

# Two-stream Multi-dimensional Convolutional Network for Real-time Violence Detection

Dipon Kumar Ghosh and Amitabha Chakrabarty

*Department of Computer Science and Engineering, BRAC University, Dhaka, Bangladesh*

**Abstract**—The increasing number of surveillance cameras and security concerns have made automatic violent activity detection from surveillance footage an active area for research. Modern deep learning methods have achieved good accuracy in violence detection and proved to be successful because of their applicability in intelligent surveillance systems. However, the models are computationally expensive and large in size because of their inefficient methods for feature extraction. This work presents a novel architecture for violence detection called Two-stream Multi-dimensional Convolutional Network (2s-MDCN), which uses RGB frames and optical flow to detect violence. Our proposed method extracts temporal and spatial information independently by 1D, 2D, and 3D convolutions. Despite combining multi-dimensional convolutional networks, our models are lightweight and efficient due to reduced channel capacity, yet they learn to extract meaningful spatial and temporal information. Additionally, combining RGB frames and optical flow yields 2.2% more accuracy than a single RGB stream. Regardless of having less complexity, our models obtained state-of-the-art accuracy of 89.7% on the largest violence detection benchmark dataset.

**Index Terms**—Real-time violence detection, convolutional neural network (CNN), spatio-temporal feature extraction, surveillance system

## I. INTRODUCTION

There are several advantages to detecting violent activity from surveillance video. In today's world, security cameras may be found in practically every public area such as an office, hospitals, educational institutes, and shopping malls, among other places. As the number of security cameras grows, so does the demand for more sophisticated methods of monitoring the footages. Using manual methods to monitor and identify violence in real-time from video footage is time-consuming and costly. A further disadvantage is that it may take some time to notify the appropriate authorities responsible for taking action in the event of an emergency, while an automated violence detection system may do so nearly instantaneously.

When it comes to violence detection, it may be regarded as a subset of human action recognition, which seeks to recognize conventional human activity [?]. In image recognition,  $I(h, w)$ , the spatial characteristics  $h$  and  $w$  that offer information about the scene are generally extracted. In contrast, there is another dimension in video data  $I(t, h, w)$  that contains information about the passage of time, which reveals the changes in spatial features with time. It is required to extract both spatial and temporal information to detect violence from video data.

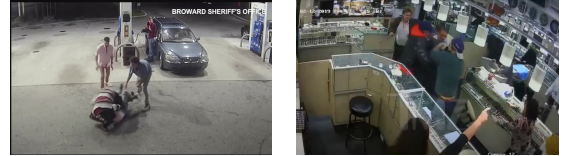
Previously, researchers employed a variety of feature extraction approaches, such as ViF [?], STIPs [?], iDT [?],



(a) Movies fight detection dataset samples.



(b) Hockey fight detection dataset samples.



(c) RWF-2000 violence detection dataset samples.

Fig. 1: Samples from different violent detection dataset.

and fed the results to classic classification models, such as support vector machine (SVM). Real-world settings, on the other hand, are complicated, as seen in Fig. ??, and it is difficult to extract relevant information from hand-crafted feature descriptors. Convolutional neural network (CNN), gated recurrent unit (GRU), and long short-term memory (LSTM) network are deep learning-based approaches that have recently been demonstrated to be effective in learning interpretable and robust features from images, identifying spatial information, and achieving cutting edge results on image classification, segmentation, and other computer vision tasks [?], [?]. Research has been carried out to extend the success of deep learning approaches to video analysis, with results that are at the state-of-the-art the field [?], [?].

The significance of extracted spatial and temporal features determines the success in violence detection as well as general human action recognition. Modern approaches, in addition to RGB frames, depend on optical flow for temporal information. Furthermore, because of the high cost of computational complexity and the vast number of parameters, they are inefficient for any real-world application. We believe that the problem lies in the way spatial and temporal characteristics are extracted in

a current deep-learning model. When developing the models, generally spatial and temporal characteristics are retrieved in a sequential manner, with temporal features extraction occurring after spatial data extraction. This results in a loss of information due to the fact that the features are not extracted from the same spatio-temporal position throughout the procedure.

To address this problem, we propose a novel architecture for violence detection named the two-stream multi-dimensional convolutional network (2s-MDCN), as illustrated in Fig. ?? . Our proposed 2s-MDCN uses both RGB and optical flow as input and thus, the model has two separate branches for RGB stream and optical flow stream. Our proposed model employs a fusion of multi-dimensional convolution processes to filter out spatial and temporal data separately retrieved from the same spatio-temporal point, therefore minimizing information loss. Our proposed method employs optical flow along with the RGB frames. To demonstrate the effectiveness of our model, we conduct extensive experiments on three benchmark datasets for violence detection, including RWF-2000 violence [?], Hockey-fight [?] and Movies-fight [?] dataset. We also performed experiments with a single RGB stream and optical flow stream. Our experiments show that adding optical flow increased the model accuracy by a significant amount. Despite having low parameters and less complexity, our model obtained state-of-the-art accuracy in violence detection benchmark datasets.

The primary contributions of this work can be summarized as follows:

- We explored how important it is to coordinate temporal and spatial features in violence detection.
- We proposed 2s-MDCN, a novel architecture for violence detection which takes RGB frames and optical flow as input. Our model extracts temporal and spatial features independently of each other whereas, current models do that in a sequential fashion which may not be suitable to fully exploit the features.
- Finally, this work can be considered a strong baseline for violence detection as our model achieved state-of-the-art accuracy in the largest violence detection dataset. We hope our baseline is helpful for future work in violent action detection.

## II. RELATED WORK

### A. Traditional Methods

Violence can be detected by traditional methods using hand-crafted feature extraction algorithms and using classical machine learning algorithms such as KNN, AVM, and Adaboost as a classifier. As done by, Harris corner detector [?], improved dense trajectory (iDT) [?], motion scale-invariant feature transform (MoSIFT) [?], space-time interest points (STIP) [?].

Hassner et al. [?] used optical flow magnitude series to detect violence in videos. The features are called violent flow (ViF) descriptors. Later, their method was improved by introducing the orientation in the violent flow (ViF) descriptors [?].

Improved dense trajectory features iDT [?] remarkably improved the accuracy of human action recognition. This method along with the fisher encoding method extracts significant spatio-temporal features from violent videos [?].

### B. Deep Learning Based Methods

Recently, deep learning-based methods gained much interest due to their improved accuracy and better performance than traditional methods. Convolutional neural network (CNN) [?], 3D convolutional network [?], and long short term memory (LSTM) network [?] are widely used architecture for the purpose of video understanding [?], [?], [?]. Different deep learning-based approaches have also been used in violence detection [?], [?].

Three input streams, RGB frames, optical flow, and accelerated flow maps, were used as input for violence detection [?]. They used LSTM network to filter out features from the input streams. Using their individual benefits, CNN and LSTM are used together in violent detection [?], [?]. The spatial features are extracted by the convolution layers and the mapping of temporal frames was done by LSTM layers. FightNet [?] used features from multiple streams, such as RGB frames, optical flow, and acceleration images, and fused them for violence detection. In [?], Flow-Gate network was proposed which used fusion of RGB frames and optical flow.

## III. THE PROPOSED 2S-MDCN

In this section, we discuss the architecture of our model in detail. Figure ?? shows the overall pipeline of our model. Raw RGB frames captured from cameras and optical flow are used as input to the model. First, the input is passed through a 3D convolutional layer which performs a 3D convolution operation and reduces the size of spatial dimension to prepare for the multi-dimensional convolutional (MDC) blocks. Two separate branches are used for RGB frames and optical flow, which are concatenated after the feature extraction from both branches. Each block in the model extracts spatial and temporal features independently from each other and detects violence.

### A. Multi-dimensional Convolutional (MDC) Block

The input to each one of the blocks is of shape  $C \times D \times H \times W$ , where  $C$  is the number of channels,  $D$  represents the frame numbers,  $H$  and  $W$  denotes height and width respectively. The values in  $H \times W$  contain spatial information of a particular frame while  $D$  frames contain temporal information for the corresponding pixel. As figure ?? illustrates, each block of our proposed model contains three main convolutional layers for extracting spatial and temporal features.

**1D convolutional layer:** The first module is 1D convolutional layer which extracts temporal features from the input. The kernel size of this convolutional layer is  $k_t \times 1 \times 1$ . So that it performs convolution operation on a particular pixel over  $k_t$  frames and extracts only temporal information of the corresponding pixel.

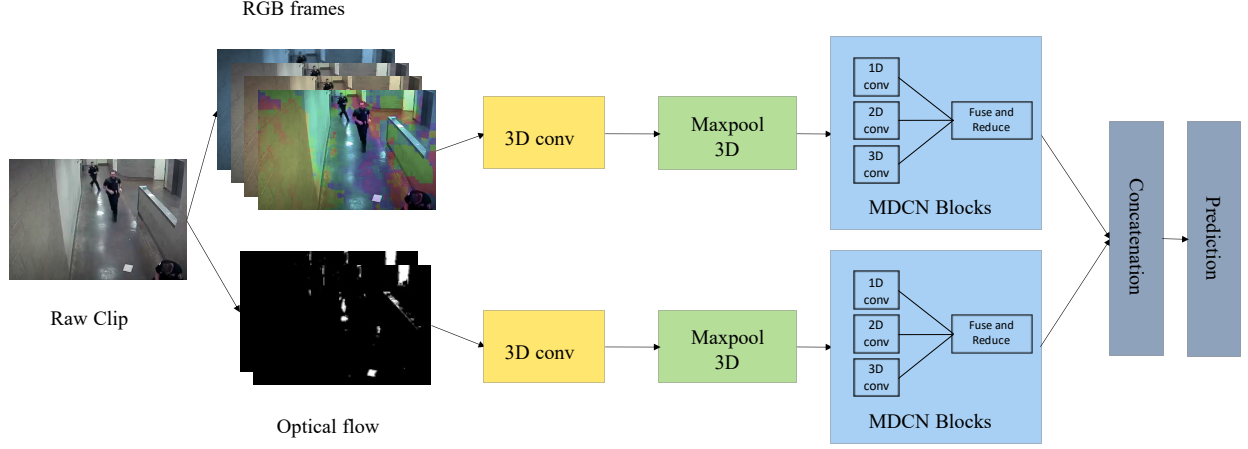


Fig. 2: Overall pipeline of 2s-MDCN.

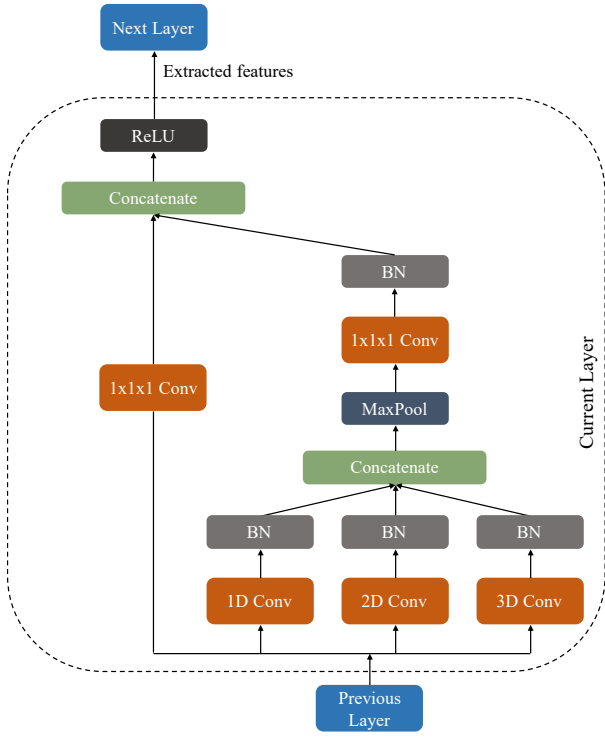


Fig. 3: A single multi-dimensional convolutional block.

**2D convolutional layer:** The second module is 2D convolutional layer which extracts temporal features from the input. The kernel size of this convolutional layer is  $1 \times k_s \times k_s$ . So that it performs convolution operation on a single frame and extracts only spatial information from the particular frame.

**3D convolutional layer:** The second module is 3D convolutional layer which extracts both spatial and temporal features from the input. The kernel size of this convolutional layer is  $k_t \times k_s \times k_s$ . So that it performs convolution operation over  $k_t$  frames and extracts temporal and spatial frames together.

Each of these three modules is followed by a batch normal-

ization layer. In our model, we used 3 the value of  $k_t$  and  $k_s$ . They extract information independently from the same spatio-temporal position of the input and fuse together. In this process, our model can extract all the information from the input which reduces information loss and improves accuracy. Then, the fused features are passed through a *maxpooling* layer and a  $1 \times 1 \times 1$  convolution layer which reduces the number of channels for the next layer. This reduction module keeps our model size small. Moreover, a concatenated skip connection [?], followed by a ReLU layer, is added to stabilize the model which also helped to improve accuracy. Skip connection concatenates features from the previous layers to the current layer which allows more information to be obtained from the previous layers and reduces loss of information. Concatenated skip connections also help the gradient to propagate better and fix the vanishing gradient problem. To match the input channel to the output channel, we have used a convolutional layer of kernel size  $1 \times 1 \times 1$  and stride of size  $1 \times 2 \times 2$ .

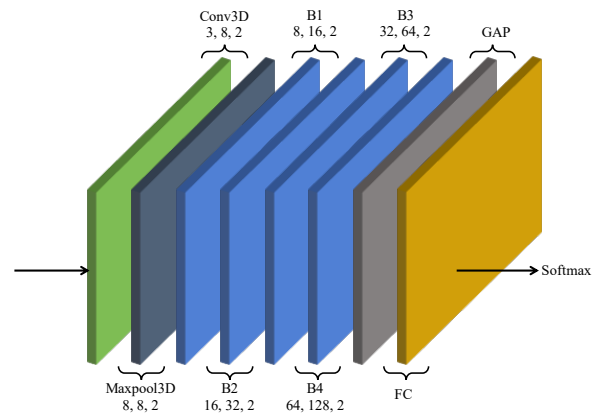


Fig. 4: Layers of 2s-MDCN.

### B. Multi-dimensional Convolutional Network

As figure ?? illustrates, our proposed network consists of 4 MDC blocks and a 3D convolutional layer at the beginning.

Layers		2s-MDCN		Output Size
		RGB	optical flow	
input				$3 \times 32 \times 224^2$ (RGB) $2 \times 32 \times 224^2$ (optical flow)
conv <sub>1</sub>		conv 3D: $5 \times 7^2$ , stride: $1 \times 2^2$		$8 \times 32 \times 112^2$
pool <sub>1</sub>		max pool: $1 \times 3^2$ , stride: $1 \times 2^2$		$8 \times 32 \times 56^2$
mdcn <sub>1</sub>	conv layer	conv 1D: $3 \times 1^2$ conv 2D: $1 \times 3^2$ conv 3D: $3 \times 3^2$		$16 \times 32 \times 28^2$
	fuse and reduce	max pool: $1 \times 3^2$ , stride: $1 \times 2^2$ conv: $1 \times 1^2$		
mdcn <sub>2</sub>	conv layer	conv 1D: $3 \times 1^2$ conv 2D: $1 \times 3^2$ conv 3D: $3 \times 3^2$		$32 \times 32 \times 28^2$
	fuse and reduce	max pool: $1 \times 3^2$ , stride: $1 \times 2^2$ conv: $1 \times 1^2$		
mdcn <sub>3</sub>	conv layer	conv 1D: $3 \times 1^2$ conv 2D: $1 \times 3^2$ conv 3D: $3 \times 3^2$		$64 \times 32 \times 14^2$
	fuse and reduce	max pool: $1 \times 3^2$ , stride: $1 \times 2^2$ conv: $1 \times 1^2$		
mdcn <sub>4</sub>	conv layer	conv 1D: $3 \times 1^2$ conv 2D: $1 \times 3^2$ conv 3D: $3 \times 3^2$		$128 \times 32 \times 7^2$
	fuse and reduce	max pool: $1 \times 3^2$ , stride: $1 \times 2^2$ conv: $1 \times 1^2$		
concatenate, global average pool, fc				#classes (2)

TABLE I: Overall architecture of 2s-MDCN.

Both RGB and optical flow streams have identical numbers of MDC blocks. The first 3D convolutional layer is of kernel size  $5 \times 7 \times 7$  and stride of size  $1 \times 2 \times 2$ . This layer is followed by a *maxpool* layer with stride value of  $1 \times 2 \times 2$ . The convolution layer and *maxpool* layer are used to reduce the size of spatial dimension. The output from these layers is passed into the 4 MDC blocks, where temporal and spatial information are extracted. The numbers of channels of output from the convolution blocks are 8, 16, 32, 64, and 128. After extracting features from RGB and optical flow stream, they are concatenated and a *global average pooling* layer is used to combine and reduce the extracted feature and finally, a fully connected (FC) and *softmax* layer is used to detect violence. Table ?? displays the kernel size, stride size, and number output channel of each block in detail. The shape of convolutional kernel is denoted by  $T \times S^2$ , where  $T$  kernel temporal dimension and  $S$  indicated spatial dimension. Stride is also represented in the same manner (temporal stride, spatial stride<sup>2</sup>). The output size is represented by channel $\times$ temporal length $\times$ spatial dimension<sup>2</sup>. The features are passed into *conv layers* independently and then concatenated and passed into *fuse and reduce* layer.

#### IV. EXPERIMENTS

In this section, we analyze the performance of our model on public violence detection datasets. Our model outperforms the previous models and achieves state-of-the-art accuracy in spite of having low computational complexity and less memory footprint.

##### A. Datasets

We used three benchmark violence detection datasets for training and validating our model, namely RWF-200 violence dataset [?], Hockey-fight dataset [?] and Movies-Fight dataset [?]. At the moment, the RWF-2000 violence dataset is the largest dataset for violence detection with 2000 clips. 1600 clips out are kept for training and 400 are used for validation. The clips from the dataset are collected from real-world surveillance cameras. The number of characters in the clips are not fixed, dynamic characteristics vary a lot and the background is complicated. Hockey-fight dataset is collected from video footage of hockey games. There are 1000 clips in the dataset, where half of them are violent and the rest are non-violent. Movies-fight dataset consists of clips from movies with a total of 200 clips.

##### B. Training details

For training, we sampled 32 frames from each clip and each frame was resized to  $224 \times 224$ . Thus, our input size for the was  $3 \times 32 \times 224 \times 224$ . Following the procedure of [?], we used brightness transformation and random rotation to augment our data in order to prevent overfitting. We implemented our model by using PyTorch [?] which is a deep-learning framework. Stochastic gradient descent (SGD) optimizer was used to train our network with nesterov momentum [?] of value 0.9. The value of weight decay was set to  $1e^{-3}$ . The initial learning rate was set to 0.1, which was reduced by a factor of 10 after 25 and 75 epochs. We trained our model for 100 epochs with a batch size of 16. Our model was trained from scratch with the corresponding datasets.

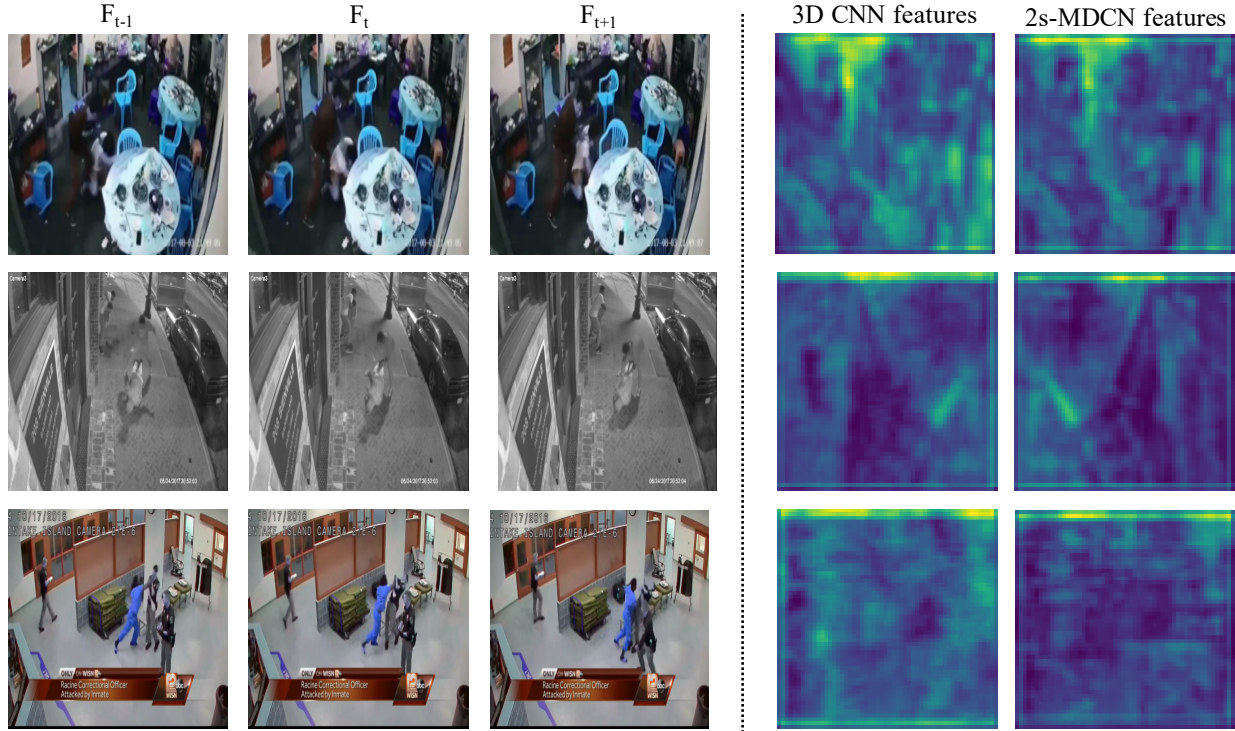


Fig. 5: Visualization of extracted features from 2s-MDCN.

## V. RESULTS

In this section, we report the outcome of the experiments which we performed with our proposed model. This includes extensive ablation study and comparison with current state-of-the-art models. All of the evidence points to our model being competitive with other state-of-the-art models, as well as lightweight and cost-effective.

Methods	Accuracy (%)
2s-MDCN(without skip connection)	86.75
2s-MDCN(with skip connection)	87.50

TABLE II: Ablation study for the impact of concatenated skip connections on 2s-MDCN.

Methods	Accuracy (%)
2s-MDCN (flow only)	80.50
2s-MDCN (RGB only)	87.50
2s-MDCN (fusion)	89.70

TABLE III: Ablation study of the performance of 2s-MDCN with respect to input type.

Methods	Frame Length	Accuracy (%)
2s-MDCN	16	86.30
2s-MDCN	32	89.7

TABLE IV: Ablation study of the performance of 2s-MDCN with respect to frame length.

### A. Ablation Study

We performed different ablation studies in order to identify the best hyperparameters, input features and model architecture.

First, we compared our model without skip connection. Table ?? displays the model accuracy with and without the concatenation of skip connection. Our model achieves 89.7% with skip connection and while we turned off the skip connection it drops at 88.7%. The results show that skip connection helps information flow more efficiently and thus helps to improve accuracy.

We also demonstrate the capability of our model with a single input stream. As shown in Table ??, our model achieves 89.7% accuracy when two different input methods are used. We achieve 87.5% accuracy while using only RGB stream, and by using only optical flow our model is able to achieve 78.5% accuracy in RWF-2000 violence detection dataset.

Second, as shown in Table ??, we experimented with different frame lengths and found that frame length of 32 gives the best accuracy for our model. When we used 16 frames as input we achieved a score of 85.3%.

### B. Qualitative Analysis

We illustrate visualization of features from an individual DSTC block and compare with the features from a single 3D CNN layer, which is used as a branch in our model. In the Fig. ??, we show three consecutive frames of samples from RWF-2000 dataset denoted by  $F_{t-1}$ ,  $F_t$ , and  $F_{t+1}$ . We also illustrate the features extracted from the first DSTC block



Methods	# Params (M)	Complexity (GFLOPS)	Accuracy(%)
R(2+1)D [?]	33.20	42.42	81.25
C3D [?]	79.90	38.62	82.75
ConvLSTM [?]	-	-	77.00
I3D (RGB) [?]	12.30	111.30	85.75
I3D (flow) [?]	12.30	102.52	75.50
I3D (two-stream) [?]	24.40	213.85	81.50
FlowGate (RGB) [?]	0.25	8.76	84.50
FlowGate (flow) [?]	0.25	8.29	75.50
FlowGate (fusion) [?]	0.27	16.98	87.25
<b>2s-MDCN (flow only)</b>	<b>0.47</b>	<b>4.47</b>	<b>80.50</b>
<b>2s-MDCN (RGB only)</b>	<b>0.47</b>	<b>4.47</b>	<b>87.50</b>
<b>2s-MDCN (fusion)</b>	<b>0.94</b>	<b>8.16</b>	<b>89.70</b>

TABLE V: Comparison of our model with other state-of-the-art methods on RWF-2000 violence dataset.

TABLE VI: Comparison of our model with other state-of-the-art methods on Hockey-fight and Movies-fight dataset.

Methods	Accuracy(%)	
	Hockey	Movies
ViF [?]	82.9	-
LHOG+LOF [?]	95.1	-
HOF+HIK [?]	88.6	59.0
HOF+HIK [?]	91.7	49.0
MoWLD+BoW [?]	91.9	-
MoSFIT+HIK [?]	90.9	89.5
FightNet [?]	97.0	100
3D ConvNet [?]	99.62	99.9
ConvLSTM [?]	97.1	100
C3D [?]	96.5	100
I3D (RGB) [?]	98.5	100
I3D (Flow) [?]	84.0	100
FlowGate [?]	98.0	100
<b>2s-MDCN</b>	<b>99.0</b>	<b>100</b>

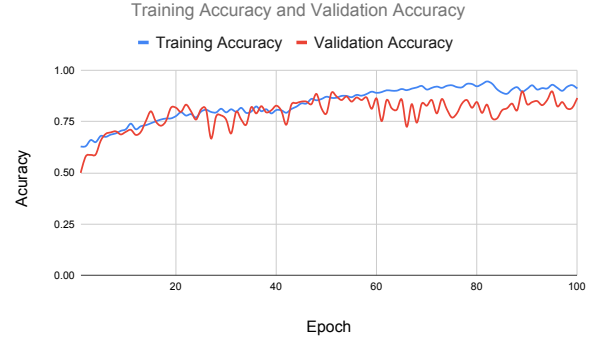
and compare them with the single 3D CNN layer. From the visualization, it is evident that the combination of 1D, 2D, and 3D CNN layers extract salient features from the input and makes our model efficient and accurate.

### C. Comparison with the state-of-the-art

In this section, we compare our models with other violence detection state-of-the-art models on the datasets stated earlier.

Table ?? shows the comparison of our model with other models on Hockey-fight and Movie-fight dataset. Our model outperforms the hand-crafted features based models and deep learning-based models as well on both datasets. 2s-MDCN obtained 100.0% and 99.0% accuracy on Hockey-fight dataset and Movies-fight dataset respectively.

The comparison on the RWF-2000 violence dataset is shown in the table ?. Hence, we have also reported the number of parameters (M) and computational complexity along with accuracy. Computation complexity is expressed in GFLOPs ( $10^9$  FLOPs), where, 1 FLOP is defined as 1 floating-point multiple-addition operation [?]. 2s-MDCN (RGB only) achieved 87.50% accuracy only using RGB clips whereas, FlowGate (RGB) achieved 84.50% with twice the complexity of our model. Moreover, our model also outperformed FlowGate with the fusion of RGB frames and optical flow, though our model has a computational complexity of 4.47 GFLOPs



(a) Comparison between training and validation accuracy.



(b) Comparison between training and validation loss

Fig. 6: Performance measurement between training and validation process.

whereas FlowGate performs 16.98 GFLOPs. 2s-MDCN (fusion) achieved an accuracy of 89.7% with 0.92M parameters and computational complexity of 8.97 GFLOPs. 2s-MDCN also outperformed other models listed in Table ?? in terms of accuracy, memory consumption and cost. Thus, our model achieves state-of-the-art accuracy with lower computational cost and less parameter size. Furthermore, the exclusion of optical flow eliminates the overhead of pre-processing of input which made our model more efficient.

Additionally, in Figure ??, a comparison between the loss and accuracy of the model in training and validation phase

on RWF-2000 dataset is reported. As illustrated in Figure ??, accuracy was stable both in the training and validation process during the whole training. However, the loss shows a slight overfitting during epoch 25 to 35 as illustrated in Figure ?. Later, it is fixed as training progress and the model learns the features.

Unlike regular human action recognition, violence detection is more complicated because of the involvement of more motion, complex background, and dynamic of character varies more. In spite of being smaller in size, our model extracts information from this complex features and achieves state-of-the-art accuracy, and also can be used in real-world situations.

Model	Processing Speed (FPS)	
	CPU	Jetson Nano
R(2+1)D [?]	7.5	13.0
C3D [?]	11.2	18.8
I3D (two-stream) [?]	3.68	12.6
FlowGate (fusion) [?]	12.4	58.18
2s-MDCN (RGB only)	<b>16.6</b>	<b>80.0</b>
2s-MDCN (fusion)	<b>13.3</b>	<b>62.0</b>

TABLE VII: Comparison of processing speed on CPU and Jetson Nano.

#### D. Performance analysis of 2s-MDCN on edge devices

The performance of our model on edge devices is also shown to demonstrate the deployability of 2s-MDCN model in a real-time surveillance system. We show the processing speed in terms of frames per second (FPS), in Table ?. We report FPS of our 2s-MDCN and other state-of-the-art models, both on a central processing unit (CPU) and an edge device called Jetson Nano. Our model can perform at 16.6 FPS on CPU and 80 FPS on Jetson Nano with only RGB frames, which makes our model more than 37% faster than FlowGate(fusion) on Jetson Nano. When combined with optical flow, 2s-MDCN processes 13.3 frames per second on CPU and 62 frames on Jetson Nano, which makes it suitable for surveillance systems. Our 2s-MDCN model, therefore, performs better and faster with low computational cost, which makes it a viable choice for real-time violence detection applications.

## VI. CONCLUSION

In this work, we have represented a novel architecture for violence detection. This model achieves state-of-the-art accuracy in the largest violence detection dataset. This is possible by extracting efficient temporal and spatial features and reducing the loss of information during the prediction process. Our model outperformed FlowGate(fusion) with almost one-fourth of computational cost and better accuracy. As our model takes less memory, consumes less power, and removes pre-processing overhead, it becomes a suitable candidate for a violence detection model which can be deployed in real-life scenarios.

## REFERENCES

- [1] Piotr Bilinski and Francois Bremond. Human violence recognition and detection in surveillance videos. In *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 30–36. IEEE, 2016.
- [2] Aleksandar Botev, Guy Lever, and David Barber. Nesterov’s accelerated gradient and momentum as approximations to regularised update descent. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 1899–1903. IEEE, 2017.
- [3] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.
- [4] Datong Chen, Howard Wactlar, Ming-yu Chen, Can Gao, Ashok Bharucha, and Alex Hauptmann. Recognition of aggressive human behavior using binary local motion descriptors. In *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 5238–5241. IEEE, 2008.
- [5] Ming Cheng, Kunjing Cai, and Ming Li. Rwf-2000: an open large scale video database for violence detection. *arXiv preprint arXiv:1911.05913*, 2019.
- [6] Fillipe DM De Souza, Guillermo C Chavez, Eduardo A do Valle Jr, and Arnaldo de A Araújo. Violence detection in video using spatio-temporal features. In *2010 23rd SIBGRAPI Conference on Graphics, Patterns and Images*, pages 224–230. IEEE, 2010.
- [7] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.
- [8] Zhihong Dong, Jie Qin, and Yunhong Wang. Multi-stream deep networks for person to person violence detection in videos. In *Chinese Conference on Pattern Recognition*, pages 517–531. Springer, 2016.
- [9] Yuan Gao, Hong Liu, Xiaohu Sun, Can Wang, and Yi Liu. Violence detection using oriented violent flows. *Image and vision computing*, 48:37–41, 2016.
- [10] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018.
- [11] Tal Hassner, Yossi Itcher, and Orit Kliper-Gross. Violent flows: Real-time detection of violent crowd behavior. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–6. IEEE, 2012.
- [12] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, Nov. 1997.
- [13] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.
- [14] 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*, 35(1):221–231, 2013.
- [15] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.
- [16] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.
- [17] Enrique Bermejo Nieves, Oscar Deniz Suarez, Gloria Bueno Garcia, and Rahul Sukthankar. Hockey fight detection dataset. In *Computer Analysis of Images and Patterns*, pages 332–339. Springer, 2011.
- [18] Enrique Bermejo Nieves, Oscar Deniz Suarez, Gloria Bueno Garcia, and Rahul Sukthankar. Movies fight detection dataset. In *Computer Analysis of Images and Patterns*, pages 332–339. Springer, 2011.
- [19] Enrique Bermejo Nieves, Oscar Deniz Suarez, Gloria Bueno García, and Rahul Sukthankar. Violence detection in video using computer vision techniques. In *International conference on Computer analysis of images and patterns*, pages 332–339. Springer, 2011.
- [20] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.
- [21] Xingjian Shi, Zhourong Chen, Hao Wang, Dit Yan Yeung, Wai Kin Wong, and Wang Chun Woo. Convolutional lstm network: A machine

- learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 2015:802–810, 2015.
- [22] Wei Song, Dongliang Zhang, Xiaobing Zhao, Jing Yu, Rui Zheng, and Antai Wang. A novel violent video detection scheme based on modified 3d convolutional neural networks. *IEEE Access*, 7:39172–39179, 2019.
  - [23] Swathikiran Sudhakaran and Oswald Lanz. Learning to detect violent videos using convolutional long short-term memory. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2017.
  - [24] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.
  - [25] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.
  - [26] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018.
  - [27] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pages 3551–3558, 2013.
  - [28] Long Xu, Chen Gong, Jie Yang, Qiang Wu, and Lixiu Yao. Violent video detection based on mosift feature and sparse coding. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3538–3542. IEEE, 2014.
  - [29] Tao Zhang, Wenjing Jia, Baoqing Yang, Jie Yang, Xiangjian He, and Zhonglong Zheng. Mowld: a robust motion image descriptor for violence detection. *Multimedia Tools and Applications*, 76(1):1419–1438, 2017.
  - [30] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018.
  - [31] Peipei Zhou, Qinghai Ding, Haibo Luo, and Xinglin Hou. Violent interaction detection in video based on deep learning. In *Journal of physics: conference series*, volume 844, page 012044. IOP Publishing, 2017.
  - [32] Peipei Zhou, Qinghai Ding, Haibo Luo, and Xinglin Hou. Violence detection in surveillance video using low-level features. *PLoS one*, 13(10):e0203668, 2018.
  - [33] Mohammadreza Zolfaghari, Kamaljeet Singh, and Thomas Brox. Eco: Efficient convolutional network for online video understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 695–712, 2018.