Not all temporal shift modules are profitable

Youshan Zhang,[a] Yong Li,[b,*] Shaozhe Guo,[a] and Qiming Liang[a]
[a]Engineering University of PAP, Graduate Student Brigade, Xi'an, China
[b]Engineering University of PAP, College of Information Engineering, Xi'an, China

Abstract. With the increasing coverage of video surveillance systems in modern society, demand for using artificial intelligence algorithm to replace humans in violent behavior recognition has also become stronger. By moving some channels in the temporal dimension, temporary shift module (TSM) can achieve the performance of three-dimensional convolution neural network (CNN) with the complexity of two-dimensional CNN, and extract the temporal and spatial information at the same time. Our intuition is that too many temporary shift modules may fuse too much action information in each frame, which weakens the capability of CNN on spatiotemporal information extraction. To verify the aforementioned conjecture, we adjusted the network structure based on TSM, proposed partial TSM, selected the optimal model through experiments, and verified the performance of the algorithm on multiple datasets and our expanded datasets. The proposed optimal model not only reduced the memory usage of hardware but also achieved higher accuracy on multiple datasets with 77.3% running time. Meanwhile, we achieved state-of-the-art performance of 91% on RWF-2000 dataset. [DOI: 10.1117/1.JEI.31.4.043030]

Keywords: violent behavior recognition; convolutional neural network; temporal shift module.

Paper 220200G received Feb. 23, 2022; accepted for publication Jul. 21, 2022; published online Aug. 5, 2022.

1 Introduction

With the increase of urban digitization year by year, more and more video monitoring equipment is applied in various public places, such as shopping malls, airports, schools, railway stations, and so on, which promotes the rapid development of the video monitoring system. In just over 20 years, the monitoring equipment in many major cities has basically covered large places and trunk roads, and the video monitoring technology has been continuously developed and improved. However, at present, most intelligent video monitoring systems can only provide video data acquisition and storage functions. If you want to analyze and understand the video monitoring content, you have to rely on manual monitoring, which is time-consuming and costly. There is no doubt that people will be exhausted after viewing for a long time. Therefore, it is rather difficult for them to process a large amount of video data effectively.

Deep learning plays a great role in the field of computer vision. Many studies[1–6] are devoted to extract and fuse spatiotemporal information quickly and effectively. Its application in behavior recognition has become an important method to replace manual video processing. According to the different backbone frameworks selected for recognition, behavior recognition mainly uses recurrent neural network, two-stream convolutional neural networks, three-dimensional (3D) convolutional neural networks, and transformer.

The main idea of two-stream convolutional neural networks is to extract spatial information and temporal information, respectively, through two convolutional neural networks, and then fuse them through appropriate information fusion method for action recognition. Simonyan and Zisserman[7] first proposed a two-stream convolutional neural networks for behavior recognition. Feichtenhofer et al.[8] first discussed different fusion strategies for spatial and temporal information in two-stream convolutional neural networks. Inspired by ResNet,[9] Feichtenhofer et al.[10] redesigned a lightweight two-stream convolutional neural networks, applied shortcut between

*Address all correspondence to Li Yong, liyong@nudt.edu.cn spatial branch and temporal branch, and used different frame rate to better focus on dynamic information.

The main idea of 3D convolutional neural networks is to extract temporal and spatial information from multiple adjacent frames at the same time by 3D convolution kernels. Due to dimensional constraints, two-dimensional (2D) convolution is difficult to extract temporal information from a single picture, Ji et al.[11] first applied the 3D convolution method to human behavior recognition. Tran et al.[12] proposed an approach that used deep 3D convolutional networks for spatio–temporal feature learning, which proved that using $3 * 3 * 3$ convolution kernel can achieve the best accuracy. Qiu et al.[13] combined the 2D spatial and one-dimensional temporal convolutions, which replaced the 2D residual module in residual neural network (ResNet) and achieved better results.

As we all know, the algorithms and ideas applied in the field of natural language processing are widely delivered to the field of computer vision. To complete the task of computer vision, the most important thing is how to extract spatial information and fuse it with temporal information between frames.

Recurrent neural network can extract the spatial information of frames and transmit the temporal information through hidden layers. Hochreiter and Schmidhuber[14] proposed a new network structure—long short-term memory (LSTM), which enables recurrent neural networks to retain feature information for a longer time in long sequence training, and solves the problem of narrowing sensing domain caused by gradient disappearance and gradient explosion to some extent. Donahue et al.[15] proposed a new network architecture long-term recurrent revolutionary networks (LRCNs), which connects convolutional neural network directly with LSTM.

Since vision transformer[16] broke the application barrier of transformer in the field of computer vision, a series of researches based on transformer have showed its potential in computer vision. The latest research results[17–22] showed that transformer's performance in computer vision tasks has exceeded or equal to that of convolution neural network (CNN).

Because of its structural characteristics, recurrent neural network has not achieved good results in the field of computer vision. Two-stream convolutional neural networks extract spatio–temporal information through two-way convolution, since the two will be fused for recognition, redundant information is extracted. The 3D CNN expands the dimension of convolution kernel, resulting in the exponential increase of its parameters and computation compared with 2D CNN. The framework based on transformer uses pure attention instead of convolution kernel, so its feature extraction efficiency is not as efficient as CNN framework, which means the framework based on transformer needs to pay expensive time cost on massive data. If we want to recognize the violent behavior in surveillance video in real time, we need a more lightweight and efficient method to extract spatio–temporal information. Lin et al.[23] proposed the temporal shift module (TSM). By shifting and splicing adjacent frames in the temporal dimension and using 2D convolution to extract spatio–temporal information at the same time, the performance of 3D convolution is achieved, and the problems of 3D convolution in parameters and calculations are solved.

It is worth mentioning that Ding et al.[24] proposed RepVGG, a simple architecture with a stack of 3×3 Conv and rectified linear unit. The running speed is much higher than ResNet-50 and ResNet-101, and higher accuracy is achieved at the same time. This architecture has no branches, which means every layer takes the output of its only preceding layer as input and feeds the output into its only following layer. This has something in common with the idea of this paper.

Our intuition is that some TSMs can fuse the information of adjacent frames on a single picture, but too many TSMs may fuse too much action information on each frame, which weakens the ability of CNN on spatio–temporal information extraction. To verify the above conjecture, we adjust the network structure based on TSM and propose partial TSM (P-TSM), experiments are carried out to explore whether to use temporary shift module in different stages of the network. The main contributions of this paper are as follows:

1. A P-TSM is proposed, which reduces the network complexity, protects the feature extraction capability of the backbone convolution network, and improves the accuracy of the algorithm.

2. Introduce the two-cascade TSM into partial TSM. The experiments show that the combination of single-cascade TSM and two-cascade TSM can not further improve the accuracy compared with the full version of the two-cascade TSM.
3. The performance of the algorithm is verified on the existing open source datasets and the expanded violent behavior recognition dataset we established.

2 Background

2.1 *Temporal Shift Module*

To better complete the task of behavior recognition, it is very important to effectively extract temporal and spatial information from continuous frames of video. The traditional 2D convolution uses 2D convolution kernel, which can only extract the spatial information of a single frame, but cannot extract the temporal and spatial information of multiple frames at the same time such as 3D convolution. TSM proposed by Lin et al.[23] have solved this problem in a novel way. By moving some channels along the temporal dimension, the spatial information between adjacent frames can be mixed. TSMs were inserted before 2D CNN, so the 2D convolution kernels can extract spatial information and temporal information at the same time. By this way, the performance of 3D CNN is achieved with 2D CNNs complexity.

As shown in Fig. 1(a), the original input tensor is stacked by several adjacent frames, and different colors represent frames at different times. As shown in Fig. 1(b), by moving some channels along the temporal dimension in the same input batch, 2D CNN can extract spatial and temporal information at the same time.

For better expression, we assume that the input is an infinite one-dimensional matrix X and a $1 * 3$ convolution kernel $W_1 = (a, b, c)$, then the convolution operation $Y = \mathrm{Conv}(W, X)$ can be written as

$$Y_i = a * X_{i-1} + b * X_i + c * X_{i+1}. \tag{1}$$

We shift the input X by -1, 0, $+1$, the shift operation can be written as

$$X_i^{-1} = X_{i-1}, \quad X_i^0 = X_i, \quad X_i^{+1} = X_{i+1}. \tag{2}$$

After that, the convolution block followed will complete the multiply accumulate operation:

$$Y = a * \sum X_{i-1} + b * \sum X_i + c * \sum X_{i+1}, \tag{3}$$

$$Y = a * X^{-1} + b * X^0 + c * X^{+1}. \tag{4}$$

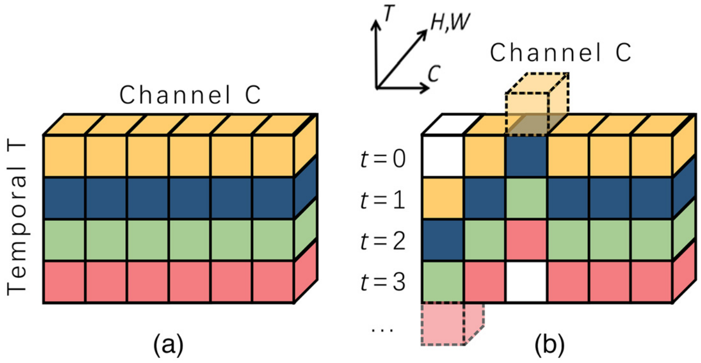


Fig. 1 (a) The original tensor without shift and (b) the tensor after shift.

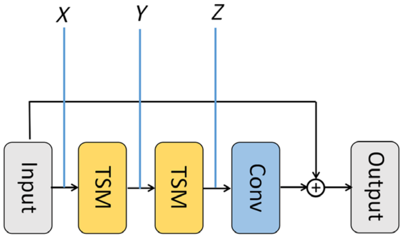


Fig. 2 Two-cascade temporal shift module.

2.2 *Two-cascade TSM Residual Module*

Based on the work above, Liang et al.[25] proposed that TSM network acquires limited long-term information during behavior recognition. And the network structure is so simple that over-fitting is prone to occur in the process of feature learning. To solve this problem, a two-cascade TSM is proposed as shown in Fig. 2, which expands the receptive field of temporal dimensions and enhances the long-term information extraction capacity.

On the result Y of the first TSM, carry out the second TSM, and set the $1 * 3$ convolution kernel of the second stage as $W_2 = (w1, w2, w3)$. A $1 * 5$ convolution effect can be achieved through two cascades, the output result Z is

$$Z = w_1 * Y^{-1} + w_2 * Y^0 + w_3 * Y^{+1}, \tag{5}$$

$$Z = w_a * X^{-2} + w_b * X^{-1} + w_c * X^0 + w_d * X^{+1} + w_e * X^{+2}, \tag{6}$$

and

$$w_a = w_1 * a, \tag{7}$$

$$w_b = w_1 * b + w_2 * a, \tag{8}$$

$$w_c = w_1 * c + w_2 * b + w_3 * a, \tag{9}$$

$$w_d = w_2 * c + w_3 * b, \tag{10}$$

$$w_e = w_3 * c. \tag{11}$$

3 Partial Temporal Shift Module

3.1 *Intuition*

We first explain the intuition behind P-TSM. The way CNN extracts spatio–temporal information is that the deeper the network is, the more specific shape the network can recognize. When the network is shallow, most of the extracted shape is points and lines. When the network reaches a deeper level, some specific objects or actions can be recognized. Excessively shift channels to adjacent frames will change the input tensor of the convolution layer, resulting in the decline of the overall spatial feature learning capability of the model. In the TSM model, ResNet-50 is used as the backbone and residual TSM is adopted through comparative experiments, the TSM is put inside the residual branch in a residual block. Similarly, the two-cascade TSM also uses ResNet-50 as the backbone, the same approach is adopted to insert TSM. To better explain the algorithm, we use a simple flowchart to represent ResNet-50 as shown in Fig. 3, in which TSM represents the temporary shift module, the dotted TSM represents the two-cascade TSM, Conv represents the convolution operation in each residual block, Avg represents the average pooling layer, and FC represents the full connection layer.

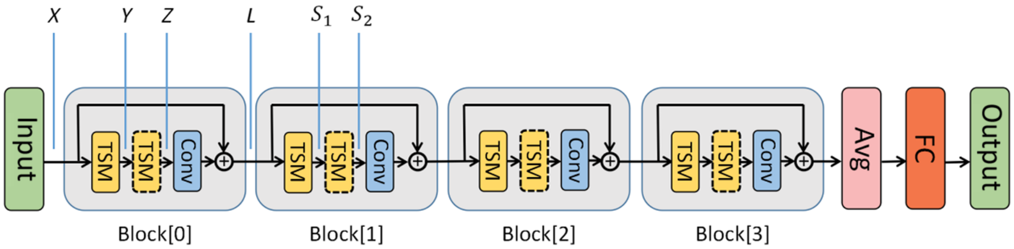


Fig. 3 A simple two-cascade TSM flowchart.

3.2 *Too Much Shifts Blur the Input Tensor*

This is like a video played at multiple speed. When playing at $2n + 1$ speed, you can see more single frames, with the problem of blurring the characteristic information of the current time. In the two-cascade TSM, the input of the second residual block can be written as

$$L = R_1(Z, X). \tag{12}$$

Among them, R_1 represents the convolution and shortcut of the first residual block in ResNet-50

$$S_1 = T_1(L), \tag{13}$$

$$S_2 = T_2(S_1), \tag{14}$$

$$S_2 = T_2(T_1(L)), \tag{15}$$

$$S_2 = w_a * L^{-2} + w_b * L^{-1} + w_c * L^0 + w_d * L^{+1} + w_e * L^{+2}. \tag{16}$$

Among them, T_1 and T_2 indicates two temporary shift modules in the second residual block. At this time, the input tensor of the second convolution layer has strode across nine frames. When the network becomes deeper, the input tensor of the convolution layer will contain more frames. After n TSMs, the input tensor will contain the information of $2n + 1$ frames, which is equivalent to expand the receptive field of temporal dimensions. However, this will lead to two serious problems: (1) weaken the CNNs capability of extracting spatial information at the current time and (2) the input tensor at the current time contains too much information of frames at other times, which means that the temporal information extracted by the subsequent convolution layers is useless for behavior recognition.

3.3 *Fewer Shift Performs Better*

Although the two-cascade TSM does expand the receptive field of temporal dimensions compared with TSM, we find that it is not worth using two-cascade TSM. Specifically, the two-cascade TSM gains 1% accuracy improvement with higher calculation cost, compared with the original TSM. If we only apply the temporary shift module on some blocks of ResNet-50, it will bring two significant advantages: (1) less data movement brings higher efficiency (less GPU occupancy and algorithm running time). Although the shift operation does not increase calculation cost, but it involves data movement which will increase the memory usage and inference latency on hardware. (2) The spatial modeling capability of 2D CNN backbone is protected. By reducing some temporary shift modules, some blocks take the output of its only preceding layer as input without any branches, so as to retain the information of all channels in the current frame. We observed that when some temporary shift modules were appropriately reduced, the accuracy increased to a certain extent, compared with 2D CNN baseline (TSM).

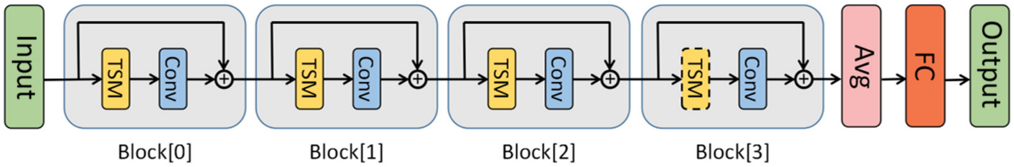


Fig. 4 Partial TSM with block [1,1,1,0].

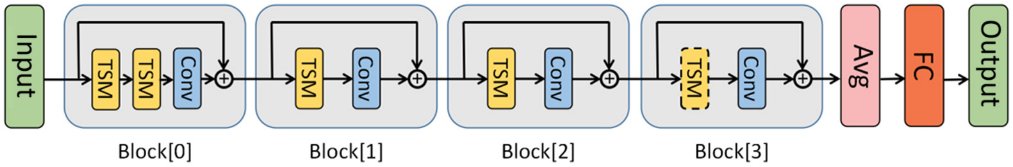


Fig. 5 Partial TSM with block [2,1,1,0].

3.4 *Module Design*

The temporal shift operation endows the 2D CNN with the ability to learn temporal features and harms the spatial modeling capability of the 2D CNN backbone. To balance the spatial and temporal feature learning ability of the algorithm, we first measured the accuracy of the single-cascade TSM, two-cascade TSM and model without inserting temporary shift module under the same experimental conditions, and discussed the impact of temporary shift module on the performance of the algorithm from the following two aspects.

3.4.1 *Frequency and positions of single temporary shift module's insertion*

To study the influence of the frequency and positions of single TSMs insertion on the performance of the algorithm, we used different combination strategies and measured the accuracy. We measure the model with ResNet-50 backbone and eight-frame input using 0–4 temporary shift modules with all possible insertion positions. The experiments show that the results of inserting temporary shift modules in three blocks are all better than those of the baseline regardless of the insertion positions.

As shown in Fig. 4, block [1,1,1,0] represents that when ResNet-50 is used as the backbone, the single temporary shift module is inserted into the first, second and third residual block, the fourth residual block remain unchanged. As shown in Table 2, the block combination of [1,1,1,0] achieves the best performance in the combination of frequency and positions of single temporary shift module insertion. Therefore, in the following experiments, we use this combination to further improve the performance of the algorithm.

3.4.2 *Combination of two-cascade and single temporary shift module*

Since the two-cascade TSM can expand the receptive field of temporal dimensions, we use the combination of the two-cascade and single TSM with all possible combinations. The experiments show that the proper combination of two-cascade and single TSM cannot better improve the performance of the model, which also confirms our conjecture.

As shown in Fig. 5, block [2,1,1,0] represents that when ResNet-50 is used as the backbone, the two-cascade temporary shift module is inserted into the first residual block, the single temporary shift module is inserted into the second and the third residual block, and the fourth residual block remain unchanged.

4 Experiments Design

We first show that P-TSM can further improve the performance of 2D CNN in violent behavior recognition. Then we further explore the potential of the combination of single-cascade and two-cascade TSM to prove our conjecture. Finally, we demonstrate the performance of our method on multiple datasets and show a lower calculation cost of our method compared with other optimal methods.

4.1 *Training and Testing Setups*

We carried out experiments on video violence recognition task. To be comparable with the two-cascade TSM, this paper refers to the hardware setting and deep-learning framework used in Ref. 25. During the whole experiments, the deep-learning framework used is pytorch1 5, the operating system is Ubuntu 16.04 and the CPU is Intel I9-10920x. Use CUDA10.2 to accelerate the GPU and use two NVIDIA RTX2080super GPU with 8 GB of video memory for parallel computing.

During training, we used the pre-trained model on Kinetics provided by Lin et al.[23] to reduce the computational complexity of network training. When training on RWF-2000 dataset, we also found that the training loss of the experiment decreased from the beginning of the training until it is stable, and the verification loss of the experiment also decreased continuously, however, when the training reaches about 30 epochs, the verification loss of the experiment will rise sharply. This indicates that over-fitting occurred during the experiments. Therefore, this paper also uses the learning rate adjustment method proposed by Liang et al.,[25] the initial learning rate is set to 0.001, and the learning rate is adjusted to 90% of the original every two epochs, which not only speeds up the adjustment speed of learning rate, but also accelerates the model learning process.

4.2 *Model*

To compare with the original TSM and the two-cascade TSM, we also use ResNet-50 as the backbone. The method we used to insert TSM is the same as Refs. 23 and 25.

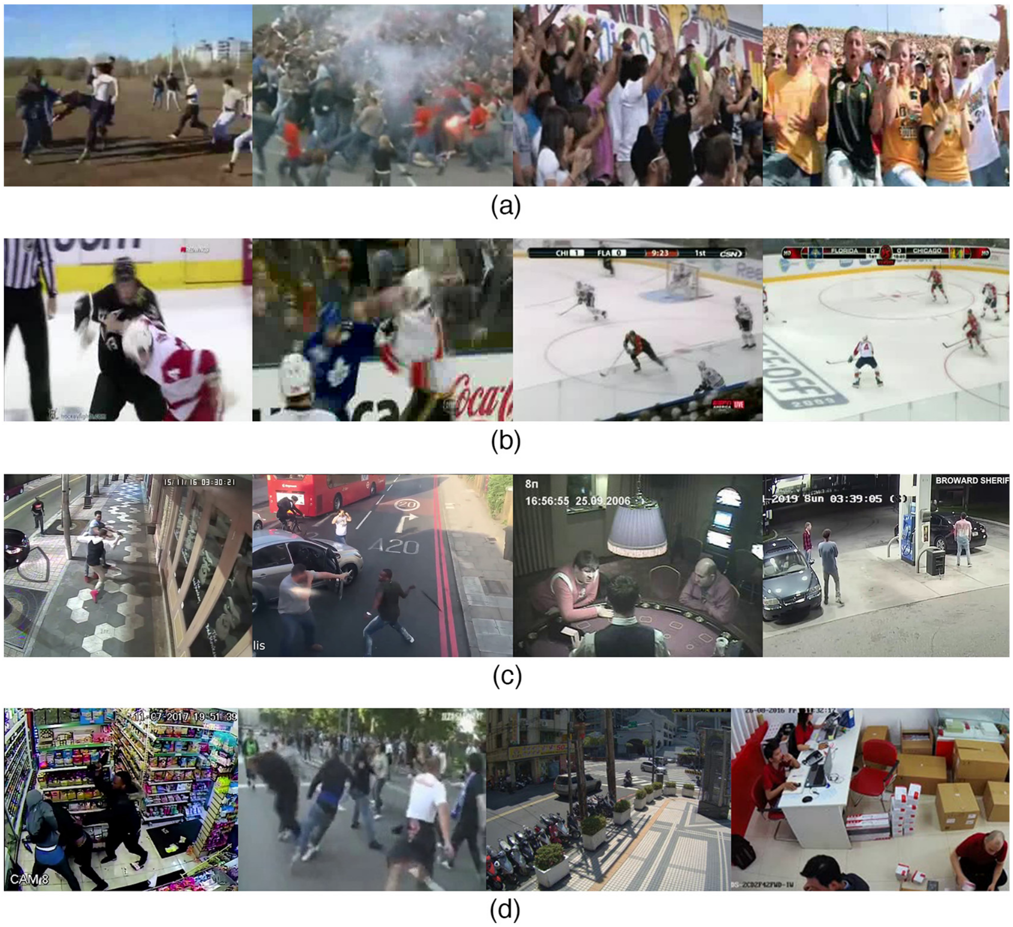
4.3 *Datasets*

To fully test the performance of the algorithm and verify our conjecture, experiments are carried out on the four datasets mentioned in Ref. 25. A summary of the used datasets is shown in Table 1.

The crowd violence dataset mainly contains the scenes of crowd, but due to long shooting distance and low resolution, most of the scenes are chaotic and vague. The hockey dataset contains 1000 violent and non-violent videos collected from ice hockey game. The training set includes 800 video clips, the verification set includes 100 video clips, and the test set includes 100 video clips. The latest published RWF-2000[26] dataset contains 2000 surveillance video clips collected from Youtube. The training set includes 1600 video clips, the verification set includes 200 video clips and the test set includes 200 video clips. Each video clip is 5 s and contains 150 frames. It mainly includes violent behaviors such as two persons, multiple persons, and crowds. The scenes are so rich and complicated that it is difficult to recognize. All video clips are obtained through the security camera. Without multimedia technology transformation, they fit the actual scene and have high research value. Figure 6 shows the basic situation of the dataset.

Table 1 Introduction of used datasets for violence recognition.

Dataset	Year of release	Clips include	Frames per second	Frame resolution
Crowd violence	2012	246	25	320 × 240
Hockey	2011	1000	25	360 × 288
RWF-2000	2021	2000	30	300 × 240, 320 × 240, 480 × 360, 920 × 720, 1280 × 720
Expended dataset	2021	5000	25,30	300 × 240, 320 × 240, 360 × 288, 480 × 360, 920 × 720, 1280 × 720


(a)

(b)

(c)

(d)

Fig. 6 Basic situation of dataset: (a) crowd violence dataset; (b) hockey dataset; (c) RWF-2000 dataset; and (d) expanded dataset.

Compared with other datasets with high utilization rate, the above three datasets are still too small, and there is still serious over-fitting phenomenon in the process experiments, which is not conducive to the application of violence recognition in real life. Therefore, our team expanded the dataset on the basis of the open source violence recognition dataset UCF-Crime, we collect hockey dataset, movies dataset, violent-flow dataset, HMDB51 dataset, and other scenes of violence in the video, and collect UCF101 and HMDB51 datasets as the main non-violent scenarios in the expanded dataset. Adobe Premiere Pro is used for editing. Because the duration of violence is always short. To better learn the characteristics of violent behavior, the video duration is uniformly edited to 1 or 5 s, which enriches the scene of RWF-2000 dataset, greatly increases the number of samples, solves the problem of over-fitting, and increases the universality of the dataset.

5 Results

Because the dataset is so small that the model become over-fitting between 10–30 epochs, and this phenomenon persists in the subsequent training process. We set the number of training rounds to 100. We will explain our experimental results from two aspects: single-cascade P-TSM and the combination of single-cascade and two-cascade P-TSM.

5.1 *Single-cascade P-TSM*

As shown in Table 2, we tried all insertion strategies for temporary shift modules in single-cascade P-TSM. And the accuracy of different block combination are showed. The number

Table 2 Different combination strategies of P-TSM compared with baseline.

Block	Accuracy	Accuracy comparison		
Combination		ResNet-50	TSM	Two-cascade TSM
[0,0,0,0]	84	0	–4	-5
[1,0,0,0]	83.5	–0.5	–4.5	–5.5
[0,1,0,0]	84	0	–4	–5
[0,0,1,0]	87.75	+3.75	–0.25	–1.25
[0,0,0,1]	86.75	+2.75	–1.25	–2.25
[1,1,0,0]	86.25	+2.25	–1.75	–2.75
[1,0,1,0]	87.75	+3.75	–0.25	–1.25
[1,0,0,1]	87	+3	–1	–2
[0,1,1,0]	87.75	+3.75	–0.25	–1.25
[0,1,0,1]	87.25	+3.25	–0.25	–1.75
[0,0,1,1]	86.25	+2.25	–1.75	–2.75
[1,1,1,0]	**91**	+7	+3	+2
[1,1,0,1]	88.75	+4.75	+0.75	–0.25
[1,0,1,1]	89	+5	+1	0
[0,1,1,1]	88.75	+4.75	+0.75	–0.25
[1,1,1,1]	88	+4	0	–1

Note: Bold value emphasizes optimal performance.

1 indicates that we have inserted single temporary shift module into the corresponding residual block and number 0 indicates that we keep the corresponding residual block unchanged.

We compared the results with related work. ResNet-50,[9] which equals to block [0,0,0,0], is used as the basic framework. TSM,[23] which equal to block [1,1,1,1], represents that TSMs are inserted in all blocks of the basic framework. Two-cascade TSM,[25] which equal to block [2,2,2,2], represents that TSMs are inserted twice in all blocks of the basic framework. We can see that most of the experiments using temporary shift modules have better performance than those without, which shows that even inserting one temporary shift module can expand the receptive field of temporal dimensions and enhance the long-term information extraction capacity. All P-TSM inserting three temporary shift modules have achieved better results than the single-cascade TSM, the best combination block [1,1,1,0] is 3% higher than original TSM and 2% higher than two-cascade TSM. It shows that using too many temporary shift modules in different blocks does weaken the backbone's capability of extracting spatial information. Properly reducing the temporary shift modules can not only reduce the complexity of the model but also improve the performance.

P-TSM with one or two temporary shift modules only achieves the best performance of 87.75%, which is even lower than that of original TSM. It shows that inserting a moderate number of temporary shift modules could better improve the model's capability of extracting spatio–temporal information, then improve the performance of the model.

5.2 *Combination of Single-cascade and Two-cascade P-TSM*

As shown in Table 3, we tried all insertion strategies for temporary shift modules in single-cascade and two-cascade P-TSM. The number 2 indicates that we have inserted two-cascade temporary shift module into the corresponding residual block.

Table 3 Different combination strategies of single-cascade and two-cascade TSM.

Block		Accuracy comparison
Combination	Accuracy	Two-cascade TSM
[2,1,1,0]	89.25	+0.25
[1,2,1,0]	88	–1
[1,1,2,0]	89.25	+0.25
[2,2,1,0]	88.25	-0.75
[2,1,2,0]	89	0
[1,2,2,0]	88.25	–0.75
[2,2,2,0]	88.75	–0.25

We can see that the combination of single-cascade and two-cascade P-TSM does not achieve excellent results, and each combination performs close to two-cascade TSM. We tried to train directly using the pre-trained model on Kinetics or the model we trained with the best results of single-cascade P-TSM (block [1,1,1,0]), the best accuracy we can achieve is 89.25%. We infer that there are two reasons for this phenomenon:

1. Too many temporary shift modules break the balance between spatial and temporal feature learning ability of the optimal model we already trained and
2. The combination of two-cascade and single TSM does not further improve the model's capability of extracting spatio–temporal information.

5.3 *Comparison of Optimal Methods*

To further verify the performance of the P-TSM, we conducted experiments on different datasets compared with other algorithms. Table 4 gives the specific situation of different algorithms on four violence recognition datasets.

As can be seen in Fig. 7 that the algorithm proposed in this paper further improve the accuracy compared with the original TSM and two-cascade TSM. In the crowd violence dataset, the P-TSM is 1% higher than the original TSM, which is equal to two-cascade TSM. In the hockey dataset, the P-TSM is 1% higher than the original TSM and 0.5% higher than the two-cascade TSM. In the RWF-2000 dataset, the P-TSM is 3% higher than the original TSM and 2% higher

Table 4 Comparison of optimal accuracy.

Algorithm	Crowd violence	Hockey	RWF-2000	Expanded dataset
3D-CNN[11]	94.3	94.4	82.75	91.7
LRCN[15]	94.57	97.1	77	92.3
I3D[27]	88.89	97.5	85.75	93.3
AR-Net[28]	95.918	97.2	87.3	92.8
TSM[23]	95.95	97.5	88	94.6
TEA[29]	96.939	97.7	88.5	93.8
Two-cascade TSM[25]	96.939	98.05	89	94.8
P-TSM (ours)	96.939	98.5	**91**	95.8

Note: Bold value emphasizes optimal performance.

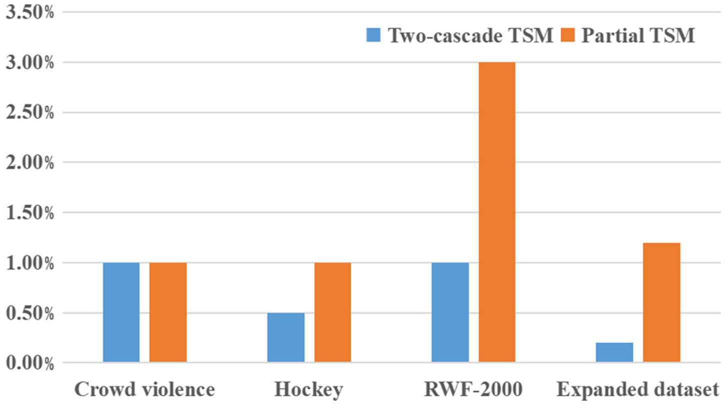


Fig. 7 Improvement of P-TSM and two-cascade TSM compared with original TSM.

Table 5 Comparison of computational cost.

Algorithm	Params (MB)	Estimated total size (MB)	Time cost
3D-CNN[11]	297.56	2647.70	4 h 8 m 23 s
LRCN[15]	237.83	1212.93	3 h 5 m 7 s
I3D[27]	46.88	1000.20	2 h 3 m 48 s
TSN[30]	89.69	390.05	46 m 41 s
TSM[23]	89.69	397.71	1 h 53 m 26 s
TEA[29]	91.95	479.78	3 h 10 m 48 s
Two-cascade TSM[25]	89.69	397.71	2 h 8 min 20 s
P-TSM (ours)	89.69	396.57	1 h 27 min 38 s

than the two-cascade TSM. In the expanded dataset, the P-TSM is 1.2% higher than the original TSM and 1% higher than the two-cascade TSM.

To better show the improvement of our method. We make a comparison of the computational cost of the proposed method with existing methods which is showed in Table 5. Torchsummary is used to calculate models' parameters and estimated total size. For better comparison, the input of summary function is set to fixed value. Batch size is set as 8. Picture size is $224 * 224$. The number of input channels is 3. Since torchsummary cannot deal with LSTM which is a part of LRCN, torchinfo is used to make relevant calculations of LRCN. Tensorboard is used to record relevant data which provides the time used after training and testing for 100 epochs.

TSN is the basic of TSM, TEA, Two-cascade TSM and our method. We can see that the algorithms based on TSN all have less parameters and model size. Since TSM only increases the memory usage of hardware rather than model size and parameters. We compare the performance of the algorithm by time cost. After training and testing 100 epochs in the RWF-2000 dataset, P-TSM takes 1 h 27 min and 38 s comparing with 1 h 53 min and 26 s taken by the original TSM. Our algorithm improves the running speed by about 23%. With far more less TSMs inserted, our proposed method greatly reduce the training time cost compared with two-cascade TSM.

6 Conclusion

To better recognize violent behavior in surveillance video, this paper improves the algorithm based on original TSM and two-cascade TSM. Our conjecture is that not all temporary shift modules can improve the performance of the algorithm. This paper proposes a P-TSM, some relevant experiments are done to prove that using the appropriate number of temporary shift modules can better balance the spatial and temporal learning capability of the algorithm. The proposed optimal model not only reduces the memory usage of hardware, but also achieves higher accuracy on multiple datasets with higher running speed. We also achieved state-of-the-art performance of 91% on RWF-2000 dataset.

Acknowledgments

This work was supported by the National Educational Science 13th Five-year Plan Project (Grant No. JYKYB2019012), the Basic Research Fund for the Engineering University of PAP (Grant No. WJY201907) and the Basic Research Fund of the Engineering University of PAP (Grant No. WJY202120).

References

1. S. Kaur, P. Kumar, and P. Kumaraguru, "Deepfakes: temporal sequential analysis to detect face-swapped video clips using convolutional long short-term memory," *J. Electron. Imaging* **29**(3), 033013 (2020).
2. T. Han et al., "Feature and spatial relationship coding capsule network," *J. Electron. Imaging* **29**(2), 023004 (2020).
3. X. Zhang et al., "Multimodal polarization image simulated crater detection," *J. Electron. Imaging* **29**(2), 023027 (2020).
4. J. Yan et al., "No-reference remote sensing image quality assessment based on gradient-weighted natural scene statistics in spatial domain," *J. Electron. Imaging* **28**(1), 013033 (2019).
5. T. Dai et al., "Research on recognition of painted faces," *J. Electron. Imaging* **31**(1), 013005 (2022).
6. S. Chen, W. Ma, and L. Zhang, "Dual-bottleneck feature pyramid network for multiscale object detection," *J. Electron. Imaging* **31**(1), 013009 (2022).
7. K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Adv. in Neural Inf. Process. Syst.*, Vol. 27 (2014).
8. C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, pp. 1933–1941 (2016).
9. K. He et al., "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. and Pattern Recognit.*, pp. 770–778 (2016).
10. C. Feichtenhofer et al., "Slowfast networks for video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pp. 6202–6211 (2019).
11. S. Ji et al., "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(1), 221–231 (2013).
12. D. Tran et al., "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 4489–4497 (2015).
13. Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 5533–5541 (2017).
14. S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.* **9**(8), 1735–1780 (1997).
15. J. Donahue et al., "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, pp. 2625–2634 (2015).
16. A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv:2010.11929 (2020).
17. H. Chefer, S. Gur, and L. Wolf, "Transformer interpretability beyond attention visualization," in *Proc. IEEE/CVF Conf. Comput. Vis. and Pattern Recognit.*, pp. 782–791 (2021).

18. Z. Dai et al., “UP-DETR: unsupervised pre-training for object detection with transformers,” in *Proc. IEEE/CVF Conf. Comput. Vis. and Pattern Recognit.*, pp. 1601–1610 (2021).
19. H. Chefer, S. Gur, and L. Wolf, “Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers,” arXiv:2103.15679 (2021).
20. W. Wang et al., “Pyramid vision transformer: a versatile backbone for dense prediction without convolutions,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pp. 568–578 (2021).
21. B. Heo et al., “Rethinking spatial dimensions of vision transformers,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pp. 11936–11945 (2021).
22. S. He et al., “TransReID: transformer-based object re-identification,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pp. 15013–15022 (2021).
23. J. Lin, C. Gan, and S. Han, “TSM: temporal shift module for efficient video understanding,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pp. 7083–7093 (2019).
24. X. Ding et al., “RepVGG: Making VGG-style convNets great again,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. and Pattern Recognit.*, pp. 13733–13742 (2021).
25. Q. Liang et al., “Violence behavior recognition of two-cascade temporal shift module with attention mechanism,” *J. Electron. Imaging* **30**(4), 043009 (2021).
26. M. Cheng, K. Cai, and M. Li, “RWF-2000: an open large scale video database for violence detection,” in *2020 25th Int. Conf. Pattern Recognit.*, IEEE, pp. 4183–4190 (2021).
27. J. Carreira and A. Zisserman, “Quo vadis, action recognition? a new model and the kinetics dataset,” in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, pp. 6299–6308 (2017).
28. Y. Meng et al., “AR-Net: adaptive frame resolution for efficient action recognition,” *Lect. Notes Comput. Sci.* **12352**, 86–104 (2020).
29. Y. Li et al., “Tea: temporal excitation and aggregation for action recognition,” in *Proc. IEEE/CVF Conf. Comput. Vis. and Pattern Recognit.*, pp. 909–918 (2020).
30. L. Wang et al., “Temporal segment networks for action recognition in videos,” *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(11), 2740–2755 (2019).

Youshan Zhang is an MS degree candidate at the Engineering University of PAP (EUPAP). His research interests include behavior recognition.

Yong Li is an associate professor at the EUPAP. His research interests include pattern recognition.

Shaozhe Guo is an MS degree candidate at the Engineering University of PAP (EUPAP). His research interests include object detection.

Qiming Liang is an MS degree candidate at the Engineering University of PAP (EUPAP). His research interests include behavior recognition.