

# Video-based physical violence detection model for efficient public space surveillance

**Erick, Benfano Soewito**

Department of Computer Science, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University,  
Jakarta, Indonesia

## Article Info

### Article history:

Received Nov 12, 2024

Revised Jun 23, 2025

Accepted Nov 5, 2025

### Keywords:

Change detection

Conv3D

ConvLSTM2D

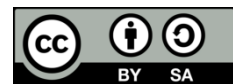
Efficient violence detection

Motion blur

## ABSTRACT

This study aims to develop an effective real-time model for detecting violence in public spaces, focusing on achieving a balance between accuracy and computational efficiency. We evaluate various model architectures, with the main comparison between the ConvLSTM2D and Conv3D models commonly used in video analysis to capture spatial and temporal features. The ConvLSTM2D model, combined with preprocessing layers such as change detection and motion blur, showed optimal performance, achieving 86% accuracy after Bayesian optimization. With a low parameter count of 25,137, this model enables fast inference in just 0.010 seconds, making it suitable for real-time applications that require efficient computation. In contrast, the Conv3D model, which is also combined with preprocessing layers such as change detection and motion blur and has more than nine million parameters, shows a lower accuracy of 77.5% as well as a slower inference time of 0.025 seconds, making it unsuitable for real-time applications. The results of this study show that the ConvLSTM2D model is promising for real-time violence detection systems in public spaces, where a fast and accurate response is essential to prevent further acts of violence.

*This is an open access article under the [CC BY-SA](#) license.*



## Corresponding Author:

Erick

Computer Science Department, BINUS Graduate Program - Master of Computer Science

Bina Nusantara University

Jakarta, Indonesia

Email: erick007@binus.ac.id

## 1. INTRODUCTION

Physical violence is a serious problem for society. Victims can suffer a range of physical injuries, from minor bruises to life-threatening ones, such as broken bones that can affect their quality of life and, in some cases, lead to death. However, its not only the physical impact, the psychological impact such as anxiety, depression, and post-traumatic stress disorder (PTSD), which can deeply affect their mental health and overall well-being. Additionally, their traumatic experiences often struggles with relationship and social connections. Many victims may also turn to substance use as a coping mechanism for their trauma [1]. Physical violence can occur in a variety of locations, including domestic settings, workplaces, educational institutions, public spaces, and public transportation. It can take the form of physical assault or the use of weapons. A study indicates that public transportation systems, such as buses and trains, are prone to violent incidents, raising concerns about safety [2]. Furthermore, another study underscores the prevalence of violence against women (VAW) in urban spaces, including parks, streets, and public transportation [3]. This highlights the necessity for enhanced safety measures in these areas. Consequently, there has been an increasing emphasis on the importance of automatic violence detection systems in these environments to more effectively address the frequent risks of violence or criminal activity.

The implementation of closed-circuit television (CCTV) surveillance systems is a common practice for monitoring and preventing violent activities in public spaces. Research suggests that urban video surveillance can be an effective tool to improve public safety by assessing the impact of surveillance on particularly serious crimes, with positive effects shown in reducing overall crime rates, auto theft, violent crimes, and property crimes in various cities [4]. Detecting physical violence through CCTV aims to quickly identify violent incidents, enabling security personnel to respond more effectively. Violence detection via surveillance cameras is applicable in diverse settings, including public facilities, streets, parks, schools, and workplaces, and can contribute to reducing cases of physical violence.

Furthermore, as illustrated in Figure 1, violent acts often occur within a very short duration compared to the total footage captured by surveillance cameras. In the image sequence, the marked red box highlights that violent activity is detected only in brief frames. Continuous manual monitoring of surveillance footage is time-consuming and prone to human error, increasing the risk of missed incidents. This approach becomes inefficient, particularly in situations that demand rapid decision-making to prevent violent acts. Therefore, automatic detection of physical violence in videos is essential to improve response times and reduce reliance on constant human oversight.

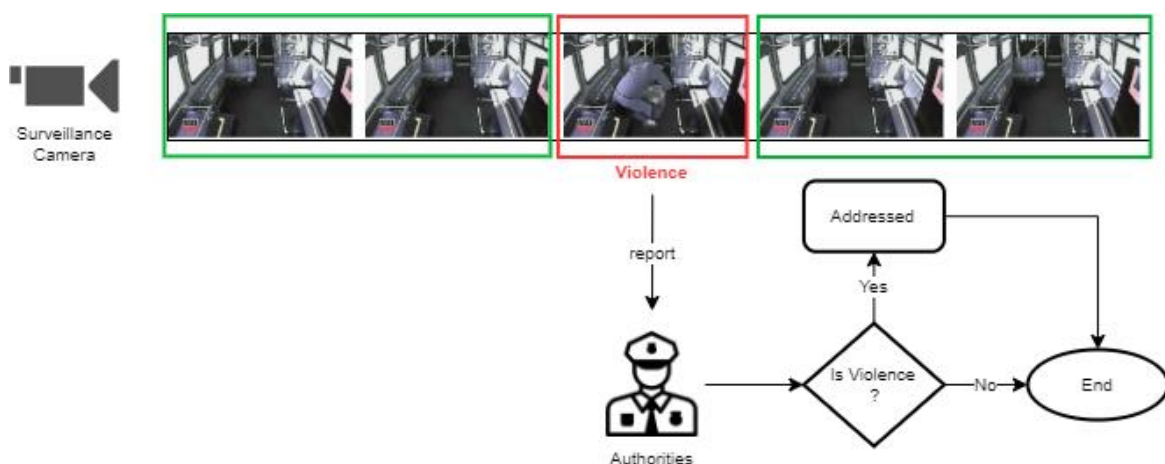


Figure 1. Illustration of the process of reporting violent activities recorded on surveillance cameras

The detection of physical violence in video has become a popular research area over the past few decades, particularly in the field of security and surveillance. The use of video data for surveillance applications to detect physical violence has existed since 2002. In that year, Datta *et al.* [5] introduced the first approach to violence detection, marking a significant milestone in this field. This initiated a new era in which researchers began to explore the use of video data to detect violent behavior. Between 2002 and 2013, the main techniques used in violence detection were based on manually crafted features. These techniques relied entirely on experts' understanding of physical violence to determine low-level features from videos. For instance, features such as interactions between individuals and movement patterns were considered important for detecting violence. Although these techniques were useful at the time, they had limitations, and their accuracy was not high enough to provide reliable results [6]. However, with advances in deep learning models, the first approach using 3D convolutional neural networks (CNNs) was introduced by [7], [8]. This approach worked independently without prior knowledge and demonstrated better results. Additionally, there have been efforts to optimize model efficiency for violence detection in video using lightweight 2D CNN architectures combined with long short-term memory (LSTM) [9]. Consequently, many researchers began introducing deep learning-based techniques, which showed far better results than the previously used manually crafted feature methods [10].

There are numerous studies on violence classification exploring various deep learning models and datasets that have made notable strides in both accuracy and efficiency. In [11], a comparative analysis of three 2D CNN models, namely ResNet50 [12], InceptionV3 [13], and VGG19 [14], each combined with LSTM to classify violent activities across three popular datasets: The datasets included violent flows [15], hockey fights, and movie fights [16]. The results indicated that ResNet50 performed best, achieving an average accuracy of 90%, followed by InceptionV3 with 89%, while VGG19 lagged behind with 79%.

Further improving the robustness of violence detection, Sernani *et al.* [17] introduced a new dataset, AIRTLab, which was specifically designed to address the issue of false positives commonly found in existing datasets. In traditional datasets, actions such as hugging, clapping, or touching were frequently misclassified as violent. Additionally, they also proposed three deep learning models for violence classification. The initial model integrated C3D (3D ConvNet) [18], a CNN optimized for processing spatio-temporal data, with a support vector machine (SVM) classifier. The second model replaced the SVM with a fully connected layer, while the third used a convolutional long short-term memory (ConvLSTM) architecture. Evaluations using the AIRTLab dataset, along with the Hockey Fight and Crowd Violence datasets, demonstrated stable performance across all models, with the C3D and SVM models achieving the highest accuracy. This study underscores the importance of addressing dataset limitations and demonstrates how deep learning models can effectively mitigate false positives in violence detection tasks.

While notable advancements have been achieved in the field of video-based violence detection, [19] underscored that the majority of extant models prioritize accuracy and performance, with a consequent disregard for computational efficiency. In the context of real-time monitoring systems, the use of lightweight models is of paramount importance. In response, this research proposed a model that strikes a proper balance between computational cost and accuracy. The model integrates a motion saliency map (MSM) to highlight moving objects, a 2D CNN with frame-grouping to extract temporal features efficiently, and a temporal squeeze-and-excitation (T-SE) module to focus on actions occurring at specific moments in the video. The model was evaluated on multiple datasets, including Hockey Fight and Movie Fight, achieving high accuracy while maintaining low computational demands, making it well-suited for real-time applications.

Similarly, [20] addressed the need for efficient models by proposing a three-part architectural configuration that combines MobileNetV2 for spatial feature extraction, LSTM for processing temporal information, and fully connected layers for classification. MobileNetV2 was selected for its minimal computational expense and relatively small parameter size, while maintaining robust accuracy. The model exhibited robust performance across a range of datasets, including RWF-2000, Hockey Fights, and Movie Fights, achieving high accuracy with minimal computational overhead. This work underscores the potential of combining efficient architectures like MobileNetV2 with LSTM to achieve high performance while maintaining low resource utilization.

From these studies, there are two main approaches used for violence classification in videos which are 3D CNN and 2D CNN combined with LSTM. However, many studies focus only on model accuracy without considering computational efficiency which is important for real-world applications where fast decision-making is required to prevent violence. Models with high computational load are not practical for continuous and real-time use in public areas. Therefore, it is crucial to prioritize low computational load, compact model size, and optimal performance when designing models for real-world violence detection.

This research proposed an architecture with the ConvLSTM2D model, which offers a lighter computational load than common 3D CNNs [21]. Moreover, to improve classification accuracy while maintaining efficiency, this approach incorporates preprocessing techniques such as change detection and motion blur, which can effectively capture important information about the movement of objects which is an important feature for identifying violent acts. By emphasizing computational efficiency in addition to accuracy, this method aims to bridge the gap between high-performance violence detection models and the practical demands of real-time surveillance applications.

## 2. RESEARCH METHOD

### 2.1. Dataset

The dataset used for this study is sourced from a public dataset commonly used for violence classification tasks, namely RWF-2000 [22]. This research use RWF-2000 because it is designed to provide a more realistic representation of violent events compared to existing datasets, making it more suitable for detecting violence in real-world scenarios. This realism is achieved by including raw videos collected from surveillance cameras available on YouTube, which better reflect the types of footage encountered in practical applications of violence detection systems. The video files in this dataset are in audio video interleave (AVI) video format. Each video in this dataset is segmented into 5-second clips with a frame rate of 30 frames per second (fps), and each clip is labeled as either violent or non-violent. The dataset is evenly split into 1000 videos for training and 1000 videos for validation, resulting in a total of 2000 videos. To optimize the processing time and computational resources required for model training, the original frame rate of 30 fps was reduced to 10 fps. Additionally, the frame size of each video was resized to 100x100 pixels. Since all the videos were collected from surveillance cameras, some videos have poor image quality due to environmental factors, lighting, and the resolution of the surveillance cameras. As shown in Figure 2, which is a frame from a video labeled as violent, the region of interest is only located in a small part of the frame, while the dark lighting makes it difficult to see the violent scenes.



Figure 2. A violent scene frame with poor lighting, where the violent scene is circled in red

## 2.2. Proposed model architecture

As shown in Figure 3, there are three parts to the proposed model architecture for violence classification. The first part contains histogram equalization, motion blur and frame difference which are used to preprocess and extract the frame difference information in the video. The next part is a spatio-temporal encoder that serves to process spatial and temporal features consisting of three block ConvLSTM2D layers. Finally, the features will be processed to a series of fully connected layers that act as classifiers.

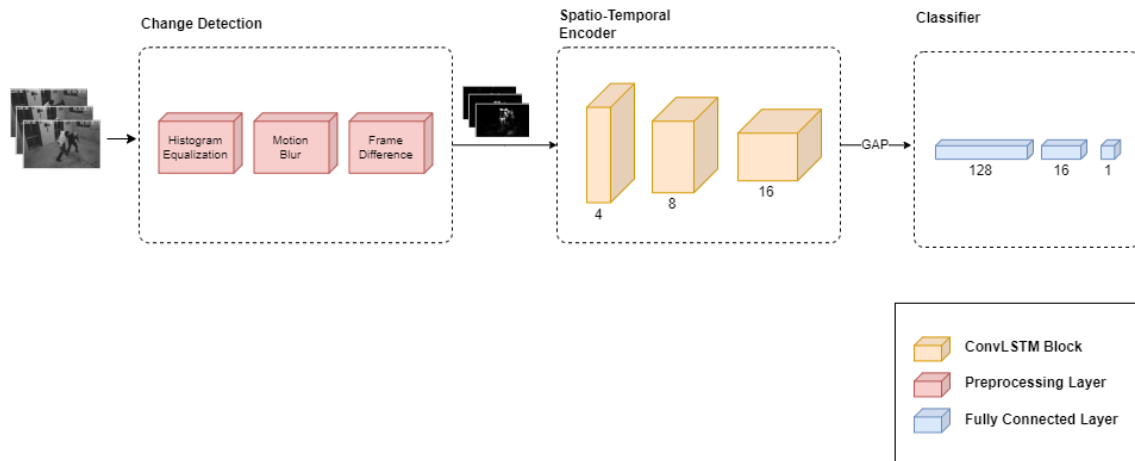


Figure 3. Proposed model architecture for violence classification

### 2.2.1. Change detection with motion blur

In addition to obtaining the change information from the video, the first part of the architecture functions to capture frame-by-frame changes in the video. By comparing frames, this algorithm can identify important changes in movement or scenes. This helps distinguish between normal activity and unusual events, contributing to a more accurate video classification system by highlighting the main events or actions in the video. As shown in Figure 4, the white lines in the output image indicate significant changes in the frames.

The technique used to detect changes between frames in this video is frame difference, which has shown the best performance for detecting changes in the video sequences [23]. This technique involves calculating the absolute or relative difference between the pixel values of corresponding pixels in two adjacent frames. By subtracting the pixel values of one frame from the other, areas with significant changes, such as movement or appearance changes, will become clearly visible. The formula for calculating the frame difference can be seen in (1).

$$frame\_diff_n = frame_{n+1} - frame_n \quad (1)$$

To improve the accuracy of change detection in low-quality videos, histogram equalization is applied as a preprocessing step. This method enhances the contrast of each frame by redistributing intensity values, making the differences between the foreground and background more distinct [24]. As a result, subtle

changes, such as variations in lighting or slight movements, become more detectable during frame comparisons. By normalizing intensity levels, histogram equalization mitigates the effects of varying illumination conditions between frames, which can otherwise reduce the accuracy of change detection. This preprocessing step makes subsequent techniques, such as frame differencing, more effective in capturing changes between frames. The effectiveness of histogram equalization in highlighting contrasts can be observed in Figure 5, which shows the improved visibility of changes after applying this technique.

Following histogram equalization, a horizontal motion blur is also applied to further enhance change detection accuracy. This blur smooths out minor variations between frames and highlights more significant changes, helping to reduce noise and small fluctuations. By averaging pixel values across a defined range in a horizontal direction, the motion blur creates a smearing effect that mimics the movement of objects. This emphasizes the direction of movement, making it easier to identify areas of significant change between frames. The blur focuses the frame difference calculation on meaningful movements, allowing regions with substantial changes to stand out more clearly, especially in videos with subtle motions or lower frame rates [25]. As illustrated in Figure 6, the horizontal blur smooths out minor variations between frames, creating an effect that highlights object movements.



Figure 4. Frame difference information extracted by using change detection technique



Figure 5. Frame difference information extracted after using histogram equalization

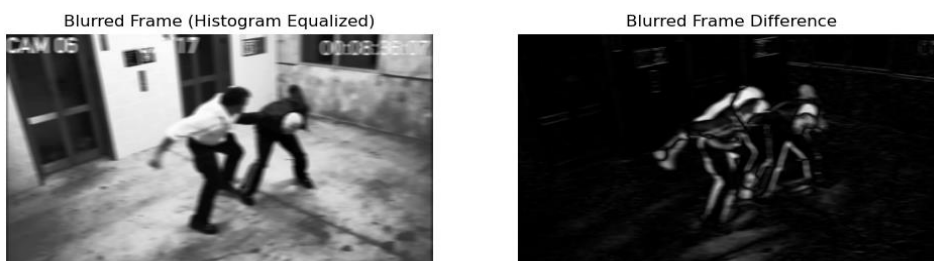


Figure 6. Frame difference information after applying motion blur

### 2.2.2. Spatio-temporal encoder

Once the change information features have been extracted, the following step is to merge the features from each frame into a video-level representation. The spatio-temporal encoder architecture contains three ConvLSTM2D blocks, which are designed to capture both the spatial and temporal aspects of the video data. As detailed in Table 1, each ConvLSTM2D block begins with a ConvLSTM2D layer, which integrates

convolutional operations with LSTM functionality. This allows the model to effectively capture both spatial and temporal information. Following the ConvLSTM2D layer, a batch normalization layer is applied that helps the model train more efficiently and ensures that it generalizes well to new data. Next, a MaxPooling3D layer is introduced to reduce the spatial dimensions while retaining important features allowing the architecture to focus on the most relevant spatial information from the video frames. Additionally, each block incorporates a dropout layer as a regularization technique to prevent overfitting. This structure enables the spatio-temporal encoder to transform frame-level information into a comprehensive video-level representation, making it highly effective for tasks that require an understanding of both spatial details and temporal dynamics, such as violence detection in videos.

Table 1. ConvLSTM2D block layers

No.	Layer name
1	ConvLSTM2D
2	BatchNormalization
3	MaxPooling3D
4	Dropout

### 2.2.3. Classifier

After the spatio-temporal encoder architecture extracts spatial and temporal features from the video, these features pass through a global average pooling (GAP) layer. The GAP layer condenses the spatial information by averaging each feature map, reducing dimensionality while retaining essential information. The condensed features are then fed into the classifier architecture, which begins with a dense layer containing 128 units and a rectified linear unit (ReLU) activation function. Following this, a second dense layer with 16 units further processes the features. Finally, the output layer, a dense layer with a single unit and a sigmoid activation function, produces a probability score. This score indicates the likelihood that the input video contains instances of violence, making it suitable for the binary classification task of distinguishing between violent and non-violent content.

### 2.3. Model training

During the model training phase, several combinations of preprocessing layers were explored such as frame difference, frame difference combined with motion blur, and using Conv3D layer instead of ConvLSTM2D. These combinations were tested to determine how effectively each could extract meaningful features from the input data. After identifying the combination that achieved the highest accuracy, the model was further refined using Bayesian optimization. The fine-tuning process use Bayesian optimization to determine the optimal parameters for the motion blur kernel size and learning rate. As shown in Table 2, the dropout rates for each layer in the ConvLSTM2D block are searched within a range of 0.0 to 0.5. The motion blur kernel size is evaluated with options of 3, 5, and 7. Additionally, the learning rate for the Adam optimizer is optimized within a range of 0.01 to 0.00001 to ensure the best model performance [26], [27].

Table 2. Hyper parameter searched for fine tuning the model using bayesian optimization

Parameter	Options
1st Dropout layer rate	Min value: 0.0; Max value: 0.5
2nd Dropout layer rate	Min value: 0.0; Max value: 0.5
3rd Dropout layer rate	Min Value: 0.0; Max value: 0.5
Motion blur Kernel	3, 5, 7
Learning rate	0.001 to 0.00001

After selecting the best parameters, the model was trained using a learning rate scheduler that automatically reduced the learning rate by a factor of 0.1 when performance improvements plateaued. The training process used a batch size of 10. The training was conducted over 50 epochs to ensure sufficient learning. Additionally, early stopping was employed to prevent overfitting, halting training when the validation accuracy did not improve for five epochs. This approach ensured the model achieved a balance between accuracy and generalization. This training was performed on the Google Colab platform using a Pro Plus subscription, which provided enhanced computational resources. An NVIDIA L4 GPU with 22.5 GB of VRAM was used, offering significant processing power for complex calculations. The system also featured 53 GB of RAM, which facilitated the handling of large datasets and memory-intensive operations.

## 2.4. Model evaluation

To evaluate model performance, this study uses accuracy which is commonly used for assessing classification tasks. Accuracy is the most popular metric for evaluating deep learning models for video classification. In this case, it is the ratio of the number of correct violent or non-violent predictions to the total number of predictions. After that, the next step is to assess its efficiency in real-world scenarios, which is a primary objective of this study. This is achieved through a model efficiency analysis, focusing on the time required for the model to classify videos, known as inference time. To conduct this evaluation, 100 randomly selected violent and non-violent videos from the validation set of the RWF-2000 dataset are processed. The average inference time is then calculated for each video, providing insights into the model's performance in detecting violence under real-world conditions. This efficiency analysis is conducted on the same hardware used for model training.

## 3. RESULTS AND DISCUSSION

### 3.1. Evaluation results on different model architecture

This study evaluates the influence of different preprocessing techniques on model performance for classifying violence. The evaluation includes training the model under three conditions: (1) without a preprocessing layer, (2) with a frame difference preprocessing layer, and (3) with a combination of frame difference and motion blur layers. Furthermore, this study compares the performance of Conv3D layers to ConvLSTM2D layers to assess their effectiveness.

As shown in Table 3, the ConvLSTM2D model performs the poorest, with an accuracy of 77.25%. However, when preprocessing layers such as frame difference added and motion blur are employed, there is a notable improvement in accuracy. The addition of a frame difference layer resulted in a notable increase in accuracy, from 77.25% to 81.5%. Furthermore, the incorporation of both frame difference and motion blur layers led to an additional enhancement in accuracy, reaching 84.5%. These findings highlight the significance of preprocessing layers in extracting crucial features for identifying violent activity in video data.

In terms of model complexity, the ConvLSTM2D model is far more efficient, with just 25,137 parameters, compared to the Conv3D model's more than nine million parameters. Despite having much higher parameter count, the Conv3D model delivers a lower accuracy of only 77.5%. In contrast, the ConvLSTM2D model achieves a superior accuracy of 84.5%. These results demonstrate that the ConvLSTM2D model can be both more efficient and more effective than the Conv3D model for this task, even with a significantly smaller parameter count.

Table 3. Accuracy and total parameters on different preprocessing layer in ConvLSTM2D and Conv3D model

ConvLSTM2D	Conv3D	Frame difference	Motion blur	Accuracy	Total parameters
✓				77.25%	25,137
✓		✓		81.25%	25,137
✓		✓	✓	<b>84.5%</b>	<b>25,137</b>
	✓	✓	✓	77.5%	9,019,329

### 3.2. Evaluation results on the best model after fine-tuning

The ConvLSTM2D model, which incorporates preprocessing layers for frame difference and motion blur, demonstrated the highest accuracy following fine-tuning through Bayesian optimization with the best optimized hyperparameters, as detailed in Table 4. Figure 7 illustrates the notable enhancement in the model's accuracy resulting from this fine-tuning process. Prior to fine-tuning, the ConvLSTM2D model exhibited an initial accuracy of 84.5%. However, upon employing the refined hyperparameters obtained through Bayesian optimization, the accuracy increased to 86%. This outcome demonstrates that the integration of preprocessing techniques, specifically Frame Difference and Motion Blur, in conjunction with fine-tuning through Bayesian optimization, resulted in a notable enhancement in the performance of the ConvLSTM2D model.

Table 4. Best hyper parameters obtained from fine tuning using bayesian optimization

Hyper parameter	Best value
1 <sup>st</sup> Dropout Layer Rate	0.2
2 <sup>nd</sup> Dropout Layer Rate	0.3
3 <sup>rd</sup> Dropout Layer Rate	0
Motion Blur Kernel	5
Learning Rate	0.0002

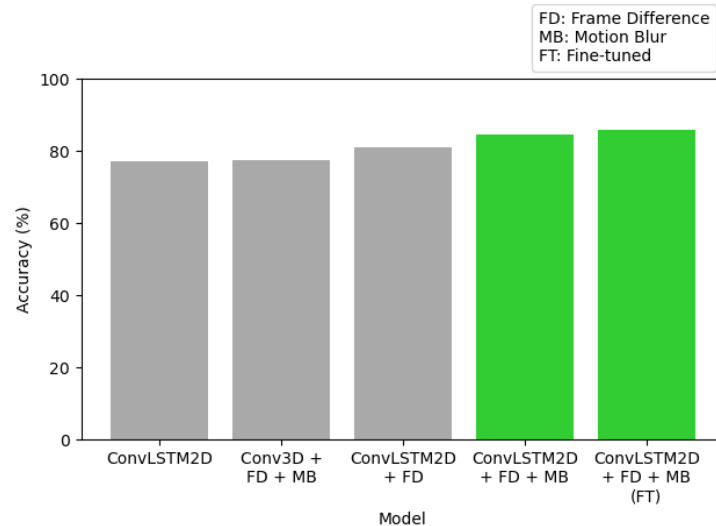


Figure 7. Accuracy of different model architectures

### 3.3. Efficiency evaluation on different models

As shown in Table 5, the models using ConvLSTM2D architecture have a consistent average inference time of 0.010 seconds for video prediction, even when additional preprocessing layers, such as frame difference and motion blur, are incorporated to enhance feature extraction. This is due to the fact that these ConvLSTM2D models have the same parameter size, which results in a stable computational complexity across the various models. In contrast, the model using the Conv3D architecture shows a longer average inference time of 0.025 seconds for video prediction. This is due to the significantly higher number of parameters in the Conv3D model, totaling 9,019,329 parameters. The increased number of parameters in the Conv3D model leads to higher computational demands during the prediction process, as more parameters need to be processed.

Table 5. Average inference time of different model architectures

Model name	Average inference time (s)	Total parameters
ConvLSTM2D	0.010	25,137
ConvLSTM2D + FD	0.010	25,137
ConvLSTM2D + FD + MB	0.010	25,137
ConvLSTM2D + FD + MB (FT)	0.010	25,137
Conv3D + FD + MB	0.025	9,019,329

### 3.4. Evaluation results summary

Table 6 shows the optimal balance between accuracy and efficiency when classifying violent activity in video data. The ConvLSTM2D model, particularly the version incorporating frame difference and motion blur preprocessing layers, demonstrated the highest level of performance, reaching an accuracy of 86% after fine-tuning through Bayesian optimization. The model also exhibited a shorter inferencing time of 0.010 seconds, due to its relatively low parameter count of 25,137. This makes it both accurate and computationally efficient. In comparison, the Conv3D model, despite having over nine million parameters, demonstrated a lower accuracy of 77.5% and required twice the inference time at 0.025 seconds for video predictions. These results indicate that ConvLSTM2D is a more suitable choice for real-time violence detection, offering an optimal balance between speed and accuracy.

Table 6. Summary of performance across difference model architectures and efficiency evaluation

Model name	Accuracy	Average inference time (s)	Total parameters
ConvLSTM2D	77.25%	0.010	25,137
ConvLSTM2D + FD	81.25%	0.010	25,137
ConvLSTM2D + FD + MB	84.5%	0.010	25,137
ConvLSTM2D + FD + MB (FT)	86%	0.010	25,137
Conv3D + FD + MB	77.5%	0.025	9,019,329



#### 4. CONCLUSION

This study highlights the significance of public violence detection systems. Various model architectures were tested to effectively recognize violence. The results show that the ConvLSTM2D-based model is more efficient in detecting violence in videos due to its lower complexity. Preprocessing steps, such as frame differentiation and motion blur, help enhance the model's accuracy in detecting violence. This suggests that the model achieves a good balance between speed and accuracy, making it suitable for detecting public violence. For future research, we recommend exploring the use of pre-trained 2D or 3D CNN models to improve the ability to extract important features, as well as investigating more advanced data extraction methods to identify characteristics associated with physical violence.

#### ACKNOWLEDGEMENTS

First of all, the authors would like to express their deepest gratitude to the creators and contributors of the RWF-2000 dataset, who have put a lot of effort in collecting and providing the data that is crucial to our research. The authors are also very grateful to Professor Benfano Soewito for his guidance and encouragement as well as his helpful contributions during the writing of this research. Finally, the authors would like to thank their fellow students from the Computer Science Department at Binus University for their valuable support and assistance during this experiment.

#### FUNDING INFORMATION

This research was supported by Bina Nusantara University. The authors gratefully acknowledge the financial support provided by the university, which made this research possible.

#### AUTHOR CONTRIBUTIONS STATEMENT

The individual contributions of each author, defined by the CRediT (Contributor Roles Taxonomy), are provided in the table below. All authors have read and approved the final manuscript.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Erick	✓	✓	✓		✓	✓		✓	✓		✓		✓	
Benfano Soewito	✓	✓		✓						✓		✓		

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

#### CONFLICT OF INTEREST STATEMENT

The authors state no conflict of interest.

#### DATA AVAILABILITY

To enhance the transparency and reproducibility of this research, the authors have made the source code publicly available. The experiments were conducted using a publicly available dataset, which can be accessed at <https://doi.org/10.5281/zenodo.15687512>. The source code is available at <https://doi.org/10.5281/zenodo.15687104>.




#### REFERENCES

- [1] F. Rivara *et al.*, "The effects of violence on health," *Health Affairs*, vol. 38, no. 10, pp. 1622–1629, Oct. 2019, doi: 10.1377/hlthaff.2019.00480.
- [2] L. Ciampi *et al.*, "Bus violence: an open benchmark for video violence detection on public transport," *Sensors*, vol. 22, no. 21, p. 8345, Oct. 2022, doi: 10.3390/s22218345.
- [3] M. Garfias Royo, P. Parikh, and J. Belur, "Using heat maps to identify areas prone to violence against women in the public sphere," *Crime Science*, vol. 9, no. 1, Aug. 2020, doi: 10.1186/s40163-020-00125-6.
- [4] R. Socha and B. Kogut, "Urban video surveillance as a tool to improve security in public spaces," *Sustainability*, vol. 12, no. 15, p. 6210, Aug. 2020, doi: 10.3390/su12156210.
- [5] A. Datta, M. Shah, and N. Da Vitoria Lobo, "Person-on-person violence detection in video data," in *Object recognition supported by user interaction for service robots*, vol. 1, pp. 433–438, doi: 10.1109/icpr.2002.1044748.




- [6] 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 Computer Science*, vol. 8, p. e920, Apr. 2022, doi: 10.7717/peerj-cs.920.
- [7] C. Ding, S. Fan, M. Zhu, W. Feng, and B. Jia, "Violence detection in video by using 3D convolutional neural networks," in *Advances in Visual Computing*, Springer International Publishing, 2014, pp. 551–558.
- [8] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, Jan. 2013, doi: 10.1109/tpami.2012.59.
- [9] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. C. Woo, "Convolutional LSTM network: a machine learning approach for precipitation nowcasting," *Advances in neural information processing systems*, vol. 28, pp. 802–810, 2015.
- [10] N. Mumtaz *et al.*, "An overview of violence detection techniques: current challenges and future directions," *Artificial Intelligence Review*, vol. 56, no. 5, pp. 4641–4666, Oct. 2022, doi: 10.1007/s10462-022-10285-3.
- [11] M. Patel, "Real-time violence detection using CNN-LSTM," *arXiv preprint arXiv:2107.07578*, 2021.
- [12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2016, vol. 2016-December, pp. 770–778, doi: 10.1109/CVPR.2016.90.
- [13] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2016, vol. 2016-December, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.
- [14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [15] T. Hassner, Y. Itcher, and O. Kliper-Gross, "Violent flows: Real-time detection of violent crowd behavior," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2012, pp. 1–6, doi: 10.1109/cvprw.2012.6239348.
- [16] E. Bermejo Nievas, O. Deniz Suarez, G. Bueno García, and R. Sukthankar, "Violence detection in video using computer vision techniques," in *Computer Analysis of Images and Patterns*, Springer Berlin Heidelberg, 2011, pp. 332–339.
- [17] P. Sernani, N. Falcionelli, S. Tomassini, P. Contardo, and A. F. Dragoni, "Deep learning for automatic violence detection: tests on the AIRTLab dataset," *IEEE Access*, vol. 9, pp. 160580–160595, 2021, doi: 10.1109/access.2021.3131315.
- [18] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 4489–4497, doi: 10.1109/iccv.2015.510.
- [19] M.-S. Kang, R.-H. Park, and H.-M. Park, "Efficient spatio-temporal modeling methods for real-time violence recognition," *IEEE Access*, vol. 9, pp. 76270–76285, 2021, doi: 10.1109/access.2021.3083273.
- [20] R. Vijeikis, V. Raudonis, and G. Dervinis, "Efficient violence detection in surveillance," *Sensors*, vol. 22, no. 6, p. 2216, Mar. 2022, doi: 10.3390/s22062216.
- [21] V. D. Huszár, V. K. Adhikarla, I. Négyesi, and C. Krasznay, "Toward fast and accurate violence detection for automated video surveillance applications," *IEEE Access*, vol. 11, pp. 18772–18793, 2023, doi: 10.1109/access.2023.3245521.
- [22] M. Cheng, K. Cai, and M. Li, "RWF-2000: An open large scale video database for violence detection," in *2020 25th International Conference on Pattern Recognition (ICPR)*, Jan. 2021, pp. 4183–4190, doi: 10.1109/icpr48806.2021.9412502.
- [23] G. Garcia-Cobo and J. C. SanMiguel, "Human skeletons and change detection for efficient violence detection in surveillance videos," *Computer Vision and Image Understanding*, vol. 233, p. 103739, Aug. 2023, doi: 10.1016/j.cviu.2023.103739.
- [24] "OpenCV: Histograms - 2: Histogram Equalization." [https://docs.opencv.org/4.x/d5/daf/tutorial\\_py\\_histogram\\_equalization.html](https://docs.opencv.org/4.x/d5/daf/tutorial_py_histogram_equalization.html) (accessed Oct. 30, 2024).
- [25] V. Rengarajan, A. Punnappurath, A. N. Rajagopalan, and G. Seetharaman, "Efficient change detection for very large motion blurred images," in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2014, pp. 315–322, doi: 10.1109/cvprw.2014.55.
- [26] P. I. Frazier, "A tutorial on Bayesian optimization." <https://arxiv.org/abs/1807.02811v1> (accessed Oct. 24, 2024).
- [27] D. P. Kingma and J. L. Ba, "Adam: a method for stochastic optimization," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. <https://arxiv.org/abs/1412.6980v9> (accessed Oct. 24, 2024).

## BIOGRAPHIES OF AUTHORS



**Erick**    is a graduate student of Master of Computer Science Department in Bina Nusantara University. His research interests include computer vision, machine learning, and image processing. He can be contacted at email: [erick007@binus.ac.id](mailto:erick007@binus.ac.id).



**Benfano Soewito**    received B.Sc. degree on Physics from Airlangga University, Surabaya, Indonesia on January 1991. He received M.Sc. and Ph.D. degree from the Electrical and Computer Engineering, Southern Illinois University, Carbondale, USA in 2005 and 2009 respectively. He is currently an Professor at Computer Science Department, Bina Nusantara University, Indonesia. His research interest include information security, sensor network, and mobile applications. He can be contacted at email: [bsoewito@binus.edu](mailto:bsoewito@binus.edu).