Metadata of the chapter that will be visualized in SpringerLink

Book Title	Neural Information Processing	
Series Title		
Chapter Title	Feature Fusion Based Deep Spatiotemporal Model for Violence Detection in Videos	
Copyright Year	2019	
Copyright HolderName	Springer Nature Switzerland AG	
Corresponding Author	Family Name	**Asad**
	Particle	
	Given Name	**Mujtaba**
	Prefix	
	Suffix	
	Role	
	Division	Institute of Image Processing and Pattern Recognition
	Organization	Shanghai Jiao Tong University
	Address	Shanghai, China
	Email	asadmujtaba@sjtu.edu.cn
	ORCID	http://orcid.org/0000-0003-0318-7379
Author	Family Name	**Yang**
	Particle	
	Given Name	**Zuopeng**
	Prefix	
	Suffix	
	Role	
	Division	Institute of Image Processing and Pattern Recognition
	Organization	Shanghai Jiao Tong University
	Address	Shanghai, China
	Email	yzpeng@sjtu.edu.cn
Author	Family Name	**Khan**
	Particle	
	Given Name	**Zubair**
	Prefix	
	Suffix	
	Role	
	Division	Institute of Image Processing and Pattern Recognition
	Organization	Shanghai Jiao Tong University
	Address	Shanghai, China
	Email	zubairkhan@sjtu.edu.cn
Corresponding Author	Family Name	**Yang**
	Particle	
	Given Name	**Jie**
	Prefix	

	Suffix	
	Role	
	Division	Institute of Image Processing and Pattern Recognition
	Organization	Shanghai Jiao Tong University
	Address	Shanghai, China
	Email	jieyang@sjtu.edu.cn
Author	Family Name	**He**
	Particle	
	Given Name	**Xiangjian**
	Prefix	
	Suffix	
	Role	
	Division	School of Electrical and Data Engineering
	Organization	University of Technology Sydney
	Address	Ultimo, Australia
	Email	xiangjian.he@uts.edu.au

Abstract

It is essential for public monitoring and security to detect violent behavior in surveillance videos. However, it requires constant human observation and attention, which is a challenging task. Autonomous detection of violent activities is essential for continuous, uninterrupted video surveillance systems. This paper proposed a novel method to detect violent activities in videos, using fused spatial feature maps, based on Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) units. The spatial features are extracted through CNN, and multi-level spatial features fusion method is used to combine the spatial features maps from two equally spaced sequential input video frames to incorporate motion characteristics. The additional residual layer blocks are used to further learn these fused spatial features to increase the classification accuracy of the network. The combined spatial features of input frames are then fed to LSTM units to learn the global temporal information. The output of this network classifies the violent or non-violent category present in the input video frame. Experimental results on three different standard benchmark datasets: Hockey Fight, Crowd Violence and BEHAVE show that the proposed algorithm provides better ability to recognize violent actions in different scenarios and results in improved performance compared to the state-of-the-art methods.

Keywords

Violence detection - CNN - LSTM - Autonomous video - Surveillance spatiotemporal features

Feature Fusion Based Deep Spatiotemporal Model for Violence Detection in Videos

Mujtaba Asad[1(✉)], Zuopeng Yang[1], Zubair Khan[1], Jie Yang[1(✉)], and Xiangjian He[2]

[1] Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai, China
{asadmujtaba,yzpeng,zubairkhan,jieyang}@sjtu.edu.cn
[2] School of Electrical and Data Engineering, University of Technology Sydney, Ultimo, Australia
xiangjian.he@uts.edu.au

Abstract. It is essential for public monitoring and security to detect violent behavior in surveillance videos. However, it requires constant human observation and attention, which is a challenging task. Autonomous detection of violent activities is essential for continuous, uninterrupted video surveillance systems. This paper proposed a novel method to detect violent activities in videos, using fused spatial feature maps, based on Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) units. The spatial features are extracted through CNN, and multi-level spatial features fusion method is used to combine the spatial features maps from two equally spaced sequential input video frames to incorporate motion characteristics. The additional residual layer blocks are used to further learn these fused spatial features to increase the classification accuracy of the network. The combined spatial features of input frames are then fed to LSTM units to learn the global temporal information. The output of this network classifies the violent or non-violent category present in the input video frame. Experimental results on three different standard benchmark datasets: Hockey Fight, Crowd Violence and BEHAVE show that the proposed algorithm provides better ability to recognize violent actions in different scenarios and results in improved performance compared to the state-of-the-art methods.

AQ1

Keywords: Violence detection · CNN · LSTM · Autonomous video · Surveillance spatiotemporal features

1 Introduction

The use of surveillance cameras is essential nowadays, as the rate of public violence has increased in the recent era. To make cities safe from undesirable events, it is important to continuously monitor for detrimental events before subsequent

T. Gedeon et al. (Eds.): ICONIP 2019, LNCS 11953, pp. 1–13, 2019.
https://doi.org/10.1007/978-3-030-36708-4_33

disorder, as violent behavior or criminal activity may occur. As the probability of occurrence of these types of events is diminutive and manually identifying abnormal behavior is a difficult task. Observing the surveillance footage requires constant attention and human resources, so it is a primarily infeasible and tiresome task for a human to monitor a surveillance video uninterruptedly. It is necessary to have a system that can monitor a video feed incessantly and has the ability to automatically detect violent activities without the supervision of a human. Violent behavior detection relies on techniques that are used in related fields of computer vision for action recognition, object detection, tracking, and video classification [8,29]. Compared to action recognition and other computer vision applications, there is not much work for violence detection. There are difficulties in detecting the violent action of a person. For example, some actions may look aggressive or violent, but they may actually be normal. Detection of abnormal behavior is somewhat of a subjective nature, so it may somehow be misclassified. The CCTV cameras, through which the surveillance videos are acquired, are mostly of low resolution and occasionally miss the minor details that might be useful for detecting a particular action. There are some methods such as [3,4,27] which used both the visual and audio features for detection of violent actions. However, most of the CCTV surveillance cameras have only visual information and do not include any audio information, although the latter might be valuable for identifying a specific action related to violence. The presented work in this paper uses only visual features for detection of violent behavior.

Most of the existing techniques, for action recognition, used handcrafted based features, which takes a considerable amount of computational resources and are application specific, so they are not generalized to work on different datasets. The rapid progress in deep learning has also helped to achieve many tasks related to computer vision, such as image classification and object detection [22], video classification [21] action recognition [32] and image fusion [28] etc. A deep-learning architecture can well extract features on unseen data. Furthermore, it does not need a complex prior pre-processing on the data. Contrary to hand-crafted features, a deep-learning architecture can learn features in the form of raw-pixels, so it can be applied to various other related tasks without too much altering the architecture.

In this paper, we proposed a novel end-to-end trainable architecture to learn both spatial and temporal features to efficiently detect violent behavior in videos. Our contributions are as follow:

- Instead of using optical flow as motion characteristics, the proposed method used multi-level spatial feature fusion of two sequential frames to incorporate motion information.
- Additional residual layer blocks are added to the pre-trained CNN to learn multi-level fused spatial feature maps that generate a combined feature vector of two sequential frames which incorporates both local and global motion patterns.

The remaining organization of the paper is in following order. Section 2 discusses the related work in the field. Section 3 explains the proposed architecture and implementation details. Section 4 describes the experimental details and a comparison of the proposed scheme with the existing methods. The conclusion of the paper is presented in Sect. 5.

2 Related Work

Many methods for violence detection have been proposed using visual features [8,29] audio features [14], or both audio and visual features [3,4,15]. As most of the surveillance videos do not have audio information, so our work in this paper considers only the visual features. The existing violence detection architectures can be grouped into two categories: the first ones using handcrafted based features and the latter ones using features learned by an end-to-end deep-learning based model. Before the deep-learning era, most state-of-the-art approaches were based on handcrafted features, which were extracted manually, and used a learning model to learn local features, where the outliers are considered as abnormal events. Local features were first introduced for action recognition and most of the techniques were based on these features, including Space-Time Interest Points (STIP) [23], Histogram of Oriented Flows (HOFs) [7], Histogram of Oriented Gradient (HOGs) [6], Bag of Words(BoW) [5] and Motion Scale-Invariant Feature Transform (MoSIFT) [2]. Bermejo et al. [29] proposed a method for detecting violent behavior by combining the Scale Invariant Feature Transform (SIFT) and MoSIFT feature descriptors with (BoW) features. In [9] de Souza et al. compared both (SIFT) [24] and STIP features and presented that having both spatial and temporal features increase the accuracy compared with the approaches using only scale-invariant features. In [18] Hassner et al. introduced a new feature descriptor named Violent Flow (ViF) to classify violent and non-violent videos. To learn the feature descriptor this technique used a simple liner Support Vector Machine (SVM). Datta et al. in [8] used a trajectory-based method to extract motion information and the directions of moving of human limbs to detect violent activities. Several other techniques also used similar motion patterns to extract the spatiotemporal features. Xu et al. in [38] proposed a method using sparse coding by combining the MoSIFT and (BoW) descriptors for feature extraction and used kernel density estimation for selecting low noise features.

Recently, with the development of deep-learning architectures particularly for image classification, a convolutional neural network (CNN) can effectively learn spatial features from images [22]. Following the exceptional capability of CNN for extracting both high-level and low-level image features, Karpahty et al. [21] proposed a method for action recognition by exploiting CNNs to learn spatiotemporal features from videos using the fusion information of multiple frames across time. Simonyan and Zisserman [32] used two separate CNNs, one for learning spatial features and the other for learning temporal features. They used optical flow frames as input and used an SVM for classification by combining the scores from the two networks. Following the work of [21],

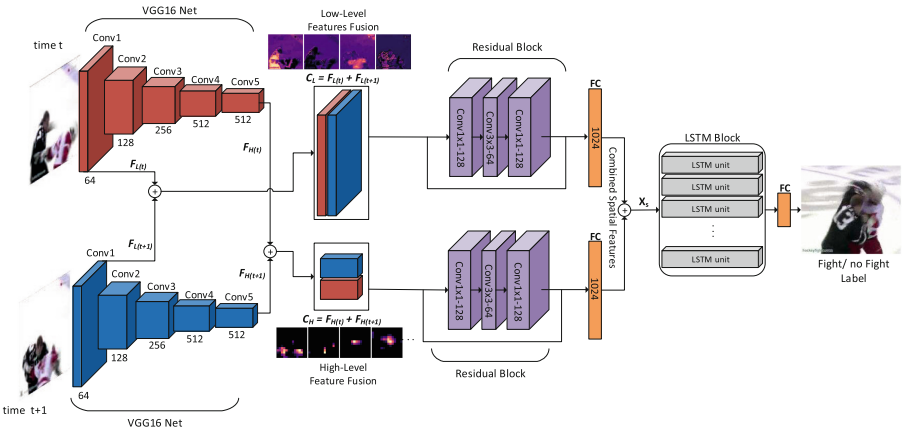

Fig. 1. The architecture of propose violence detection network. The models take two sequential input frames from the video and combine the low and high-level features maps of both frames whereas the additional residual layers are used to learn these fused features of two frames, then the LSTM units learn the temporal patterns in the sequence of two frames to effectively classify the violent actions.

Tran et al. proposed a 3D convolutional network to learn both spatial and temporal information from a video volume but they were unable to learn long temporal dependencies. Donahue et al. [11] provided a full end-to-end trainable model in the form of a Long-term Recurrent Convolutional Network (LRCN), which combined CNN and Recurrent Neural Network (RNN) and used Long Short-Term Memory (LSTM) for temporal feature learning at the end of convolution blocks. Similar to this technique Xinjian et al. [36] proposed a convLSTM architecture for precipitation nowcast by replacing the Fully Connected (FC) layer of the convolutional network with both FC layers and LSTMs or (FC-LSTM). In [25] Medel and Savakis used convLSTMs for the purpose of detecting anomalies in videos. Recently Sultani et al. [35] used a 3D-CNN to extract features from videos, labeled them as positive and negative bags and then incorporated multiple instance rankings to classify videos into abnormal or normal events. In [34], Sudhakaran and Lanz use LSTM along with CNN for detection of violent actions. Similarly, Hanson et al. [17] used bi-directional LSTM instead of simple LSTMs for the same purpose while giving good results.

Based on these above-discussed techniques, this proposed work exploits fusion of spatial feature maps from different layers of CNNs along with LSTM network to efficiently detect violent actions in videos at the frame level. Our method has an ability to run on variable length videos resulting in improved accuracy compared to the existing methods.

3 Proposed Network Architecture

This paper proposes a novel method to detect violent activities in videos using an end-to-end trainable model as shown in Fig. 1, which uses a combination of convolutional neural networks, residual layer blocks and long short-term memory (LSTM) network to learn both spatial and temporal features from the sequence of input frames to classify the videos into violent or non-violent categories. The spatial features from two convective frames are combined using the fusion of low and high-level feature maps to incorporate motion characteristics in the spatial features whereas the LSTM units are used to learn the temporal dependencies between the sequential frames.

3.1 Spatial Feature Learning Through Fusion

In order to extract spatial feature maps from the input video frames, the network takes advantage of transfer learning by using pre-trained VGG-16 [33] network on ImageNet datasets which extracts both low-level and high-level feature maps from input frames. To learn the local and global motion patterns, it is important to include, change of motion information across multiple frames, in this paper instead of using optical flow as motion feature which is computationally complex and results in slow training, spatial feature fusion of sequential video frames is used to assimilate the motion information. The two sequential video frames are used as input to the network. In order to incorporate local motion patterns the low-level features of the frame at time t are combined with low-level features of the frame at time $t+1$. Similarly, for global motion changes, the high-level features of frame t are combined with the high-level features of the frame at time $t+1$. Where t is the current timestamp of the input video frame.

Let $F_{L(t)}, F_{L(t+1)} \in \mathbf{R}^{h \times w \times d}$ denote the low-level feature maps of frames at time t and $t+1$ and $F_{H(t)}, F_{H(t+1)} \in \mathbf{R}^{h \times w \times d}$ represents high-level feature maps of frames at time t and $t+1$ respectively. Where h, w represents the height and width, and d is the channel depth. The low-level feature concatenation is described by the Eq. (1) whereas high-level features are concatenated by using Eq. (2)

$$C_L = F_{L(t)} + F_{L(t+1)} \tag{1}$$

$$C_H = F_{H(t)} + F_{H(t+1)} \tag{2}$$

Here C_L and C_H represents the combined low-level and high-level features of two sequential input frames, respectively. To further improve the classification accuracy and to add more depth to the network, the fully connected layers of the pre-trained network are replaced by residual layer blocks [19] which are used to learn the fused low-level (C_L) and high-level (C_H) features maps. Separate residual layer blocks are incorporated to learn C_L and C_H as shown in Fig. 1. The output from each residual layer block can be represented by Eqs. 3 and 4.

$$R_1 = f(C_L, \{W_{C_L}\}) + C_L \tag{3}$$

$$R_2 = f(C_H, \{W_{C_H}\}) + C_H \tag{4}$$

where f is the identity function learned by residual block and W_{C_L} and W_{C_H} are the respective weights of each residual block. The output from fully connected layers of residual layer blocks is then fused together to form a combined spatial feature vector X_s for two sequential input frames. This combined feature vector X_s is then fed to the LSTM unit to learn temporal dependencies.

3.2 Temporal Feature Learning Based on LSTMs

To learn long term temporal dependencies in the input sequences, LSTMs provide very promising results, specially, in the field of natural language processing (NLP), machine translation, image captioning, and other problems involving sequence learning. As videos consist of sequences of image frames, so to successfully classify the videos into violent or non-violent categories, it is important to capture the temporal information by incorporating LSTM units. Unlike simple RNN, LSTM has memory cells and gated inputs through which it selects which input to pass through and which one to forget. This addition of memory cells and gates provided LSTM the ability to overcome the exploding and vanishing gradients problem [20] and made it well suited for the applications of sequence learning. The following Eqs. (5) to (10) shows the working of an LSTM cell unit.

$$f_t = \sigma_g(W_f.[h_{t-1}, x_t] + b_f) \tag{5}$$

$$i_t = \sigma_g(W_i.[h_{t-1}, x_t] + b_i) \tag{6}$$

$$\widetilde{C} = tanh(W_c.[h_{t-1}, x_t] + b_c) \tag{7}$$

$$C_t = (f_t \times C_{t-1} + i_t \times \widetilde{C}) \tag{8}$$

$$o_t = \sigma_g(W_o.[h_{t-1}, x_t] + b_o) \tag{9}$$

$$h_t = o_t \times tanh(C_t) \tag{10}$$

Where the Eqs. (5), (6) and (9) represent the forgot, input and output gate operations receptively. These gates operations use sigmoid function σ_g to generate their output between 0 and 1. Where the value 0 means the gate is blocking the information and 1 means all information is passing through the gate. Where the parameters W and b are the weights and bias of the respective gates. h_{t-1} is the previous output at timestamp $t-1$ and x_t is the current input at time t. Whereas the Eqs. (7), (8) and (10) describes the cell states and hidden state of LSTM where C_t is memory state of cell and $\widetilde{C}$ cell state candidate at time t.

To learn the temporal dependencies between the sequence of video frames, the combined spatial feature vector X_s is then passed to LSTM units to learn global motion changes. The LSTM layer makes a final prediction by averaging label probabilities $Y_t' \in \theta$, for each pair of input frame sequence using a softmax classifier, where θ is a finite set of outcomes. The prediction distribution $P(Y_t)$ is defined as the following Eq. (11).

$$P(Y'_t = j|X_s) = softmax(Y'_t) = \frac{e^{Y'_t,\theta}}{\sum_{j=0}^{k} e^{Yr_t^j,\theta}} \tag{11}$$

Here $Y'_t = W_t h_t + b_h$ is the linear prediction layer, W (weight) and b (bias) are trainable parameters and t is current timestamp. The output of LSTM through a softmax layer classifies each frame into violent or non-violent categories. As the model is classifying each frame of the videos as a violent or non-violent frame, so to classify the whole videos, the prediction form each frame is accumulated and used as a threshold to classify at the video level. Figure 1 shows the proposed network model architecture.

3.3 Implementation Details

For the network training, the dataset is first divided into two categories of violent and non-violent videos, and equally spaced frames from each video are extracted and resized to 224×224 pixels. The number of frames in each training video is fixed to 30 frames. If a given video is longer than 30 frames then the intermediate frames are skipped at regular intervals to avoid redundant computation. All the training frames are normalized in the binary range $[0, 1]$. For spatial feature extraction pre-trained weights are used for VGG16 and only the additional residual layer blocks and the LSTM layers are trained on input data whereas the pre-trained model is not trained during the training phase. The residual layer block contains the stack of three convolution layers with a filter size of (1×1), (3×3) and (1×1) where each layer has total of 128, 64 and 128 filters respectively as shown in Fig. 1. The recurrent neural networks take more time to converge because of the sequential inputs and accumulation of gradients [30]. This causes the loss to propagate between the intermediate states, and because of this, the gradient starts to fluctuate and can result in exploding gradient problem, so gradient clipping method [30] is used to solve this problem. Also, dropout and regularization [39] is used between the LSTM cells. The batch size of 5 videos is selected and the shape of the batch data is (b, f, w, h, d) where b is batch size, f is number of frames in a video and w, h and d is the width, height, and depth of input respectively. The initial learning rate of 10^{-5} is used, and the network is trained for 50 epochs. The network is trained using ADAM optimizer to minimize the cross-entropy loss. The LSTM layer has a total number of 1024 units, and it outputs, through a softmax, one of the two classes of videos, i.e., violence or non-violence. As the network is trained using sequential pair of input frames of a video so the final prediction is made at the frame-level and the prediction at video level is calculated by accumulating frame-level predictions using a threshold value which gives maximum accuracy.

4 Experimental Results

In order to check the effectiveness of the proposed network to classify the fight and non-fight videos in different situations, it has been tested on three different

popular datasets: Hockey Fights, Crowd Violence and BEHAVE. The network is implemented in Python programming language with the tensorflow framework. The training and testing of the proposed model are done using Nvidia GTX Titan X graphics processing unit. The results of the proposed method are compared with several state-of-the-art violence detection techniques which are based on both hand-crafted features and deep neural network (DNN) based methods. Table 1 shows the testing accuracies of proposed methods and comparison with the existing methods.

Fig. 2. Different samples of violent and non-violent scenes from three datasets. Hockey Fight (1st row), Crowd violence (2nd row) and BEHAVE (3rd row). (a) shows sample frames which do not contain any violent activities whereas (b) corresponds to actions that contain violent actions.

4.1 Datasets

Three popular benchmark datasets are used for testing the effectiveness of proposed model. The first is Hockey Fight dataset [29] which contains violent activities from ice hockey matches of similar context. This dataset contains a total of 1000 videos with an equal number of fight and non-fight videos. The duration of each video is 50 frames. The second dataset "Violent-Flows" introduced in [18] is based on real-world crowd violent activities, e.g. fights in protests or in football matches. This dataset consists of a total of 246 videos each with a duration of 1 to 6 s. The third dataset BEHAVE [1] contains ten different activities performed by a group of people with a static background and no camera movements, similar to fixed CCTV surveillance footages. The activities include meeting in a group, running, fighting, chasing, following, and some other group activities. Each of the four clips has a frame size of 640×480 and have different length and contains multiple activities in a single video. However, for the purpose of violence detection, only the video portions which contain fight scenes are taken and labeled as "fights" whereas the rest of the clips are labeled as "non-fights". For the experiments, 20 clips involving fights and 50 clips without

Table 1. The accuracy evaluation of the proposed model and a comparison with the existing methods.

Method	Datasets		
	Hockey Fight	Crowd Violence	BEHAVE
ViF [18]	82.9 ± 0.14%	81.3 ± 0.21%	83.62 ± 0.19%
OViF [13]	87.5 ± 1.7%	88 ± 2.45%	-
MoSIFT+BoW [29]	88.8 ± 0.75%	83.42 ± 8.0%	81.65 ± 0.23 %
MoSIFT+HIK [29]	90.9%	-	-
Mohammadi et al. [26]	-	85.43 ± 0.21%	-
HOG+BoW [29]	88.77 ± 0.73%	57.43 ± 0.37	58.97 ± 0.34
MoSIFT+KDE+SC [38]	94.3 ± 1.68%	89.05 ± 3.26%	87.07 ± 0.13%
Gracia et al. [16]	82.4 ± 0.4%	-	-
Deniz et al. [10]	90.1 ± 0%	-	-
MoI-WLD [40]	96.8 ± 1.04%	93.19 ± 0.12%	88.83 ± 0.11%
LaSIFT+BoW [31]	94.42 ± 2.82%	93.12 ± 8.77%	-
AMDN [37]	89.7 ± 1.13%	84.72 ± 0.17%	84.22 ± 0.17%
Three streams+LSTM [12]	93.9%	-	-
ConvLSTM [34]	97.1 ± 0.55%	94.57 ± 2.34%	-
BiConvLSTM [17] (Spatiotemporal model)	97.9 ± 0.55%	96.32 ± 1.52%	-
Proposed Method	**98.8 ± 0.5%**	**97.3 ± 1.7%**	**94.8 ± 2.3%**

fighting scenes are used from each of the four clips, each with a duration of 2 to 5 s. Figure 2 shows some sample frames from three datasets used.

4.2 Results and Discussion

The 5-fold cross validation method is used to evaluate accuracy of the proposed algorithm where the dataset is divided into five equal size partitions, and four folds are used for training, and one is used for testing. For each test fold, maximum accuracies of each epoch are collected and overall model accuracy is obtained by calculating mean and standard deviation. The cross validation is used to tune the network hyper-parameters. The network model parameters are selected based on maximum accuracy on the validation dataset. Figure 3 shows the graph of testing performance accuracies on three different datasets. For each dataset, the network is run for a total of 50 epochs. In Fig. 3(a) the maximum performance accuracy of the proposed model on Hockey Fight dataset is 98.8%, which occurs first on the 18th epoch. In Fig. 3(b)the test accuracy on Crowd Violence dataset is 97.3% which is occurred first at 25th epoch whereas for the BEHAVE dataset, for which only fighting category is used in this proposed methods, gives the test accuracy of 94.8% as shown in Fig. 3(c). The proposed

method provides significant improvement in performance accuracies when compared to existing state-of-the-art methods. Table 1 lists the experimental results and shows extensive comparison with the existing techniques for the detection of violence and demonstrates the superiority of the proposed compared to existing state-of-the-art methods.

The use of feature fusion method and additional residual layer block have significantly improved the performance accuracy of the network architecture. The additional residual layers learn the features of both input frames, which contain combined high-level and combined low-level feature maps of two frames. The recurrent networks with the help of LSTMs then learn the temporal dependencies from combined feature map from two sequential frames whereas the pre-trained VGG16 model helps to extract spatial features and significantly reduce the training times effectively.

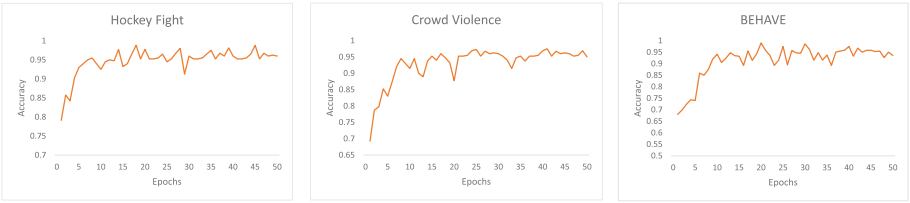

Fig. 3. Test accuracy graphs for three datasets. For (a) Hockey Fight (b) Crowd violence (c) BEHAVE.

5 Conclusion

This paper proposed a novel method for detecting the violent activities in videos by using multi-level spatial feature fusion instead of the optical flow for motion information. The spatial features fusion combined the multi-frame features to provide local and global motion changes pattern between the frames. Additional residual layer blocks, apart from the pre-trained network, that are used to learn these combined spatial features have significantly improved the performance. The LSTM layer block learned the temporal dependencies in the input frame sequence from fused spatial features of two input frames. The proposed method is tested on three standard datasets hockey fight, crowd violence and BEHAVE where the experimental results show significant improvement in performance compared to state-of-the-art violence detection methods

References

1. Blunsden, S., Fisher, R.: The behave video dataset: ground truthed video for multi-person behavior classification. Ann. BMVA **4**(1–12), 4 (2010)

2. Chen, M., Hauptmann, A.: MoSIFT: recognizing human actions in surveillance videos. Research showcase. Computer Science Department, School of Computer Science, Carnegie Mellon University (2009)
3. Cheng, W.H., Chu, W.T., Wu, J.L.: Semantic context detection based on hierarchical audio models. In: Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval, pp. 109–115. ACM (2003)
4. Cristani, M., Bicego, M., Murino, V.: Audio-visual event recognition in surveillance video sequences. IEEE Trans. Multimedia **9**(2), 257–267 (2007)
5. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Workshop on Statistical Learning in Computer Vision, ECCV, Prague, vol. 1, pp. 1–2 (2004)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: International Conference on Computer Vision & Pattern Recognition (CVPR 2005), vol. 1, pp. 886–893. IEEE Computer Society (2005)
7. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 428–441. Springer, Heidelberg (2006). https://doi.org/10.1007/11744047_33
8. Datta, A., Shah, M., Lobo, N.D.V.: Person-on-person violence detection in video data. In: Object Recognition Supported by User Interaction for Service Robots. vol. 1, pp. 433–438. IEEE (2002)
9. De Souza, F.D., Chavez, G.C., do Valle Jr, E.A., Araújo, A.D.A.: Violence detection in video using spatio-temporal features. In: 2010 23rd SIBGRAPI Conference on Graphics, Patterns and Images, pp. 224–230. IEEE (2010)
10. Deniz, O., Serrano, I., Bueno, G., Kim, T.K.: Fast violence detection in video. In: 2014 International Conference on Computer Vision Theory and Applications (VISAPP), vol. 2, pp. 478–485. IEEE (2014)
11. Donahue, J., et al.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2625–2634 (2015)
12. Dong, Z., Qin, J., Wang, Y.: Multi-stream deep networks for person to person violence detection in videos. In: Tan, T., Li, X., Chen, X., Zhou, J., Yang, J., Cheng, H. (eds.) CCPR 2016. CCIS, vol. 662, pp. 517–531. Springer, Singapore (2016). https://doi.org/10.1007/978-981-10-3002-4_43
13. Gao, Y., Liu, H., Sun, X., Wang, C., Liu, Y.: Violence detection using oriented violent flows. Image Vis. Comput. **48**, 37–41 (2016)
14. Giannakopoulos, T., Pikrakis, A., Theodoridis, S.: A multi-class audio classification method with respect to violent content in movies using Bayesian networks. In: 2007 IEEE 9th Workshop on Multimedia Signal Processing, pp. 90–93. IEEE (2007)
15. Giannakopoulos, T., Pikrakis, A., Theodoridis, S.: A multimodal approach to violence detection in video sharing sites. In: 2010 20th International Conference on Pattern Recognition, pp. 3244–3247. IEEE (2010)
16. Gracia, I.S., Suarez, O.D., Garcia, G.B., Kim, T.K.: Fast fight detection. PloS One **10**(4), e0120448 (2015)
17. Hanson, A., Pnvr, K., Krishnagopal, S., Davis, L.: Bidirectional convolutional LSTM for the detection of violence in videos. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)
18. Hassner, T., Itcher, Y., Kliper-Gross, O.: Violent flows: real-time detection of violent crowd behavior. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 1–6. IEEE (2012)

19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
20. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
21. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1725–1732 (2014)
22. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
23. Laptev, I.: On space-time interest points. Int. J. Comput. Vis. **64**(2–3), 107–123 (2005)
24. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
25. Medel, J.R., Savakis, A.: Anomaly detection in video using predictive convolutional long short-term memory networks. arXiv preprint. arXiv:1612.00390 (2016)
26. Mohammadi, S., Kiani, H., Perina, A., Murino, V.: Violence detection in crowded scenes using substantial derivative. In: 2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–6. IEEE (2015)
27. Mu, G., Cao, H., Jin, Q.: Violent scene detection using convolutional neural networks and deep audio features. In: Tan, T., Li, X., Chen, X., Zhou, J., Yang, J., Cheng, H. (eds.) CCPR 2016. CCIS, vol. 663, pp. 451–463. Springer, Singapore (2016). https://doi.org/10.1007/978-981-10-3005-5_37
28. Mustafa, H.T., Yang, J., Zareapoor, M.: Multi-scale convolutional neural network for multi-focus image fusion. Image Vis. Comput. **85**, 26–35 (2019)
29. Bermejo Nievas, E., Deniz Suarez, O., Bueno García, G., Sukthankar, R.: Violence detection in video using computer vision techniques. In: Real, P., Diaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W. (eds.) CAIP 2011. LNCS, vol. 6855, pp. 332–339. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23678-5_39
30. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: International Conference on Machine Learning, pp. 1310–1318 (2013)
31. Senst, T., Eiselein, V., Kuhn, A., Sikora, T.: Crowd violence detection using global motion-compensated lagrangian features and scale-sensitive video-level representation. IEEE Trans. Inf. Forensics Secur. **12**(12), 2945–2956 (2017)
32. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Systems, pp. 568–576 (2014)
33. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint. arXiv:1409.1556 (2014)
34. Sudhakaran, S., Lanz, O.: 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), pp. 1–6. IEEE (2017)
35. Sultani, W., Chen, C., Shah, M.: Real-world anomaly detection in surveillance videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6479–6488 (2018)
36. Xingjian, S., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.C.: Convolutional LSTM network: a machine learning approach for precipitation nowcasting. In: Advances in Neural Information Processing Systems, pp. 802–810 (2015)

37. Xu, D., Yan, Y., Ricci, E., Sebe, N.: Detecting anomalous events in videos by learning deep representations of appearance and motion. Comput. Vis. Image Underst. **156**, 117–127 (2017)
38. Xu, L., Gong, C., Yang, J., Wu, Q., Yao, L.: Violent video detection based on MoSIFT feature and sparse coding. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3538–3542. IEEE (2014)
39. Zaremba, W., Sutskever, I., Vinyals, O.: Recurrent neural network regularization. arXiv preprint. arXiv:1409.2329 (2014)
40. Zhang, T., Jia, W., He, X., Yang, J.: Discriminative dictionary learning with motion weber local descriptor for violence detection. IEEE Trans. Circuits Syst. Video Technol. **27**(3), 696–709 (2016)

Author Queries

Chapter 33

Query Refs.	Details Required	Author's response
AQ1	Per Springer style, both city and country names must be present in the affiliations. Accordingly, we have inserted the city name in affiliation. Please check and confirm if the inserted city name is correct. If not, please provide us with the correct city name.	