A Novel Channel Attention Mechanism for Human Action Recognition Based on Convolutional Kernel

Xin Shi, Haiyang Jiang and Yuanyao Lu*

School of Information Science and Technology, North China University of Technology, Beijing 100144, China;
*Email: luyy@ncut.edu.com

Abstract. With the improvements of computer performance, deep learning has gradually expanded from 2D image tasks to 3D video tasks. Human action recognition is a typical 3D video task, which can achieve category classification by capturing human action characteristics. However, most of the videos are processed by encoding and decoding technology at current, thus the motion details are blurry, which makes it difficult for human action recognition. To solve this problem, we utilize the attention mechanism to "ignore" the blurred feature caused by video coding and decoding technology. Therefore, we hope to embed the attention mechanism in 3D spatiotemporal CNN to overcome this problem. Compared with 3D CNN, the effectiveness of our method is verified on UCF101 and HMDB51 dataset. And our method also proves the convolutional kernel implies channel-wise dependence. Although the improvement of the proposed method is limited, we hope the channel attention mechanism can help people to study neural network.

1. Introduction
Recently, the convolutional neural network (CNN) has been proved to be effective in 3D video tasks. However, is the feature of moving objects really acquired by CNN? When we observe videos, we always focus on the region of interest and the meaningful dynamic content, thus we empirically ignore some irrelevant details. Because of the unique characteristics of human eye observation, some researchers pay attention to how to achieve better observation effect with smaller bandwidth and storage memory, i.e. video coding and decoding technology. For example, H.264 standard, a typical video coding and decoding standard, was developed in response to the growing need for higher compression of moving pictures [1]. As is shown in Figure 1, the details of the hand and the arrow are blurry when they move too fast. Therefore, we propose a novel channel attention based on convolutional kernel embedded in 3D CNN for human action recognition to “ignore” the blurry region.

2. Related works
2.1. 3D spatiotemporal CNN
With the advent of AlexNet [2], some researchers hope to learn from the successful experience of 2D CNN to build 3D CNN. For example, Shuwang Ji [3] et al. firstly proposed 3D CNN for human action recognition. Then Kensho Hara et al. [4] modified 2D ResNet to 3D ResNet for human action recognition in 2017. In the same year, Joao Carreira et al. [5] proposed the I3D method constructed by ImageNet pre-trained model. CNN + LSTM is another better method of human action recognition,
which is different from 3D CNN. In recent years, with the increase of dataset, the recognition accuracy of 3D CNN has been greatly improved by pre-training in large-scale dataset.

2.2. Attention mechanism

Attention mechanism can selectively emphasize informative features and suppress less useful ones [6]. In recent years, a variety of attention mechanisms have been proposed for computer vision. One is the channel attention mechanism, which allocates the weight of feature maps in channel-wise dimension, such as SEnet [6]. Another method is the spatial attention mechanism, which allocates the weight of all feature maps in the spatial dimension. CBAM [7] and BAM [8], for example, use channel attention and spatial attention at the same time. However, those attention mechanisms obtain dependence by feature maps, while convolutional kernel maybe is also an important characteristic to obtain channel-wise dependence.

![Figure 1. Some archery pictures of UCF101(v_Archery_g01_c01).](image)

3. Method

Normally, a convolution operation takes $X \in \mathbb{R}^{C \times T \times H \times W}$ as input feature maps, which can generate feature maps $U \in \mathbb{R}^{C \times T \times H \times W}$ by the kernel size of $K=[k_1, k_2, \ldots, k_C]$, where $k_c$ refers to the parameters of the $c$-th filter. The outputs can be written as $U=[u_1, u_2, \ldots, u_C]$, where

$$u_c = k_c \ast X = \sum_{i=1}^{C} k_c^i \ast x^i.$$  \hspace{0.5cm} (1)

Here $\ast$ denotes convolution, $k_c=[k_1^c, k_2^c, \ldots, k_C^c]$, $X=[x^1, x^2, \ldots, x^C]$ and $u_c = R^{T \times H \times W}$.

Our method, named kernel-channel attention, is inspired by SEnet, which acquires channel-wise dependence by feature maps. While we consider that the convolutional kernel implies channel-wise dependence. Thus, we creatively squeeze the convolutional kernel to get the weight of each channel.

The kernel-channel attention consists of squeeze and excitation, as shown in Figure 2.

![Figure 2.](image)
3.1. Squeeze
To obtain the channel weight, we first clone the convolutional kernel of the generated feature map to squeeze it. The channel weight can be written as:
\[
z_c = F_{sq}(k_c) = \| k_c \| = \left( \sum_{i=0}^{h} \sum_{j=0}^{w} \sum_{k=0}^{c} k_{c\theta\nu} \right)^{\frac{1}{2}},
\]
where \( k_{c\theta\nu} \) is the value of the corresponding position of \( k_c \), \( z = [z_1, z_2, \ldots, z_c] \).

3.2. Excitation
Then we perform the excitation operation for the information aggregated in the squeeze operation to fully capture channel-wise dependencies. The channel-wise dependencies can be written as:
\[
s = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma_1(W_z \sigma_2(W_1 z)),
\]
where \( \sigma_1 \) denotes sigmoid function, \( W_1 \in \mathbb{R}^{C \times C} \), where \( \sigma_2 \) denotes ReLU function, \( W_2 \in \mathbb{R}^{C \times r} \), \( r = 16 \) and \( s = [s_1, s_2, \ldots, s_c] \).

Finally, we redistribute the channel weights to the feature map \( U \) and obtain ultimate feature map \( \tilde{X} \):
\[
\tilde{x}_c = F_{scale}(u, s_c) = s_c u_c,
\]
where \( \tilde{X} = [\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_c] \).

Figure 3 depicts the schema of the KC-ResNet (kernel-channel attention embed in ResNet) module.
4. Experiments

In this section, we show the details and results of the experiment. We choose 3D ResNet34 (R3D-34) network for comparison and train those models from scratch on UCF101 [9] and HMDB51 [10] dataset. We report the accuracy by split-1.

4.1. Network architectures

Table 1 provides the architectures of R3D-34 and our method. The input is 16 consecutive RGB images (i.e. L=16), and the size of each frame is resized and cropped to 112×112 pixels. On this basis, we build our method: from conv2 to conv5, we add the kernel channel attention module (KC-Attention).

4.2. Details

All the experiments were conducted on a server with a NVIDIA RTX 2080Ti GPU and installed with the windows10 system. The specific parameter settings are as follows:

- Python version: 3.7
- Deep learning framework: pytorch1.4
- Loss function: categorical cross entropy
- Optimizer: SGD
- Momentum: 0.9
- Weight decay: 1e-3
- Batch sizes: 70
- Total epoch: 200
- Learning rate: 0.1 (lr drops at 50 100 150 epochs.)
### Table 1. The architectures of R3D-34 and our method.

| Layer name | Output size          | R3D-34                                      | Our method                                      |
|------------|----------------------|---------------------------------------------|-------------------------------------------------|
| Conv1      | $L \times 56 \times 56$ | $7 \times 7 \times 7,64 ,\text{stride} 1 \times 2 \times 2$ | $7 \times 7 \times 7,64 ,\text{stride} 1 \times 2 \times 2$ |
| Conv2      | $L \times 56 \times 56$ | $[3 \times 3 \times 3,64] \times 3$ | $[3 \times 3 \times 3,64] \times 3$ |
|            |                      | $[3 \times 3 \times 3,64] \times 3$ | KC-Attention |
| Conv3      | $\frac{L}{4} \times 28 \times 28$ | $[3 \times 3 \times 3,128] \times 4$ | $[3 \times 3 \times 3,128] \times 4$ |
|            |                      | $[3 \times 3 \times 3,128] \times 4$ | KC-Attention |
| Conv4      | $\frac{L}{4} \times 14 \times 14$ | $[3 \times 3 \times 3,256] \times 6$ | $[3 \times 3 \times 3,256] \times 6$ |
|            |                      | $[3 \times 3 \times 3,256] \times 6$ | KC-Attention |
| Conv5      | $\frac{L}{8} \times 7 \times 7$ | $[3 \times 3 \times 3,512] \times 3$ | $[3 \times 3 \times 3,512] \times 3$ |
|            |                      | $[3 \times 3 \times 3,512] \times 3$ | KC-Attention |
|            | $1 \times 1 \times 1$ | Spatiotemporal pooling, fc layer with softmax | Spatiotemporal pooling, fc layer with softmax |

4.3. Results

### Table 2. R3D-34 comparison to our method.

| Method      | UCF101 | HMDB51 |
|-------------|--------|--------|
| Top1        |        |        |
| R3D-34      | 51.04% | 16.73% |
| Our method  | 51.57% | 17.58% |
| Top5        |        |        |
| R3D-34      | 72.75% | 45.69% |
| Our method  | 74.76% | 48.46% |

Table 2 compares our method to the original R3D-34. The top1 accuracy of our method is improved by 0.53% and 0.85% on the UCF101 and HMDB51 dataset, respectively. And the top5 accuracy of our method is improved by 2.01% and 2.77% on the UCF101 and HMDB51 dataset, respectively.

In Figure 4, it could be seen that the loss of the model with the KC attention mechanism was lower on the UCF101 and HMDB51 dataset.
Figure 4. Comparison of the model (red) with R3D model (green). (a) Comparison of losses between the two networks on UCF101; (b) Comparison of losses between the two networks on HMDB51.

Compared with the original 3D Resnet34, the accuracy of the 3D CNN with embedding KC-attention is improved and the loss of the 3D CNN with embedding KC-attention is lower. In addition, the effect of this improvement is more obvious on the HMDB51 dataset in the terms of accuracy and loss. Therefore, the performance of our method is more powerful.

Although the kernel-channel attention mechanism we proposed is more effective, it cannot prove that our method overcomes the interference caused by the video coding and decoding technology to blur the video frame.

5. Conclusion
In this paper, we proposed a kernel-channel attention mechanism based on the 3D ResNet34 to achieve video classification. Compared with the original 3D Resnet34, our method improved the accuracy of the model on UCF101 and the HMDB51 dataset, which proves that convolutional kernel also contains channel-wise dependence. The kernel-channel attention mechanism we proposed is only verified on the 3D ResNet34 structure, and whether it is effective on other structures still requires more experiments to prove. We hope that our proposed kernel-attention mechanism can inspire a new construction of attention mechanism. In the future, we will further explore the impact of video coding and decoding technology on human action recognition and demonstrate whether the attention mechanism can “ignore” the blurry region in the video frame.

Acknowledgments
This work is funded by the National Natural Science Foundation of China (61971007, 61571013).

References
[1] Krauss J 2003 Advanced Video Coding for Generic Audio-Visual Services, ITU-T Recommendation H.264. 29 66.
[2] Krizhevsky A, Sutskever I and Hinton G 2012 ImageNet Classification with Deep Convolutional Neural Networks, NIPS. Curran Associates Inc.
[3] Ji S, Xu W, Yang M and Yu K 2013 3D Convolutional Neural Networks for Human Action Recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 2013.35 221-231.
[4] Hara K, Kataoka H and Satoh Y 2017 Learning Spatio-Temporal Features with 3D Residual Networks for Action Recognition. ICCVW 2017. 3154-3160.
[5] Carreira J and Zisserman A 2017 Quo vadis, Action Recognition? A New Model and the Kinetics Dataset. CVPR 2017. 6299-6308.
[6] Hu J, Shen L, Albanie S, Sun G and Wu E 2020 Squeeze-and-Excitation Networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2020. **42** 2011-2023.

[7] Woo S, Park J, Lee J Y and Kweon I S 2018 CBAM: Convolutional Block Attention Module. *ECCV 2018*. Ferrari V, Hebert M, Sminchisescu C and Weiss Y. Eds. *Springer* 3-19.

[8] Park J, Woo S, Lee J Y and Kweon I S 2018 BAM: Bottleneck Attention Module. *arXiv*.

[9] Soomro K, Zamir A R and Shah M 2012 UCF101: A Dataset of 101 Human Action Classes from Videos in the Wild. *CRCV-TR-12-0*.

[10] Kuehne H, Jhuang H, Garrote E, Poggio T and Serre T 2011 HMDB: A Large Video Database for Human Motion Recognition. *ICCV 2011*. **1** 2556-2563.