A Spatio-Temporal Attention Convolution Block for Action Recognition

Junjie Wang1, Xueyan Wen1a*

1Information Engineering and computer school, Northeast Forestry University, Haerbin, HeilongJiang, 150036, China
*Corresponding author’s e-mail: 635108363@qq.com

Abstract. We propose a simple and effective 3D neural network module (STAT) embedded in spatiotemporal attention for action recognition. For a given intermediate feature map, our module sequentially infers the distribution of attention along the two dimensions of space and time, and multiplies it with the current feature map in the form of residual to achieve adaptive generation of the next stage feature map. STAT is a 3D convolution general module combined with attention. It is compatible with any 3D convolution network and can easily replace the 3D convolution kernel. The additional overhead it generates is negligible, and it can be trained end-to-end together with ordinary 3DCNN. By comparing the performance of the currently popular 3D networks on the UCF101 and HMDB51 datasets, experiments show that STAT has certain improvements on most 3D networks, which proves that STAT has a certain universality.

1. Introduction
The emergence of the convolutional neural network represented by AlexNet [1] indicates that deep learning brings a revolution to complex AI tasks. Among them, human action recognition is challenging and very interesting work. Although the network ResNet-152 [2] has been successful in 2D images, it's hard to deal with data with temporal relationships, such as videos. But the time series on video is the key issue for video analysis. In the video, the target character is unique in space and time, even in the same space and time period. It’s natural for people to regard video as a time series of spatial problems, simultaneously modeling of time and space using 3D CNN, and an effective method for action recognition is obtained[3][4][5]. Some work like (I3D [4]) has achieved better results in motion recognition, compared with the traditional method (IDT [6]), its improvement is not as good as that of image recognition. According to the network architecture proposed by C3D[7], stacking the 3D convolution layer is more and more common. Although 3DCNN Networks can model video spatiotemporal better than the traditional methods, its time complexity and flexibility are cursed. Video usually has a complex background and is accompanied by the interaction between people and other objects, which has a lot of redundancy in space-time, which will affect the differential representation of the two states by the model. The human visual system[8] is rather sensitive to changes in time and space, and this gift allows humans to observe other people's movements at certain times. Therefore, it is a wise choice to introduce the attention mechanism into the model of video action recognition.

In this paper, we introduce a Spatial-Temporal Attention (STAT) module to apply the attention machine to the video action recognition model, the test case is shown in Figure 1. STAT models the
Figure 1. Time and space focus on the pole vault. From this picture we can see that spatial attention has been tracking the movements of the task, while temporal attention is shown in the middle two rows with a smaller range of movements, thus giving less attention than the other two rows.

video in space and time respectively. STAT only appears in a part of 3D convolution, it can replace some 3D convolution functions or insert into the 3DCNN model, which can not only capture time and motion information but also increase the flexibility of the model without increasing the complexity of the model. Finally, we improve the performance of most current 3D models with fewer parameters and less time.

2. Related work

With the development of computer science, human beings have accumulated a large amount of data in life and production. After great success in two-dimensional images, people have shifted their focus to video analysis. The core of video analysis is human action recognition. The accuracy and rapidity of action recognition will directly affect the follow-up results of the video analysis system, and most of the research on video analysis focuses on spatial and temporal modeling. Before the rise of deep learning, most video classifications were based on features designed by hand and typical machine learning methods. Some proposed video representations include SIFT-3D [9], spatiotemporal interest points (STIPs)[10] and Cuboids [11]. However, the most typical one is the improved Dense Trajectory (IDT), which is widely regarded as the most advanced [6]. It shows a powerful effect in video classification. After coding, it is input into the classification function to get the final classification category.

The emergence of deep learning pushes video classification to a new stage. Convolution neural network has achieved great success in image processing and has become a popular method for image processing[12-14]. Ji et al.[15] proposed to extract the temporal and spatial characteristics of video data through 3D convolution core, capture the motion information of the video stream, and finally combine the generated channel information to obtain the final feature description. Karpathy et al.[16] Studied the fusion of time information and spatial information and hoped that the time characteristic could enhance the connectivity between convolutional layers and calculate the activation amount through spatiotemporal convolution. However, the result was that the network performance of processing a single frame was similar to that of the whole spatiotemporal video volume. In the field of image classification, the training model obtained by ImageNet can be well applied to other fields and tasks. This has inspired J. Carreira et al.[4] to make a large-scale video data set, on which the trained models can perform better on a small set (HMD51[17], UCF101[18], etc.). After analyzing the most advanced dual-stream network and 2D convolution+LSTM network at that time [19-22], a dual-stream model based on 3D convolution was designed. However, after pre-training on Kinetics data sets and migration to HMDB51 and UCF101, the effect was unexpectedly wonderful. However, it seems that the pooling or RNN operation added on the 2D network cannot well extract the connection between the low-level information of the video. Therefore, some R2+1D[23-25] structures are proposed to obtaining additional
nonlinearity. In this work, we pay more attention to the flexibility and lightweight of the network, using (R2 + 1D) as our backbone network, even if it can be applied in other action recognition models.

![Diagram](image_url)

Figure 2. The structure of Spatial(up) and Temporal(down) Attention Block

With the development of attention mechanisms, the class of deep learning has been promoted to a higher level. The combination of video and attention is based on RNN structure[26-28], using only spatial or temporal attention structures[29,30], or Spatio-Temporal attention[22,31]. Inspired by the classical non-local means in computer vision, Xiaolong Wang et al.[32] proposed a non-local operation to capture the long-distance dependence and establish the connection between two pixels with a certain distance on the image and the connection between two frames in the video. It has successfully applied self-attention mechanisms to data in video structures and can be inserted into many computer vision structures, which is undoubtedly a successful start. Compared with the high complexity of non-local, Woo S, et al[33] proposed a lightweight universal space-channel attention mechanism. With the addition of the CBAM module, the model has better performance and interpretation and pays more attention to the target object itself.

Our work focuses on embedding the spatial-temporal attention mechanism into a flexible R2+1D network. On the one hand, it aims to model space and time more accurately and extract more useful information. The other is to increase the flexibility of the network, reduce the complexity of the 3D network, and reduce the network operation time.

3. Spatial-Temporal Attention block

Suppose there is an intermediate feature map $F \in \mathbb{R}^{C \times T \times W \times H}$, where $C, T, H, W$ denote the channel number, the frame number of the input video clip, the height and width of the frame-level feature map respectively. Take $F$ as the input to SATA, compared with R2 + 1D structure, STAT contains a 2D spatial attention map $F_s \in \mathbb{R}^{C \times W \times H}$ and a 1D temporal attention mechanism $F_t \in \mathbb{R}^{C \times T}$, as illustrated in Fig.1. It should be noted that for the previous 3D or dual-stream networks, the feature map of each channel is obtained through 3D convolution kernel or 2D convolution incorporating time perception. Therefore, it captures the combined information between the target object and the environment. CBAM[33] focuses on the channel and spatial information and ignores the time information. The objective of STAT is to integrate attention into the network module to refine the feature map of different stages, as Figure 2 shows. In terms of space, the model is expected to focus more on the target and the interaction between the target and the environment, and for the time, it is expected to focus more on the moment when the behavior occurs. The manifestation of attention function is as follows:
\[ M = G(F) \], \[ F' = M \otimes F \]

Where \( G \) denotes attention machine, and specifies the element-wise multiplication operation.

To integrate attention into convolution operation and promote effective and efficient learning, we consider a convolution module embedded in the attention mechanism, which integrates the separated convolution and attention mechanism. Therefore, the process of redefining the above feature attendance into a two-step sequence is as follows:

\[ M = f_s(F(W, H)), F' = f_s(M) \otimes F(W, H) \]
\[ N = f_t(F'(T)), F'' = f_t(N) \otimes F'(T) \]

Where \( f_s \) and \( f_t \) denote the spatial and temporal attention, respectively. The function \( f \) represents the spatial and temporal pooled connection operation. The details of each attention module are described below.

3.1. Spatial attention block

Spatial attention pays more attention to the target objects and the interaction between them in the video. It tries to locate them over time and generate the spatial attention feature map by using the spatial relationship between features. To calculate spatial attention, we first apply the average pooling and maximum pooling operations on the channel axis, denoted as \( \mathbf{d}_{s_{avg}} \in \mathbb{R}^{C \times H \times W} \) and \( \mathbf{d}_{s_{max}} \in \mathbb{R}^{C \times H \times W} \), which cannot obtain all the information on the feature map. Inspired by the residual block, we introduce the information of the original feature map and connect them to generate an effective feature descriptor. In this way, the information area can be highlighted effectively, but the features ignored by attention will not be lost. Then concatenate the three maps, deploy a spatial attention network \( f_{s_{feature}} \) the feature map throughout the convolution process, and set up a three-dimensional convolution layer to output features-level spatial attention. \( M_{s_{feature}} \). In this way, the characteristics of average collection and maximum collection in the channel are gathered together, and then they are connected and convolved through the standard convolutional layer to generate our 3D spatial attention diagram. In short, spatial attention can be described as follows:

\[ M_{s_{feature}} = F \otimes \sigma(f_{s_{feature}}^\text{kernel} ([\text{Avg}(F); \text{Max}(F); F])) \]
\[ = F \otimes \sigma(f_{s_{feature}}^\text{kernel} ([\mathbf{d}_{s_{avg}}, \mathbf{d}_{s_{max}}, F])) \in \mathbb{R}^{C \times d_x \times d_y \times d_z} \]

where \( \sigma \) denotes the sigmoid function and the kernel size can be set to \( 3 \times 3 \times 3 \).

3.2. Temporal attention block

Temporal attention focuses on the "when" aspect of video attention. Specifically, it measures the motion pattern of the target object in a particular way over time in a video sequence. Each channel of the feature map is an important feature detector [33] and contains the meaning of a given image. In the entire convolution process, the number of channels of the intermediate feature map is always increased to obtain more abundant semantics. Although increasing the number of channels can extract image information more finely, the model often produces an overfitting phenomenon. A high number of channels may lead to a large amount of redundant information, which affects the extraction of favorable information from the model.

To facilitate computation, we compress the spatial dimension \( W \times H \) to design a compact temporal descriptor \( \mathbf{d}_t \in \mathbb{R}^{C \times d_t} \). Although the average pooling has a strong ability to aggregate the global space, we still add the maximum pooling to prevent missing smaller or occluded targets. Max pool pays more attention to important local features, while average pooling pays more attention to global features. We
combine maximum pooling and average pooling into composite pooling. The combined feature map can be obtained by the following formula:

\[
P_{\text{avg}\_\text{max}} = m * P_{\text{max}}(F) + n * P_{\text{avg}}(F) = m + n = 1, 0 < m < 1, 0 < n < 1
\]

where \(m, n\) are parameter factors. Generally, we will take \(m=n\). This way can enrich the feature layer and is a pooling strategy extended for the advantages of the two pooling methods. Besides, we also introduce overlapping pooling to pool the information of adjacent pixels in the process of pooling output features, to obtain the key feature information lost by non-overlapping again. Firstly, we aggregate the spatial information of feature map by using average pooling, max-pooling operations to generate two different spatial context descriptors: \(f'_{\text{max}}, f'_{\text{avg}}\), which represent the average aggregation feature and the maximum aggregation feature, respectively. After they are calculated and combined on the feature map \(d'_{\text{max}}\) and \(d'_{\text{avg}}\) we forward them into a shared MLP network \(f'_{\text{avg}\_\text{max}}\) to produce the combine attention feature map. Then, the generated feature map will be input into the function \(f'_{\text{over}}\), that is, Overlapping Pooling. Finally, through a single layer MLP neural network, we get the final temporal attention feature map \(M_{t\_\text{feature}}\). The process is described below:

\[
M_{\text{feature}} = \text{MLP}\left(P_{\text{avg}\_\text{max}}(f'_{\text{max}}(F), f'_{\text{avg}}(F))\right) = f'_{\text{avg}\_\text{max}}(d'_{\text{avg}} \oplus d'_{\text{max}}) \in \mathbb{R}^{C \times d}
\]

\[
M_{t\_\text{feature}} = \sigma(\text{MLP}(f'_{\text{over}}(M_{\text{feature}})))
\]

3.3. Attention module settings

For a specific input image, spatial and temporal attention from two perspectives complement each other to focus on “what” and “when”. Because it can replace some units in the network, the two modules can be used as the minimum unit or embedded in the residual block. We find that the algorithm is embedded in the residual block and the result is better when the channel changes in the convolution process. We will discuss the details in the next section.

4. Experiments

We utilized two popular fine-grained action recognition benchmarks: HMDB51 and UCF101. HMDB51 contains 6766 videos with 51 categories. Each action contains at least 51 videos with a resolution of 320 * 240. It mainly includes general facial movements, body movements, and human-object interaction actions. UCF101 contains 101 types of actions, each of which is performed by 25 people. There are 13320 videos with a resolution of 320 * 240. It mainly includes five kinds of actions: human-object interaction, body movement, human to human interaction, playing music equipment, and various sports.

A common practice we follow is as follows [23]. We divided the data into a training set, validation set, and test set by 7:1:2 or use cross-validation. Each clip center in the video is cropped to a size of 112x112. To be fair, all models use the same settings. We take Top1 / 5, time consumption, and model size as performance evaluation.

| Table 1. The accuracy for different forms of 3Dmodel on the test set. |
|----------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|               | HMDB51       |              |              | UCF101       |              |              |              |              |
|               | Size   | Time   | Top1       | Top5       | Size   | Time  | Top1       | Top5       |
| C3D[7]        | 78M    | 4.3h   | 25.10      | 58.23      | 78M    | 10.9h | 51.63      | 78.64      |
| (2+1)D_base[23] | 33M    | 7.3h   | 24.97      | 54.18      | 33M    | 19.3h | 63.60      | 85.92      |
| (2+1)D unit   | 88M    | 10.9h  | 26.10      | 55.60      | 88M    | 20.2h | 63.50      | 83.76      |
| (2+1)D Spatial | 32M    | 5.8h   | 27.32      | 56.20      | 32M    | 16.4h | 63.70      | 83.80      |
| (2+1)D Tempo  | 32M    | 6.2h   | 26.94      | 52.48      | 32M    | 17.5h | 62.04      | 86.17      |
| STAT(ours)    | 33M    | 6.4h   | 28.90      | 60.00      | 33M    | 18.3h | 65.70      | 87.44      |
The purpose of our experiment is to verify whether STAT can improve the fitting ability and generalization ability of the model itself. The experiment was carried out in a three-dimensional network. We will discuss STAT as the smallest unit, insert remaining modules and compare with baseline, and discuss the role of adding spatial and temporal attention separately. We choose some pure 3D networks [7][23] [2] [4][34], which are very popular in recent years to verify.

For [7] and [23], we do not use the pre-training model, and divide the data into a training set, verification set, and test set, and replace the 3D convolution module with STAT in the residual block. Compared with (2 + 1)d replacing STAT(unit) and R2plus1D(base). Of course, we also set up two sets of experiments: spatial attention and temporal attention. The results are shown in Table 1.

Table 2. Action recognition accuracy for different 3D models on the dataset.

|               | HMDB51  | UCF101 |
|---------------|---------|--------|
| P3D[23]       | 88.6    | 76.8   |
| P3D_STAT      | 89.7    | 76.2   |
| I3D[4]        | 97.8    | 80.9   |
| I3D_STAT      | 98.2    | 81.2   |
| S3D[34]       | 96.8    | 75.9   |
| S3D_STAT      | 96.9    | 76.4   |

From Table 1 and figure 2, we can see that the introduction of temporal attention (2+1) D_Tempo and spatial attention (2+1) D_Spatial can promote the results, but they have different performance on two different data sets. They perform worse than introducing them together, which may be that replacing them in the form of (2+1)D performs better than in the residual block. Under the same conditions,
replacing the convolutional block in the residual attention module with STAT has better performance on top1 and top5, and there is only a slight overhead in model size and time.

For [2], [4], [34], we remove the last convolution block of each model and replace it with stat. at the same time, we cut the center of each frame in the video by 224x224, and use triple cross-validation to train. The results are shown in Table 2.

Table 2 shows us the effectiveness of the introduction of STAT. These complex 3D networks have strong information extraction and characterization capabilities. However, the design of the network itself is very complicated and prone to overfitting. STAT has introduced different pooling operations many times, which reduces the complexity of the model to a certain extent and also ensures the relative integrity of the information, which proves that STAT is effective to a certain extent.

5. Conclusion

We propose an attentional convolution module (STAT) to enhance the 3DCNN network representation. Attention-based feature refinement is introduced in two different modules of time and space, which improves the generalization ability of 3D models while reducing network overhead. In terms of the time dimension, we suggest the use of maximum pooling, average pooling, and overlapping pooling, to know the occurrence of the action, and finally, use of spatial attention to further promote the performance. Our proposed STAT learning "what" and "where" effectively highlights or suppresses intermediate features. To verify its effectiveness, we conducted extensive experiments using various advanced models to prove that the STAT model improves the performance of two different baseline datasets (UCF101 and HMDB51). We hope STAT can become an important part of 3D network architecture.

References

[1] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, 2012.
[2] Z. Qiu, T. Yao, and T. Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In ICCV, 2017.
[3] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional two-stream network fusion for video action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1933–1941, 2016.
[4] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In CVPR, 2017.
[5] G. Varol, I. Laptev, and C. Schmid. Long-term temporal convolutions for action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.
[6] H. Wang and C. Schmid. Action recognition with improved trajectories. In ICCV, 2013.
[7] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional net-works. In ICCV, 2015.
[8] Rensink, R.A.: The dynamic representation of scenes. Visual Cognition 1(1–3), 17–42, 2000.
[9] P. Scovanner, S. Ali, and M. Shah. A 3-dimensional sift descriptor and its application to action recognition. In ACM MM, 2007.
[10] I. Laptev and T. Lindeberg. Space-time interest points. In ICCV, 2003. 2.
[11] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In Proc.ICCV VS-PETS, 2005.
[12] L. Chang, X. M. Deng, M. Q. Zhou, Z. K. Wu, Y. Y uan, and S. Yang, “Convolutional neural networks in image understanding,” Acta Autom.Sinica, vol. 42, no. 9, pp. 1300–1312, 2016.
[13] G. Cheron, I. Laptev, and C. Schmid, “P-CNN: Pose-based CNN features for action recognition,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 3218–3226.
[14] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, “Learning and transferring mid-level image representations using convolutional neural networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 1717–1724.
[15] S. Ji, W. Xu, M. Yang, and K. Yu, “3D convolutional neural networks for human action recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 1, pp. 221–231, Jan. 2013.

[16] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In CVPR, 2014.

[17] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In ICCV, 2011.

[18] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human action classes from videos in the wild. In CRCV-TR-12-01, 2012.

[19] Tran, D., Wang, H., Torresani, L., Feiszli, M.: Video classification with channel-separated convolutional networks. In: IEEE International Conference on Computer Vision (2019)

[20] G. Varol, I. Laptev, and C. Schmid. Long-term temporal convolutions for action recognition. IEEE transactions on pattern analysis and machine intelligence, 2017.

[21] Fan, Q., Chen, C.F.R., Kuehne, H., Pistoia, M., Cox, D.: More is less: Learning efficient video representations by big-little network and depthwise temporal aggregation. NIPS.2019

[22] Du, W., Wang, Y., Qiao, Y.: Recurrent spatial-temporal attention network for action recognition in videos. T-IP 27(3), 1347–1360.2017.

[23] Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition.CVPR.2018.

[24] Li, Z., Gavrilyuk, K., Gavves, E., Jain, M., Snoek, C.G.: Videolstm convolves,attends and flows for action recognition. CVIU 166, 41–50.2018.

[25] Meng, L., Zhao, B., Chang, B., Huang, G., Sun, W., Tung, F., Sigal, L.: Interpretable spatio temporal attention for video action recognition. In: IEEE International Conference on Computer Vision Workshop. pp. 0–0 (2019).

[26] Wang, Y., Wang, S., Tang, J., O’Hare, N., Chang, Y., Li, B.: Hierarchical attention network for action recognition in videos. arXiv preprint arXiv:1607.06416 (2016).

[27] Girdhar, R., Ramanan, D.: Attentional pooling for action recognition. In: Advances on Neural Information Processing Systems. pp. 34–45 (2017).

[28] Torabi, A., Sigal, L.: Action classification and highlighting in videos. arXiv preprint arXiv:1708.09522 (2017).

[29] Wang, L., Zang, J., Zhang, Q., Niu, Z., Hua, G., Zheng, N.: Action recognition by an attention-aware temporal weighted convolutional neural network. Sensors 18(7), 1979 (2018).

[30] Wang, Y., Wang, S., Tang, J., O’Hare, N., Chang, Y., Li, B.: Hierarchical attention network for action recognition in videos. arXiv preprint arXiv:1607.06416 (2016).

[31] Wang X , Girshick R , Gupta A , et al. Non-local Neural Networks[J]. 2017.

[32] Woo S , Park J , Lee J Y , et al. CBAM: Convolutional Block Attention Module[J]. 2018.

[33] Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks.In: Proc. of European Conf. on Computer Vision (ECCV). 2014.

[34] Saining Xie, Rethinking Spatiotemporal Feature Learning: Speed-Accuracy Trade-offs in Video Classification, ECCV.2018.