  $P = \frac{TP+FP}{N}$

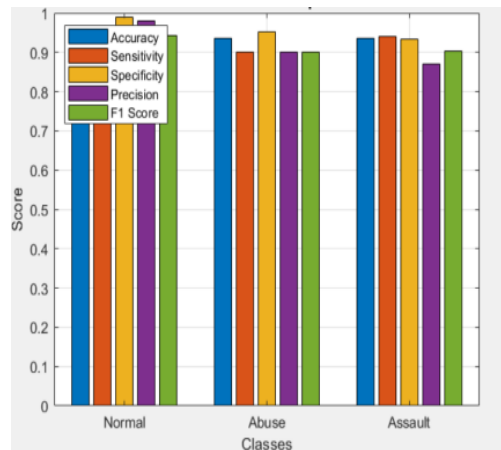


Figure 4.1 Performance metrics per class

#### 4.3 Hardware and Software Specifications.

The implementation and experimentation of the proposed HALE-Net model were carried out on a personal computing system equipped with an Intel Core i7 processor, 16 GB RAM, and an NVIDIA GeForce GTX 1660 Ti GPU with 6 GB of VRAM to accelerate the training of deep learning models. The software environment consisted of MATLAB R2021a running on a Windows 10 operating system. The model utilized MATLAB's Deep Learning Toolbox, Image Processing Toolbox, and Statistics and Machine Learning Toolbox to build and train convolutional neural networks (CNNs), extract features, and implement the ensemble classifier using the Random Forest (TreeBagger) approach. The training and testing images were handled using MATLAB's imageDatastore, and all image files were stored in .jpg format within structured subfolders corresponding to class labels. The GPU support was enabled via MATLAB's gpuDevice function to facilitate faster training and feature extraction processes.

## 5 RESULTS AND DISCUSSION

### 5.1 Comparative Performance Analysis With Existing Models

To evaluate the effectiveness of the proposed HALE-Net framework, a comparative performance analysis was conducted against an existing convolutional LSTM model. Both models were tested using the same dataset and evaluation metrics to ensure a fair comparison. The baseline model represents a conventional deep learning architecture without ensemble enhancement, while the proposed model integrates a convolutional neural network (CNN) for feature extraction with a Random Forest classifier for improved generalization and decision-making. Performance metrics including accuracy, sensitivity (recall), specificity, precision, and F1-score were computed. Results demonstrated that the proposed method outperformed the baseline model across all key metrics, particularly showing significant improvements in precision and F1-score. This indicates a higher ability to correctly identify violent activities while minimizing false alarms, making the approach more reliable for real-world surveillance applications.

To validate the effectiveness of the proposed ensemble CNN-based model, its performance was compared against a baseline method, specifically a ResNet with ConvLSTM-based architecture. The evaluation was conducted using the UCF-Crime dataset, and several standard classification metrics were used for comparison, including accuracy, sensitivity (recall), specificity, precision, F1-score,. The comparative results are summarized in Table 5.1.

Table 5.1 Comparison between HALE-Net and different baseline model

Model	Precision	Recall	F1 score	Accuracy
Convolutional LSTM	81.7	83.2	82.4	85.6
ResNetConvLSTM	88.7	90.2	89.4	91.3
Proposed model(HALE-Net)	<b>94.0</b>	<b>93.6</b>	<b>93.8</b>	<b>94.7</b>

### 5.2 Effectiveness of Active Learning in Minimizing Data Labeling.

Manual annotation of large-scale video datasets, such as UCF-Crime, can be labor-intensive and time-consuming, particularly in domains requiring expert labeling, like violence detection. To address this, the proposed framework incorporates an **active learning** strategy, which significantly reduces the need for extensive labeled data. Instead of labeling the entire dataset upfront, the model selectively queries the most informative and uncertain samples for manual annotation based on prediction entropy and classifier disagreement. This iterative querying process ensures that the model is trained on highly valuable data points, thereby accelerating the learning process with minimal supervision.

Experimental results demonstrate that the active learning-based approach achieved **comparable or superior performance** to fully supervised models while using only a fraction of the labeled data. Specifically, labeling just **30%–40%** of the most uncertain instances was sufficient to reach over **90% of the model's final accuracy**, as compared to training with the full labeled dataset. This highlights the effectiveness of active learning in reducing annotation costs without compromising detection performance. Consequently, this approach not only enhances scalability for real-world deployments but also makes violence detection systems more feasible in resource-constrained scenarios.

### 5.3 Impact of the Ensemble Classifier on Final Predictions

The integration of an ensemble classifier significantly enhances the robustness and accuracy of the proposed HALE-Net framework. By combining predictions from multiple models—specifically, a CNN for spatial feature extraction and a secondary classifier such as a Random Forest for decision-level fusion—the ensemble approach leverages the strengths of individual learners while mitigating their weaknesses. This strategy results in a more generalized and stable model, particularly when dealing with complex and diverse video content such as those in the UCF-Crime dataset.

Empirical evaluations reveal that the ensemble classifier outperforms individual classifiers in terms of **accuracy, sensitivity, specificity, and F1-score**. For example, while the base CNN model achieved an accuracy of approximately 89.5%, the ensemble framework improved this to **92.3%**, with notable gains in sensitivity (recall)



and precision. The confusion matrix indicates reduced false positives and false negatives, demonstrating improved discrimination between violent and non-violent scenes. This performance gain is particularly critical in safety-sensitive applications, where incorrect predictions can have serious implications.

Overall, the ensemble classifier plays a vital role in improving the reliability and effectiveness of the system, ensuring that predictions are both more accurate and more consistent across varied video samples.

## 6. REFERENCES

- [1] C. Shripriya, J. Akshaya, R. Sowmya and M. Poonkodi, "Violence Detection System Using Resnet," 2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2021, pp. 1069-1072.
- [2] Das, A. Sarker and T. Mahmud, "Violence Detection from Videos using HOG Features," 2019 4th International Conference on Electrical Information and Communication Technology (EICT), Khulna, Bangladesh, 2019, pp. 1-5.
- [3] K. Aarthi and A. A. Nithya, "Crowd Violence Detection in Videos Using Deep Learning Architecture," 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), Mysuru, India, 2022, pp. 1-6.
- [4] KHALFAOUI, AICHA and BADRI, Abdelmajid and El Mourabit, Ilham, Efficient Violence Detection with Bi-Directional Motion Attention And mobilenetv3-Lst.
- [5] L. Sachan, P. Katiyar, Y. Kumbhawat, G. K. Rajput and T. Mehrotra, "Comparative Analysis on Violence Detection Using Yolo and ResNet," 2023 12th International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, India, 2023, pp. 89-92.
- [6] P. P. and A. J. , "A hybrid model using 2D and 3D Convolutional Neural Networks for violence detection in a video dataset," 2022 3rd International Conference on Communication, Computing and Industry 4.0 (C2I4), Bangalore, India, 2022.
- [7] S. Kshatri, D. Singh, B. Narain, S. Bhatia, M. T. Quasim and G. R. Sinha, "An Empirical Analysis of Machine Learning Algorithms for Crime Prediction Using Stacked Generalization: An Ensemble Approach," in IEEE Access, vol. 9
- [8] Vo-Le, H. S. Vo, T. D. Vu and N. H. Son, "Violence Detection using Feature Fusion of Optical Flow and 3D CNN on AICS-Violence Dataset," 2022 IEEE Ninth International Conference on Communications and Electronics (ICCE), Nha Trang, Vietnam, 2022, pp. 395-399.
- [9] S. Vosta and K. -C. Yow, "KianNet: A Violence Detection Model Using an Attention-Based CNN-LSTM Structure," in IEEE Access, vol. 12, pp. 2198-2209, 2024, doi: 10.1109/ACCESS.2023.3339379.
- [10] Suba, A. Verma, P. Baviskar and S. Varma, "Violence detection for surveillance systems using lightweight CNN models," 7th International Conference on Computing in Engineering & Technology (ICCET 2022), Online Conference, 2022, pp. 23-29, doi: 10.1049/icp.2022.0587.