Segments-Based 3D ConvNet for Action Recognition

Wei Li*, Ning Xu, Ge Liu, Linglan Zhao and Xiangzhong Fang
Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China
Email: liweihfyz@sjtu.edu.cn

Abstract. Learning to capture both long-range and short-range temporal information is crucial for action recognition task. Previous works utilize 3D ConvNets to capture short-range temporal dynamics in replacement of optical-flow which needs time-consuming extraction. However, dramatically increased parameters limit the capacity for modeling long-term interactions. In this paper, we propose Segments-based 3D ConvNet (S3D) to integrate both long-term and short-term temporal dynamics. Firstly, we utilize 3D ResNet without temporal downsampling to capture short-range video contents. Secondly, we integrate a sparse sampling strategy to model long-range temporal structure. Finally, experiments on UCF-101 and HMDB-51 datasets show the effectiveness of our S3D compared with corresponding 3D ConvNet.

Keywords. 3D ConvNet; sparse sampling; long-term temporal structure; short-term temporal dynamics.

1. Introduction
Action recognition task has drawn a rising amount of attention during past few years [1−4], which has a wide range of applications, e.g., security and behavior analysis. Unlike still images, videos contain rich temporal dynamics besides visual appearance, which poses a great challenge for action recognition task. Traditional two-stream convolutional networks handle visual content and temporal dynamics separately with Spatial 2D ConvNets and Optical-flow ConvNets, which achieve significant advantage over traditional methods with hand-crafted features. However, extracting optical-flow can be time-consuming which is not suited to real-world applications.

Recently, 3D ConvNets have been proposed to learn spatio-temporal filters, which directly create hierarchical representations of spatio-temporal data [3, 4]. Additional kernel dimension leads to expensive computational cost, which limits its capacity for long-term temporal content modeling. With the observation that consecutive video frames are highly similar, we propose to sample short video clips from uniformly divided segments. The overall architecture of our proposed model is shown in figure 1. The unnormalized scores of each class of all clips are fused together with fusion function to obtain a final prediction. Our Segments-based 3D ConvNet (S3D) are capable of capturing long-range temporal structure with near-equal inference computation cost compared with original 3D ConvNets. Also, short-term temporal dynamics of each clip is well captured by 3D ConvNets compared with 2D ConvNets, which is beneficial for recognizing actions with fast motion patterns.

In summary, we propose Segments-based 3D ConvNets (S3D) to integrate both long-term and short-term temporal information. Firstly, we experiment with different kernel settings of temporal convolution to select best performing 3D ConvNet structure. Secondly, different fusion functions are
compared in our experiments on UCF-101 dataset. Finally, we achieve superior performance compared with corresponding 3D ConvNet on both UCF-101 and HMDB-51 datasets.

**Figure 1.** The overall architecture of our proposed S3D. The unnormalized scores of all clips sampled from uniformly divided segments are fused together to obtain a final prediction.

The remaining of the paper is organized as follows. In Section 2, some related works of action recognition with 2D ConvNets or 3D ConvNets will be introduced. Section 3 describes our proposed model S3D in detail. Experimental results are represented in Section 4.

2. Related Works

2.1. 2D ConvNets

2D Convolutional networks have shown inspiring success in image recognition [5-7] and object detection [9-11]. As for action recognition, several approaches have been proposed to leverage 2D ConvNets to model visual information within each individual video frame [12, 13]. Karpathy et al. [14] compared several 2D ConvNets for action recognition. However, motion patterns are ignored with simply average pooling all predictions. To capture the rich temporal dynamics of video, Ref. [1] introduced a second ConvNet stream, which takes optical-flow frames as input. The predictions of these two streams are fused together with mean pooling function. Moreover, instead of training ConvNets with a single frame or consecutive frames with fixed length, TSN is proposed to model long-range temporal structure with a sparse sampling strategy. It enables efficient learning using the entire videos of arbitrary length. We extend the sparse sampling strategy with 3D ConvNet to integrate short-term motion patterns.

2.2. 3D ConvNets

In contrast to modeling temporal dynamics with optical-flow which needs time-consuming extraction, 3D ConvNets learn spatio-temporal filters to directly extract spatio-temporal representations from videos. Tran et al. [3] firstly exploited 3D ConvNets (C3D) for large-scale supervised training datasets with a clip length of 16 frames as input. Ref. [15] further decompose 3D convolution with a novel “R(2+1)D” with less computation cost. It alternates between spatial and temporal convolutions across the network. However, without the benefits of ImageNet pretraining, these architectures are designed relatively shallow and needs to be trained from scratch. In order to solve this problem, I3D [4] is proposed to inflate the model weights of 2D ConvNets pretrained on ImageNet to initialize 3D ConvNets. It leverages successful ImageNet architecture designs and their parameters to accelerate training process and obtains considerable performance improvements. Moreover, based on the diverse characteristics of categorical semantics and motion, Feichtenhofer et al. Ref. [16] extended I3D with
SlowFast networks, which involve a Slow pathway for spatial semantics and a Fast pathway to capture motion patterns.

3. Our Method

Given a video, we need to its semantic action category based on visual semantics and video dynamics. Previous works normally utilize optical-flow or 3D ConvNets to capture short-term temporal correlations. However, videos can be arbitrary length with rich long-term temporal structure. Both long-term and short-term motion patterns are important for accurate prediction. To integrate both types of motion into video recognition, we propose to combine a sparse sampling strategy with 3D ConvNets to model long-range video contents.

3.1. Model Architecture

Formally, with an input video, we uniformly divide it into N segments \( \{G_1, G_2, \ldots, G_N\} \). For each segment \( G_i \), we randomly sample a video clip \( C_i \) of \( T \) frames with a temporal stride of \( M \) to model short-term temporal dynamics. In order to guarantee a valid sampling, each segment contains the start frame of each clip and there is more than \( T \times M \) frames after the last segments. Instead of time-consuming extraction of optical-flow, in all our experiments, we utilize 3D ResNet-50 as our feature extractor to obtain a compact spatio-temporal representation. The detailed structure of our used backbone is shown in Table 1. The kernel dimensions are represented as \( \{T \times S^2, c\} \) for temporal, spatial and channel sizes. Strides and output dims are denoted as \( \{T \times S^2\} \). The strides for stage res2-5 are the same as original 2D version with an additional temporal dimension of 1.

| Stage     | Kernel dims | Stride | Output dims |
|-----------|-------------|--------|-------------|
| Input     | \( k_1 \times 7^2, 64 \) | 1 x 2^2 | \( K \times 224^2 \) |
| conv1     | \( 1 \times 3^2, 64 \) | 1 x 2^2 | \( K \times 112^2 \) |
| pool1     | \( k_2 \times 1^2, 64 \) \[ \begin{bmatrix} k_2 \times 1^2, 64 \\ 1 \times 3^2, 64 \\ 1 \times 1^2, 256 \end{bmatrix} \] | 1 x 2^2 | \( K \times 56^2 \) |
| res2      | \( 1 \times 3^2, 128 \) \[ \begin{bmatrix} 1 \times 3^2, 128 \\ 1 \times 1^2, 512 \end{bmatrix} \] | \times 3 | \( K \times 56^2 \) |
| res3      | \( 1 \times 3^2, 256 \) \[ \begin{bmatrix} 1 \times 3^2, 256 \\ 1 \times 1^2, 1024 \end{bmatrix} \] | \times 4 | \( K \times 28^2 \) |
| res4      | \( 1 \times 3^2, 512 \) \[ \begin{bmatrix} 1 \times 3^2, 256 \\ 1 \times 1^2, 1024 \end{bmatrix} \] | \times 6 | \( K \times 14^2 \) |
| res5      | \( 1 \times 3^2, 1024 \) \[ \begin{bmatrix} k_5 \times 1^2, 512 \\ 1 \times 1^2, 2048 \end{bmatrix} \] | \times 3 | \( K \times 7 \) |
| Global average pooling | \( 1 \times 1^2 \) |  |  |

In order to benefit from ImageNet pretraining, we inflate model weights corresponding 2D ConvNets to initialize 3D ConvNets. Temporal downsampling is not included as for our short clip length \( STS \). In order to select the best kernel settings \( \{k_1, k_2, k_3, k_4, k_5\} \) of temporal convolutions, we use corresponding 3D ResNet-50 without sparse sampling to examine the effectiveness of capturing short-term temporal dynamics. The detailed experiment results are shown in Section 4.

The entire architecture follows the design of 2D ResNet-50 [8] with external temporal convolution dimensions. There is no temporal downsampling in our architecture which means that feature maps after res5 stage has a same temporal dimension with the input video clip. After global average pooling
along both temporal and spatial dimension, we can obtain an overall representation vector for each clip. With compact spatio-temporal features $V = \{v_1, v_2, ..., v_n\}$ for all clips, we fuse the unnormalized scores of all clips together to obtain the final prediction, which can be formulated as follows:

$$p_i = W_p v_i + b_p$$

$$p^* = \text{softmax}(F(p_1, ..., p_N))$$

where $F$ means fusion function, $p_i$ denotes the unnormalized scores of all classes for clip $i$, $p^*$ represents the final prediction and $W_p, b_p$ are trainable parameters. We explore different fusion functions for our S3D: max pooling, mean pooling and non-linear projection. The detailed comparison results are shown in Section 4.

3.2. Training Strategy

With the fusion function, our model can be trained in an end-to-end way. In our experiments, cross-entropy loss is utilized to measure the performance of our model, which can be denoted as:

$$L(y, p) = \sum_i y_i \log p_i$$

where $y_i$ means ground-truth label of the video and $p_i$ denotes the final prediction result. We adopt stochastic gradient descent (SGD) for all our experiments with a weight decay rate of 0.0001 and a momentum of 0.9. During training, we reshape the shorter side of the video to a random value in [256, 320] pixels and randomly crop $224 \times 224$ pixels from the video. Random flipping is also utilized to augment the dataset. When testing, we rescale the shorter side of the video to 256 pixel and take the center crop. As for the temporal axis, we uniformly sample $N$ clips from the video for the evaluation of our S3D and original 3D ConvNets for a fair comparison.

4. Experiments

4.1. Datasets and Experiment Details

We evaluate our model on UCF-101 [17]. UCF-101 dataset contains 13,320 videos belong to 101 action categories. Videos are well trimmed for action recognition.

In order to fairly compare our S3D and corresponding 3D ConvNet, we choose 3D ResNet-50 from Ref. [16] as their backbone. Considering the memory limits, we set the length of input clip $T$ to 8 and temporal downsampling is not performed to mitigate the detrimental effect. The sampling stride $M$ for UCF-101 is set to 4 or 1 for a more detailed and convincing comparison with 3D ConvNet. We set the number of segments $N$ to 4 and only RGB frames are used. We use pytorch [18] for all our experiments. The dropout ratio before last fully-connected layer is set to 0.5 and no batch normalization parameter is frozen during training.

4.2. Evaluation of Our S3D

In this subsection, we focus on the evaluation of our Segments-based 3D ConvNet. We firstly show the effect of different kernel settings of temporal convolutions for capturing short-term temporal dynamics. Then we study the effect of different fusion functions to obtain the final prediction. Finally, we compare our results of S3D with corresponding 3D ConvNet to show the effectiveness of the sparse sampling strategy to model long-term temporal structure.

(1) Different Settings of Kernels Sizes. In our experiments, due to the memory limit, the temporal length of each clip is only 8. Temporal downsampling is not performed to mitigate the negative effect. Also, in early stages of 3D ConvNet, there is little correlations within small receptive field when objects move fast. Using temporal convolutions with kernel size $>$1 is detrimental for the performance. The experiment results of different kernel settings of temporal convolutions are shown in table 2. The early layers are of small receptive fields. We test 4 different kernel settings of temporal convolutions to select the best model architecture. When testing, we only use the center crop of 4 uniformly sampled
clips for the experiments. We can see that there is a significant accuracy drop when set \( k_1 \) to 3 which verifies our assumption. We obtain best performance with the setting of \( \{1,1,1,3,3\} \) and we utilize this as the default setting for all other experiments.

**Table 2.** Experiment results of different kernel settings on UCF-101 dataset (split 1).

| \( \{k_1,k_2,k_3,k_4,k_5\} \)      | top-1 | top-5 |
|------------------------------------|-------|-------|
| \( \{3,3,3,3,3\} \)               | 78.4  | 93.7  |
| \( \{1,3,3,3,3\} \)               | 83.1  | 95.6  |
| \( \{1,1,3,3,3\} \)               | 83.7  | 95.3  |
| \( \{1,1,1,3,3\} \)               | 83.8  | 95.9  |

(2) Different Fusion Functions. In equation (2), we use fusion function \( F \) to aggregate unnormalized class scores of all clips to obtain the final prediction. Here we evaluate three candidates, max pooling, mean pooling and non-linear projection. The experiment results are summarized in table 3. We can see mean pooling achieves the best performance and we choose it as the default fusion function in the following experiments.

**Table 3.** Experiment results of different fusion functions on UCF-101 datasets (split 1).

| Fusion Function       | top-1 | top-5 |
|-----------------------|-------|-------|
| Max Pooling           | 80.3  | 93.9  |
| Mean Pooling          | 85.9  | 96.4  |
| Non-linear Projection | 82.3  | 95.0  |

4.3. Comparison with 3D ConvNet

To verify the effectiveness of our S3D compared with corresponding 3D ConvNet, we conduct experiments on UCF-101 dataset. To make a fair comparison, we uniformly sample N clips with center cropping and fuse the predictions of different clips with mean pooling for the evaluation of 3D ConvNet. As for our S3D, we fuse the unnormalized class scores with mean pooling to obtain the final prediction with the same input clips. The experiment results on UCF-101 dataset are shown in table 4.

We fix the length of input clip \( T \) to 8 due to memory limit. We experiment with temporal stride \( M = 1 \) or \( M = 4 \) and our S3D obtains consistent performance improvements compared with 3D ConvNet. Our S3D surpasses its counterpart by 2.1% when \( M = 4 \) and 3.8% when \( M = 1 \). We can see a performance drop when \( M = 1 \) for original 3D ConvNet. This is due to the visual redundancy within adjacent frames with small temporal stride. The short-term temporal information is more distinct with large temporal stride. However, we can benefit from large interval between each sampled clip with the integration of the sparse sampling strategy. To some extent, the long-term temporal structure alleviates the detrimental effect of small temporal stride. As a result, our S3D achieves a superior performance with \( M = 1 \) compared with \( M = 4 \) for both top-1 accuracy and top-5 accuracy.

**Table 4.** Experiment results of S3D and 3D ConvNet on UCF-101 dataset (split 1).

| Model        | \( T \) | \( M \) | top-1 | top-5 |
|--------------|---------|---------|-------|-------|
| 3D ConvNet   | 8       | 4       | 83.8  | 95.9  |
|              | 8       | 1       | 82.8  | 96.3  |
| Our S3D      | 8       | 4       | 85.9  | 96.4  |
|              | 8       | 1       | 86.6  | 96.8  |
4.4. Comparison with the State-of-the-Arts
In order to verify the effectiveness of our proposed S3D, we compare with two the state-of-the-art methods on UCF-101 dataset in table 5. To have a fair comparison, we report their results with only RGB modality. We achieve considerable performance improvements with a top-1 accuracy of 86.6.

Table 5. Results of comparison with the state-of-the-arts (RGB only) on UCF-101 dataset (split 1).

| Methods  | top-1 |
|----------|-------|
| C3D [3]  | 85.2  |
| TSN [2]  | 86.2  |
| Our S3D  | 86.6  |

5. Conclusion
In this paper, we propose a novel framework, Segments-based 3D ConvNet (S3D), to integrate both short-term and long-term temporal information for action recognition. We utilize 3D ConvNet to capture short-term temporal dynamics and experiment with different kernel settings of temporal convolutions. A sparse sampling approach is further integrated to model long-term video contents. The experiments on UCF-101 dataset verify the effectiveness of our S3D compared with original 3D ConvNet.

References
[1] Simonyan K and Zisserman A 2014 Two-stream convolutional networks for action recognition in videos Advances in Neural Information Processing Systems (NIPS) eds Ghahramani Z, Welling M, Cortes C, Lawrence N D and Weinberger K Q.
[2] Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X and Gool L V 2016 Temporal segment networks: Towards good practices for deep action recognition European Conference on Computer Vision (ECCV).
[3] Tran D, Bourdev L, Fergus R, Torresani L and Paluri M 2015 Learning spatiotemporal features with 3D convolutional networks IEEE Inter-national Conference on Computer Vision (ICCV).
[4] Carreira J and Zisserman A 2017 Quo vadis, action recognition? A new model and the kinetics dataset arXiv preprint arXiv:1705.07750.
[5] Krizhevsky A, Sutskever I and Hinton G E 2012 Imagenet classification with deep convolutional neural networks Advances in Neural Information Processing (NIPS).
[6] Simonyan K and Zisserman A 2014 Very deep convolutional networks for large-scale image recognition arXiv preprint arXiv:1409.1556.
[7] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V and Rabinovich A 2015 Going deeper with convolutions IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
[8] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
[9] Redmon J, Divvala S, Girshick R and Farhadi A 2016 You only look once: Unified, real-time object detection IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
[10] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y and Berg A C 2016 SSD: Single shot multibox detector European Conference on Computer Vision (ECCV).
[11] Ren S, He K, Girshick R and Sun J 2015 Faster R-CNN: Towards real-time object detection with region proposal networks Advances in Neural Information Processing Systems (NIPS).
[12] Donahue J, Hendricks L A, Guadarrama S, Rohrbach M, Venugopalan S, Saenko K and Darrell T 2015 Long-term recurrent convolutional networks for visual recognition and description IEEE Conference on Computer Vision and Pat-tern Recognition (CVPR).
[13] Sigurdsson G A, Divvala S, Farhadi A and Gupta A 2017 Asynchronous temporal fields for action recognition IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[14] Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R and Li F-F 2014 Large-scale video classification with convolutional neural networks IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[15] Tran D, Wang H, Torresani L, Ray J, LeCun Y and Paluri M 2018 A closer look at spatiotemporal convolutions for action recognition IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[16] Feichtenhofer C, Fan H, Malik J and He K 2018 Slowfast networks for video recognition arXiv preprint arXiv:1812.03982.

[17] Soomro K, Zamir A R and Shah M 2012 UCF101: A dataset of 101 human actions classes from videos in the wild arXiv preprint arXiv:1212.0402.

[18] Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, Desmaison A, Antiga L and Lerer A 2017 Automatic differentiation in pytorch Advances in Neural Information Processing Systems Autodiff Workshop (NIPS-W).