A Yolo-based Violence Detection Method in IoT Surveillance Systems

Hui Gao
College of Computer and Information Engineering
Xinxiang University
Xinxiang 453000, Henan, China

***Abstract*—Violence detection in Internet of Things (IoT)-based surveillance systems has become a critical research area due to their potential to provide early warnings and enhance public safety. There have been many types of research on vision-based systems for violence detection, including traditional and deep learning-based methods. Deep learning-based methods have shown great promise in ameliorating the efficiency and accuracy of violence detection. Despite the recent advances in violence detection using deep learning-based methods, significant limitations and research challenges still need to be addressed, including the development of standardized datasets and real-time processing. This study presents a deep learning method based on You Only Look Once (YOLO) algorithm for the violence detection task to overcome these issues. We generate a model for violence detection using violence and non-violence images in a prepared dataset divided into testing, validation, and training sets. Based on accepted performance indicators, the produced model is assessed. The experimental results and performance evaluation show that the method accurately identifies violence and non-violence classes in real-time.**

Keywords*—Violence detection; IoT; surveillance systems; Yolo; deep learning

I. Introduction

The use of Internet of Things (IoT) based surveillance systems has elevated significantly in the past few years, particularly for detecting and preventing violent incidents in public spaces[1–3]. Violence detection in IoT-based surveillance systems has become a critical research area due to its potential to provide early warnings and enhance public safety[4,5]. These systems can process and analyze real-time data from sensors and cameras, enabling quick and efficient identification of violent incidents [6].

There have been significant advances in violence detection technologies in IoT-based surveillance systems in recent years [7,8]. There has been significant research on vision-based systems for violence detection, including traditional and deep learning-based methods [9,10]. Traditional methods, such as motion detection, background subtraction, and object tracking, have been widely used for violence detection in surveillance systems [6,11,12]. However, these methods have limitations in terms of accuracy and robustness, particularly in complex and cluttered environments.

Recent studies have shown that deep learning-based model, like recurrent neural networks (RNNs), convolutional neural networks (CNNs), and YOLO, is able to significantly improve the accuracy and efficiency of violence detection in vision-based systems [13–15]. These models can process and analyze image and video data, extract complex features, and identify violent events in real time.

Despite the recent advances in violence detection, using deep learning-based methods has shown great promise in ameliorating the efficiency and accuracy of violence detection [13,16]. Nevertheless, significant limitations and research challenges still need to be addressed, including the development of standardized datasets and real-time processing algorithms. According to these challenges, the lack of standardized datasets leads to generating inaccurate model for violence detection. Moreover, it is required to address an efficient model to perform in real-time requirement. This makes comparing the different models' performances challenging and limits their generalizability. Addressing this challenge is essential to advance the field and ensure the accurate and efficient detection of violent events in surveillance systems.

To deal with the research challenge in this work, the YOLO algorithm is utilized for the violence detection task in order to overcome these issues. The most recent object identification technique, YOLO, is highly accurate and quick in detecting several items in a picture. We generate a model for violence detection using violence and non-violence images in a dataset that has been divided into testing, validation, and training sets. Based on accepted performance indicators, the produced model is assessed. The system can be taught to recognize violence patterns and accurately identify violence and non-violence classes in real time.

The rest of this paper is structured as follows; Section II presents literature review. Section III discuss about the methodology. Experimental results and performance evaluation presents in Section IV Finally, the paper concludes in Section V.

II. Literature Review

This section presents the literature review and related works on the violence detection research domain. Ullah et al. [3], in IoT-based industrial surveillance networks, this research presented an edge vision technique with AI assistance for violence detection. The technique uses cloud computing, edge devices, and deep learning-based algorithms to analyse video data and identify possible real-time risks. Some important features are custom datasets for training, cloud computing and

edge device integration, and real-time notifications for possible risks. The method successfully detects violent occurrences with low false-positive rates and high accuracy. The method's drawbacks include the sizeable computational resources needed for in-the-moment data processing and analysis, as well as the hefty infrastructure expenditures.

AlDahoul et al. [17] rendered a method for violence detection utilizing a Convolutional Neural Network-Long Short Term Memory (CNN-LSTM) based IoT node. The suggested method utilizes a custom dataset for training the CNN-LSTM model, which analyzes the video data captured by the IoT node to detect violent events. The system can process and analyze data in real-time, quickly detecting potential threats. The key features of the proposed method include the use of a CNN-LSTM model for violence detection, the integration of IoT devices for data capture, and the ability to perform real-time analysis of data. The study found that the proposed approach achieved high levels of accuracy in detecting violent events with low false-positive rates. One limitation of the proposed approach is the potential for high power consumption by the IoT node when processing and analyzing data. The authors also note that the performance of the system may vary based on environmental factors and the specific application scenario.

In [18], the research presented a weakly supervised method for detecting violence in surveillance footage. In order to identify probable violent occurrences without needing manual annotation of the training data, the technique employs a Convolutional Neural Network (CNN) model to categorise video frames as violent or non-violent. The approach's main characteristics are using CNN models for classification and weakly supervised learning, eliminating the requirement for annotated training data. The study discovered that the proposed strategy outperformed earlier state-of-the-art approaches in achieving high accuracy in identifying violent incidents. The approach may still need some manual annotation, the authors point out, in order to operate at its best.

Abdali et al. [19] developed a CNN and Long Short-Term Memory (LSTM) model-based real-time violence detection method. To analyse video frames in real time and pinpoint violent situations, the proposed technique combines the advantages of CNN and LSTM. The approach's primary characteristics include real-time video processing, a bespoke training dataset, and highly accurate violent event detection with low false-positive rates. According to the study, the suggested solution beats current approaches in terms of processing speed and accuracy, making it appropriate for use in practical applications. The approach could involve a lot of computational power, and further study is needed to improve it for various settings and environments.

III. Methodology

This section discusses the details of the procedures in our methodology. This method consists of dataset description, dataset set preparation, Yolo algorithm, and training of the Yolo model. The corresponding details explain in the following sections.

A. Description of the Dataset

The dataset includes 3333 images of resolution 416 x 416 pixels with the annotated objects. The annotations include bounding boxes around people and objects of interest and labels indicating whether the object is associated with violent behavior. The dataset includes examples of different types of violence, including fights, weapons, and attacks. The dataset is intended for use with the YOLO (You Only Look Once) algorithm, a popular object detection algorithm known for its speed and accuracy.

B. Dataset Preparation

This study's provided dataset for violence detection has undergone several pre-processing and augmentation procedures. Pre-processing refers to the process of preparing the data for machine learning tasks. In this dataset, pre-processing included resizing all images to a resolution of 416 x 416 pixels, the input size required by the YOLO algorithm. Additionally, the dataset was converted to the YOLO format, which involves creating text files that contain the bounding box annotations and labels for each image.

Augmentation procedures are utilized to enhance the dataset's diversity and size artificially, improving the model's performance by making it more robust to variations in the input data. The dataset was augmented using various techniques, such as random scaling, random horizontal flipping, random rotation, and random translation. These techniques were applied to each image and corresponding annotations to create new training samples with slightly different characteristics.

Random horizontal flipping involves randomly flipping each image horizontally, which increases the diversity of the dataset and helps prevent overfitting. Random scaling involves randomly scaling each image by a factor of 0.25 to 2.0, which helps the model learn to recognize objects at different scales. Random translation involved randomly shifting each image horizontally and vertically by up to 20% of its width and height, respectively, which helps the model learn to recognize objects in different positions. Random rotation involves randomly rotating each image by up to 10 degrees, which helps the model learn to recognize objects from different angles.

All of these pre-processing and augmentation procedures were performed to enhance the diversity and quality of the dataset, which is able to lead to better performance and generalization of the violence detection model.

C. YOLO Algorithm

YOLO (You Only Look Once) is an object detection algorithm that simultaneously forecasts class probabilities and bounding boxes for objects in an input image. It is a popular algorithm due to its real-time detection capabilities and high accuracy. The YOLO algorithm consists of two main components: a post-processing algorithm and a convolutional neural network (CNN). The CNN takes an input image and outputs a set of bounding boxes along with their class probabilities. The post-processing algorithm selects the most probable bounding boxes and discards the others.

YOLO has several versions, with YOLOv5 being the latest and most advanced version. YOLOv5 is faster and more

accurate than previous versions, thanks to several improvements, including using a backbone network architecture, improved feature extraction, and a more efficient post-processing algorithm. Fig. 1 demonstrates the architecture of YOLOv5 network. The backbone of YOLOv5 is called CSPDarknet, which stands for Cross Stage Partial Network. It is a modified version of the Darknet architecture used in previous versions of YOLO. CSPDarknet is composed of a series of convolutional layers that extract features from the input image. It is designed to be computationally efficient while still producing high-quality feature maps.

The neck of YOLOv5 is called PANet, which stands for Path Aggregation Network. It is a feature fusion module that combines features from different scales and resolutions. The PANet module uses a top-down pathway to aggregate features from high-resolution feature maps and a bottom-up pathway to aggregate features from low-resolution feature maps. The resulting feature maps are then fused to form a single feature map with rich information from multiple scales.

The head of YOLOv5 is called YOLOLayer. It is responsible for predicting class probabilities and bounding boxes for objects in the input image. YOLOLayer uses anchor boxes to predict the location and size of objects in the image. It also uses a softmax function to forecast the probability of each object belonging to a particular class. The final output of the YOLOLayer is a bounding box set with associated class probabilities.

The procedure in the YOLOv5 algorithm is based on the following steps:

1) Input image: YOLO takes an input image of size (width, height, channels) and resizes it to a fixed size (416x416x3) before feeding it to the network.

2) CNN architecture: The CNN architecture of YOLOv5 consists of a backbone network and several detection heads. The backbone network is based on CSPDarknet53, a variant of Darknet53. The detection heads are responsible for predicting bounding boxes and class probabilities.

3) Feature extraction: The feature extraction process is carried out by the backbone network, which generates feature maps of various resolutions. The feature maps are then fed to the detection heads.

4) Bounding box prediction: The detection heads predict a bounding box set for each feature map. The bounding boxes are represented as (x, y, w, h), where (x, y) is the center of the box, w is the width, and h is the height.

5) Class probability prediction: The detection heads also predict class probabilities for each bounding box. The class probabilities represent the probability that the object inside the bounding box belongs to a particular class.

6) Non-maximum suppression: After predicting bounding boxes and class probabilities, YOLOv5 applies non-maximum suppression to eliminate duplicate detections. Non-maximum suppression selects each object's most probable bounding box and discards the others.

In summary, YOLOv5 is a state-of-the-art object detection algorithm that utilizes a CNN to forecast class probabilities and bounding boxes for objects in an image. It has several improvements over previous versions, making it faster and more accurate. The algorithm consists of a backbone network, detection heads, and a post-processing algorithm that applies non-maximum suppression.

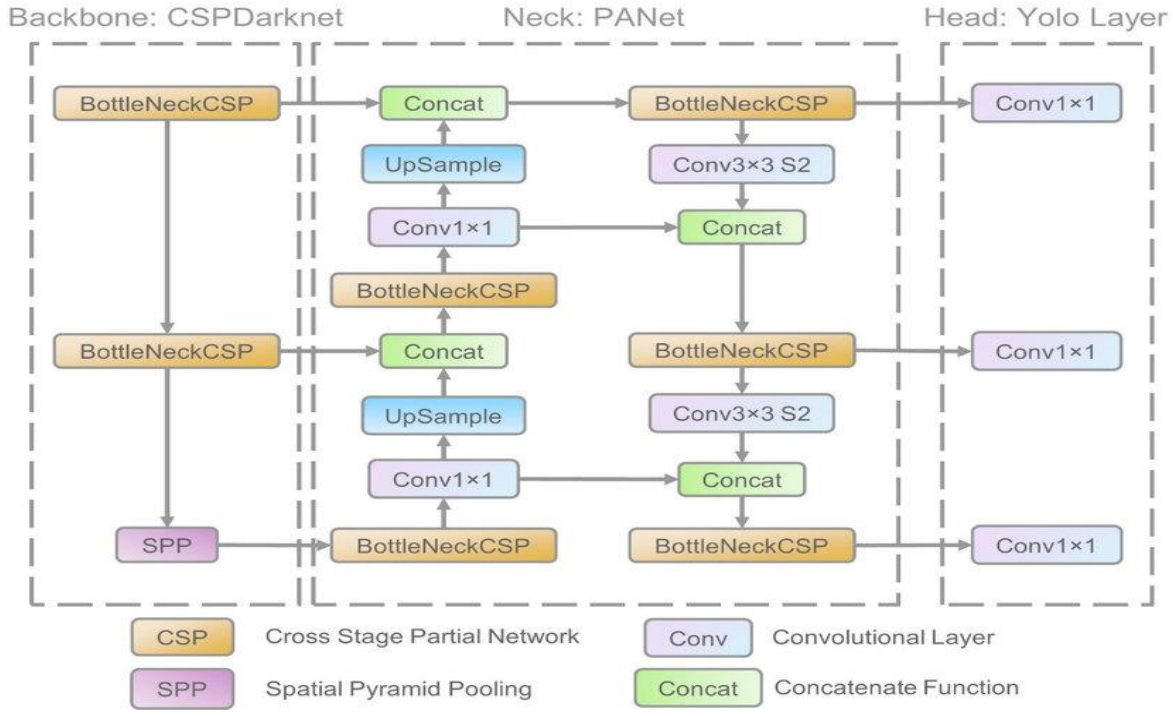


Fig. 1. The architecture of Yolov5 network [20].

D. Training the YOLOv5 Model

Training a YOLO (You Only Look Once) model for violence detection involves preparing the dataset, selecting a pre-trained model, fine-tuning it, and evaluating its performance. The first step discussed in section 3.2 is to prepare the dataset by resizing the images, annotating them with bounding boxes around the violent objects, and saving the annotations in a format the YOLO model can understand. Next, a suitable pre-trained YOLO model for violence detection must be selected for YOLOv5.

In the fine-tuning step, the pre-trained YOLO model is trained on the dataset, and its performance is assessed on the validation set. Hyperparameter tuning is able to be carried out to optimize the performance of the model. The batch size, learning rate, and number of epochs are some of the hyperparameters that can be tuned. Once the model is fine-tuned, its performance is assessed on the test set utilizing metrics such as F1 score, recall, and precision. If the model's performance is unsatisfactory, additional violent images can be added to the dataset, or the hyperparameters can be further tuned.

For training purpose, the dataset consisting of 2300 training samples, 662 testing samples, and 334 validation samples serves as the foundation for training the model YOLO model for violence detection using the given dataset of 2300 training samples, 662 testing samples, and 334 validation samples. Table I shows the proportion of each training, validation and testing sets. In the initial stage, the dataset is preprocessed to ensure consistency and compatibility with the YOLOv5 architecture. This involves resizing all images to a uniform input size, often in the form of squares, to facilitate streamlined processing. Annotations for each image are also processed to provide accurate bounding box coordinates and class labels corresponding to violent actions. These annotations are crucial for training the model to recognize and classify violence instances. The YOLOv5 model is then initialized with pre-trained weights, typically on a large-scale dataset, leveraging knowledge learned from a broad range of objects and features. Fine-tuning is performed on the violence detection dataset, allowing the model to adapt its features and parameters specifically for identifying violent actions. During training, the model iteratively adjusts its parameters by comparing predicted bounding boxes and class probabilities to the ground-truth annotations. This optimization process, often implemented using techniques like stochastic gradient descent, seeks to minimize the disparity between predictions and actual annotations.

To ensure the model generalizes well to new, unseen data, the training process employs techniques such as data augmentation. This involves applying transformations to the images, such as rotations, flips, and color variations, to expose the model to diverse scenarios it may encounter in real-world surveillance situations. Additionally, the training dataset is shuffled to prevent the model from memorizing the order of samples. Throughout training, the model's performance is regularly evaluated using the validation dataset. Metrics like mean average precision (mAP) are calculated to assess the model's ability to precisely localize and classify violent actions. Training continues until the model's performance plateaus or shows satisfactory convergence.

TABLE I. POTATION TESTING, VALIDATION, AND TRAINING SETS

Set name	No. of samples	Set proportion (%)
Training	2300	70%
Validation	662	20%
Testing	334	10%

IV. EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION

The experimental findings and performance assessment of the suggested Yolov5 for violence detection on customized datasets are presented in this part: one research used to fall, no-fall, and half-classless films in a bespoke dataset. Utilizing various input image sizes, training datasets, and object detection thresholds, the study assessed Yolov5's performance. Fig. 2 demonstrates some examples of experimental results.

The experimental results and model performance evaluation for the YOLOv5 model for violence detection can be measured by various metrics such as Mean Average Precision (mAP), recall, and precision. Precision is the ratio of true positive detections to all of the model's positive detections. The formula for precision is:

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

The ratio of precise positive detections to all positive cases in the dataset determines recall. The recall is the proportion of real positives to all real positives in the dataset. It displays the capacity of model to detect positive samples reliably. The formula for the recall is,

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

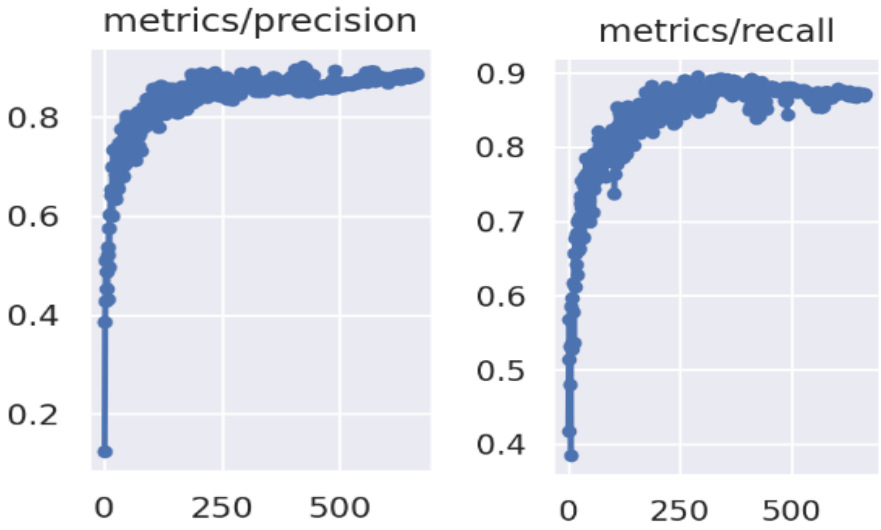
In precision and recall, where TP is the number of true positives (correctly detected violence), FP is the number of false positives (non-violences detected as anomalies), and FN is the number of false negatives (violence not detected by the algorithm). Based on obtained results, Table 2 presents performance measurements for the average precision rate for each class.

A popular performance metric, the mAP, accounts for memory and accuracy at different confidence levels. It returns a single scalar number summarising the model's overall detection performance as the average accuracy over a range of recall values. Fig. 3 illustrates performance metrics for generated Yolov5-based violence detection model.

Fig. 2. Experimental results.

TABLE II. AVERAGE PRECISION RATES BY CLASSES

Set name	Validation set	Testing set
Violence	93%	91%
Non-Violence	89%	90%
All	91%	91%



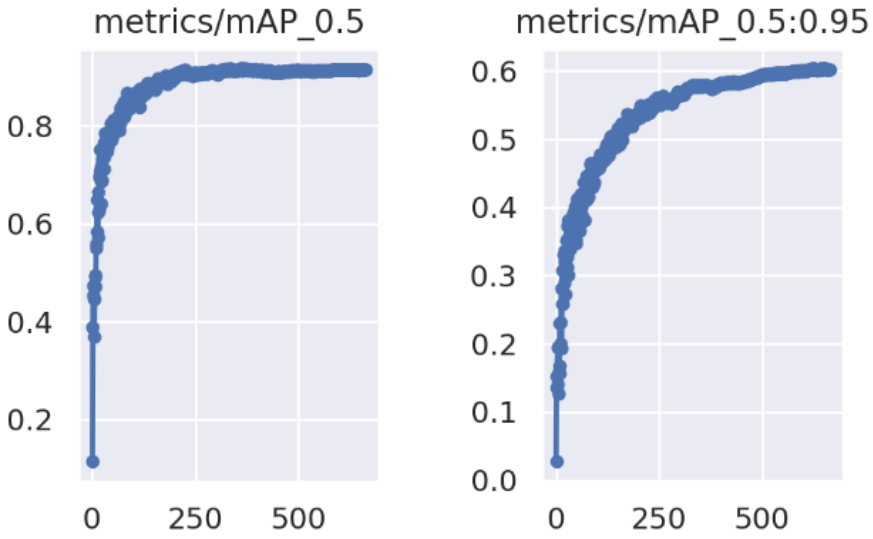


Fig. 3. Illustration of performance metrics for generated Yolov5-based violence detection model.

V. Conclusion and Future Works

A promising area of research that can improve public safety is using deep learning-based approaches for violence detection in IoT-based surveillance systems. This study presented the YOLO algorithm to address the issues related to violence detection. We utilize YOLOv5 as an advanced object identification technique that can quickly and accurately detect multiple objects in an image. We create a model for violence detection using images of violent and non-violent scenes divided into testing, validation, and training sets. The model's performance is evaluated using standard performance indicators. Our system can learn to recognize patterns of violence and accurately differentiate between violent and non-violent classes in real time. Therefore, the proposed method in this study is based on an efficient and fast deep learning architecture named as Yolov5 network. This network, as previous studies proofed and indicated, it is very effective in real-time detection algorithms. Inspiring of this, we also adopted Yolov5 algorithm and generated a model to deal with violence detection. As our experimental results indicated, the proposed method present accurate results and provide satisfy in real-time requirement. Future work could focus on improving the robustness of the model by addressing various environmental factors that may affect violence detection accuracy. Additionally, the development of larger datasets with diverse scenarios can improve the generalizability of the model. Finally, further investigation could explore the integration of multiple sensors and modalities, such as audio and motion sensors, to enhance the accuracy and reliability of violence detection systems.

References

[1] F. U. M. Ullah, M. S. Obaidat, A. Ullah, K. Muhammad, M. Hijji, and S. W. Baik, "A comprehensive review on vision-based violence detection in surveillance videos," *ACM Comput Surv*, vol. 55, no. 10, pp. 1–44, 2023.

[2] B. Omarov, S. Narynov, Z. Zhumanov, A. Gumar, and M. Khassanova, "State-of-the-art violence detection techniques in video surveillance security systems: a systematic review," *PeerJ Comput Sci*, vol. 8, p. e920, 2022.

[3] F. U. M. Ullah *et al.*, "AI-assisted edge vision for violence detection in IoT-based industrial surveillance networks," *IEEE Trans Industr Inform*, vol. 18, no. 8, pp. 5359–5370, 2021.

[4] M. Islam, A. S. Dukyil, S. Alyahya, and S. Habib, "An IoT Enable Anomaly Detection System for Smart City Surveillance," *Sensors*, vol. 23, no. 4, p. 2358, 2023.

[5] W. Ullah *et al.*, "Artificial Intelligence of Things-assisted two-stream neural network for anomaly detection in surveillance Big Video Data," *Future Generation Computer Systems*, vol. 129, pp. 286–297, 2022.

[6] L. M. Dang, K. Min, H. Wang, M. J. Piran, C. H. Lee, and H. Moon, "Sensor-based and vision-based human activity recognition: A comprehensive survey," *Pattern Recognit*, vol. 108, p. 107561, 2020.

[7] N. Mumtaz *et al.*, "An overview of violence detection techniques: current challenges and future directions," *Artif Intell Rev*, pp. 1–26, 2022.

[8] M. H. Rohit, "An IoT based System for Public Transport Surveillance using real-time Data Analysis and Computer Vision," in *2020 Third International Conference on Advances in Electronics, Computers and Communications (ICAECC)*, IEEE, 2020, pp. 1–6.

[9] M. Ramzan *et al.*, "A review on state-of-the-art violence detection techniques," *IEEE Access*, vol. 7, pp. 107560–107575, 2019.

[10] A. Singh, T. Anand, S. Sharma, and P. Singh, "IoT based weapons detection system for surveillance and security using YOLOV4," in *2021 6th International Conference on Communication and Electronics Systems (ICCES)*, IEEE, 2021, pp. 488–493.

[11] P. Zhou, Q. Ding, H. Luo, and X. Hou, "Violence detection in surveillance video using low-level features," *PLoS One*, vol. 13, no. 10, p. e0203668, 2018.

[12] Y. Gao, H. Liu, X. Sun, C. Wang, and Y. Liu, "Violence detection using oriented violent flows," *Image Vis Comput*, vol. 48, pp. 37–41, 2016.

[13] M. M. Soliman, M. H. Kamal, M. A. E.-M. Nashed, Y. M. Mostafa, B. S. Chawky, and D. Khattab, "Violence recognition from videos using deep learning techniques," in *2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, IEEE, 2019, pp. 80–85.

[14] P. Wang, P. Wang, and E. Fan, "Violence detection and face recognition based on deep learning," *Pattern Recognit Lett*, vol. 142, pp. 20–24, 2021.

[15] G. Sreenu and S. Durai, "Intelligent video surveillance: a review through deep learning techniques for crowd analysis," *J Big Data*, vol. 6, no. 1, pp. 1–27, 2019.

[16] S. U. Khan, I. U. Haq, S. Rho, S. W. Baik, and M. Y. Lee, "Cover the violence: A novel Deep-Learning-Based approach towards violence-detection in movies," *Applied Sciences*, vol. 9, no. 22, p. 4963, 2019.

[17] N. AlDahoul, H. A. Karim, R. Datta, S. Gupta, K. Agrawal, and A. Albunni, "Convolutional Neural Network-Long Short Term Memory based IOT Node for Violence Detection," in *2021 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, IEEE, 2021, pp. 1–6.

[18] D. Choqueluque-Roman and G. Camara-Chavez, "Weakly supervised violence detection in surveillance video," *Sensors*, vol. 22, no. 12, p. 4502, 2022.

[19] A.-M. R. Abdali and R. F. Al-Tuma, "Robust real-time violence detection in video using cnn and lstm," in *2019 2nd Scientific Conference of Computer Sciences (SCCS)*, IEEE, 2019, pp. 104–108.

[20] R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, "A forest fire detection system based on ensemble learning," *Forests*, vol. 12, no. 2, p. 217, 2021.