STMP: Spatial Temporal Multi-level Proposal Network for Activity Detection

Guang Chen[1], Yuexian Zou[1,2(✉)], and Can Zhang[1]

[1] ADSPLAB, School of ECE, Peking University, Shenzhen, China
zouyx@pkusz.edu.cn
[2] Peng Cheng Laboratory, Shenzhen, China

Abstract. We propose a network for unconstrained scene activity detection called STMP to provide a deep learning method that can encode effective multi-level spatiotemporal information simultaneously and perform accurate temporal activity localization and recognition. Aiming at encoding meaningful spatial information to generate high-quality activity proposals in a fixed temporal scale, a spatial feature hierarchy is introduced in this approach. Meanwhile, to deal with various time scale activities, temporal feature hierarchy is proposed to represent activities of different temporal scales. The core component in STMP is STFH, which is a unified network implemented Spatial and Temporal Feature Hierarchy. On each level of STFH, an activity proposal detector is trained to detect activities in inherent temporal scale, which allows our STMP to make the full use of multi-level spatiotemporal information. Most importantly, STMP is a simple, fast and end-to-end trainable model due to its pure and unified framework. We evaluate STMP on two challenging activity detection datasets, and we achieve state-of-the-art results on THUMOS'14 (about 9.3% absolute improvement over the previous state-of-the-art approach R-C3D [1]) and obtains comparable results on ActivityNet1.3.

Keywords: Activity detection · Spatiotemporal feature hierarchy
Multi-level proposal detector

1 Introduction

Activity detection is a very challenging task, because it not only requires precise activity localization but also accurate classification in untrimmed videos. Current state-of-the-art activity detection approaches can be roughly divided into three categories: (1) *Regression-based approaches*. Inspired by the great success of Faster R-CNN [2] and YOLO [3] in object detection, most existing wonderful works, such as R-C3D [1] and SSAD [4], regarding activity detection as a regression problem. These methods usually contain three stages: C3D [5] as the backbone network for extracting features, following by a region proposal network for generating activity proposals, and finally a classifier is used for labeling. (2) *2D CNN based methods*. These approaches usually consist of several parts, and these parts are solved independently. Take the most successful framework for example, SSN [6] contains three separate parts, including frame-level actionness score generation, proposals generation [7] and action classification. (3) *Encoding temporal information with LSTM*, such as SST [8].

I. Kompatsiaris et al. (Eds.): MMM 2019, LNCS 11295, pp. 29–41, 2019.
https://doi.org/10.1007/978-3-030-05710-7_3

In this paragraph, we will make a brief analysis of advantages and disadvantages of the above methods. Regression-based approaches are end-to-end trainable frameworks. However, these methods lose spatial information and are not suitable for multi-scale activity scenarios (activities with various temporal durations). Because they down sample the spatial resolution to 1×1 and detect activity instances in a fixed temporal resolution. 2D CNN based approaches learn deep and effective representation of spatial information by utilizing hand-crafted features [8, 9] or deep features (*e.g.* VGG [10] and ResNet [11]). Unfortunately, these approaches are the framework of multi-stages and learned separately on image/video classification tasks. Such off-the-shelf representations may not be optimal for detecting activities in diverse video domains. From the results of existing experiments, 2D CNN based methods usually achieves better performance, owing to its good representation of spatial information.

Based on the above analysis, we propose a fast, end-to-end trainable network, named Spatial Temporal Multi-level Proposal Network (STMP). In our approach, a spatiotemporal feature hierarchy network is introduced to extract multi-level spatiotemporal features. For multi-level spatiotemporal features, a multi-level activity proposal detector network is designed to handle different temporal scale activities.

We summarize our contributions as follows:

(1) To learn the effective representation of spatial information, Spatial Multi-level Proposal (SMP) network with spatial feature hierarchy and multi-level proposal detector is introduced.
(2) To deal with various time scale activities, we add a temporal feature hierarchy in SMP, which is called STMP. This capacitate our model to represent multi-level spatiotemporal information simultaneously.
(3) Our STMP model achieves the state-of-the-art results on THUMOS'14 and obtains comparable results on ActivityNet1.3.

2 Related Work

2.1 Action Recognition

Action recognition is a core computer vision task that has been studied for decades. Just as image classification network can be used in object detection, action recognition models can be used in activity detection for feature extraction. Before the breakthrough of deep learning, Improved Dense Trajectories (iDT) [9] achieves remarkable performance by using SIFT and optical flow to eliminate the influence of camera motion. Later, two-stream network [12, 13] is proposed to learn both spatial and temporal features with single frame and stacked optical flows using 2D CNN [10, 11]. Although these methods achieve higher accuracy, they are extremely time-consuming and difficult to transform to end-to-end activity detection frameworks. Other approaches try to capture spatiotemporal information directly from raw video frames with 3D convolution, e.g. C3D and P3D [14]. These methods are very efficient and can be trained end-to-end. Therefore, we adopt C3D as our backbone network.

2.2 Object Detection

Object detection is a major breakthrough of deep learning in computer visions. There are two mainstream methods. Faster R-CNN [2] and its variants are typically "detection by classification" framework, which can be categorized as proposal-based methods. Proposal-free methods like SSD [15] make the most of multi-level spatial information in order to detect different scale objects. Compared to SSD, Faster R-CNN achieves better performance due to its high quality proposals.

The consensus of all these methods is to detect objects via regression, owing to the prior knowledge that each type of object has their own size and aspect ratio. This is also the maximum commonality with temporal activity detection. Each type of activity usually has its own duration, for example, drinking water usually lasts 10 s, rather than 10 min or more. This prior knowledge allows us to detect activities through the methods of object detection.

2.3 Temporal Activity Detection

This task needs to locate when and which type of activity happens in untrimmed diverse videos. Typical datasets such as THUMOS'14 [16] and ActivityNet [17] including thousands untrimmed videos and tens thousands of activity instances with various duration scales.

RNN and its variants are widely used in temporal activity detection [18–21]. Although these methods are successfully used in natural language processing, e.g. machine translation, they are not applicable to activity detection because they do not maintain long-term memory in practice [19]. Furthermore, textual information is regular and predictable, which is completely different with video temporal information.

Aside from approaches related to RNN, many researches adopt "detection by classification" framework. For example, S-CNN [22] separates the whole work into three stages: candidate segment generation, action classification and temporal boundary refinement. SSN [6] is also a multi-stage framework, containing frame-level actionness scores generation, candidate segments generation and action recognition. These discrete frameworks are often very difficult to train. Recently, an end-to-end trainable network named R-C3D [1] was proposed. It is a representative approach to detect activity via Faster R-CNN framework. Similar to R-C3D, we adopt Faster R-CNN framework and generate activity proposals from multi-level spatiotemporal feature maps. Compared with R-C3D, our model not only can encode effective spatiotemporal information, but also has better robustness for different temporal scale activities.

3 Our Approach

In this section, we will elaborate on our Spatial Temporal Multi-level Proposal (STMP) network. The framework of our approach is shown in Fig. 1, consisting of four components: a shared 3D ConvNet feature extractor as backbone network, spatiotemporal feature hierarchy network, multi-level proposal detector and classification network. More details of each component are shown as following.

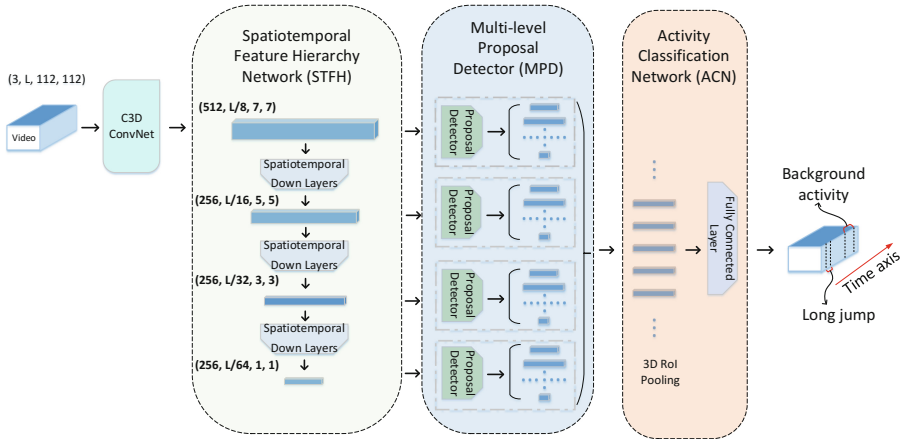

Fig. 1. Our STMP architecture. The C3D ConvNet is the backbone network and is used to extract spatiotemporal features from raw video frames. The spatiotemporal feature hierarchy is created for extracting hierarchical spatiotemporal features. On each level of the spatiotemporal feature hierarchy, an activity proposal detector is learned to detect candidate activity segments in a fixed temporal scale. These candidate segments are stacked and fed into a shared activity classification subnet, which outputs activity categories and refines temporal boundaries.

3.1 Backbone Network

We adopt the `conv1a` to `conv5b` layers from C3D ConvNet as backbone network for extracting spatiotemporal features. The input of 3D ConvNet is a sequence of RGB video frames with dimension $\mathbb{R}^{3\times L\times H\times W}$. The output is the feature maps $C_{conv5b} \in \mathbb{R}^{512\times\frac{L}{8}\times\frac{H}{16}\times\frac{W}{16}}$ (512 is the channel dimension), which is the shared input to spatiotemporal feature hierarchy and classification subnet. The number of input frames L can be arbitrary and is only limited by GPU memory. Typically, the height (H) and width (W) of the video frames are taken as 112.

Training: We pre-train the C3D network [5] on UCF101 [23].

3.2 Spatiotemporal Feature Hierarchy

In the unconstrained environment, activities in videos have various temporal scales. Besides, because of the movement of camera or object, the interest object in video often present different scales with time. Nevertheless, current mainstream solutions (*e.g.* [1, 6]) completely ignore these two facts. R-C3D down-samples spatial resolutions to 1×1, and utilizes a fixed temporal length feature for activity detection. SSN connects small basins into proposal regions by watershed algorithm.

In contrast, we introduce a network called Spatiotemporal Feature Hierarchy (STFH) that can encode multi-level spatiotemporal information simultaneously. As shown in Fig. 1, STFH takes `conv5b` feature maps as input, and outputs four hierarchical spatiotemporal feature maps. The spatial resolution of `conv5b` feature maps in the C3D ConNet is 7×7, and the temporal stride is 8. To learn hierarchical spatial

features, we add three branches with spatial feature maps size of 5×5, 3×3 and 1×1. Meanwhile, in order to detect activities of longer durations, we add three branches with temporal strides of 16, 32 and 64. Thus, there are 4 levels of the spatiotemporal feature hierarchy, each feature map $C_{stfh} \in \mathbb{R}^{256 \times \frac{L}{L_l} \times S_s}$, $L_l \in \{8, 16, 32, 64\}$, $S_s \in \{7 \times 7, 5 \times 5, 3 \times 3, 1 \times 1\}$.

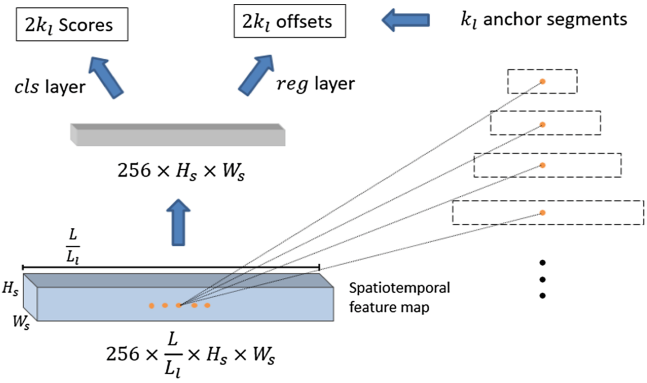

Fig. 2. A proposal detector consists of two ConvNet with kernel of size $1 \times H_s \times W_s$ and filters of $2k_l$ (one for classification, the other for regression).

3.3 Multi-level Proposal Detector

Inspired by SSD [15], a proposal detector is learned to generate high quality activity proposals for each level of spatiotemporal feature hierarchy. Similar to the RPN of Faster R-CNN, the anchor segments are pre-defined multi-scale windows centered at L/L_l uniformly distributed temporal locations. Whereby $L_l \in \{8, 16, 32, 64\}$, indicates 4 level temporal scales. Each temporal location specifies $K_l (l \in \{1, 2, 3, 4\})$ anchor segments. Thus, the total number of pre-defined anchor segments is $\sum_{l=1}^{4} K_l * \frac{L}{L_l}$.

As illustrated in Fig. 2, the $256 \times H_s \times W_s$ feature at each temporal location in C_{stfh} is fed into two sibling fully-connected layers: a segment-regression layer (*reg*) and a segment-classification (*cls*). Because the fully-connected layers are shared across all temporal locations, each proposal detector is naturally implemented with two sibling $1 \times H_s \times W_s$ convolutional layers. The first convolution layer is used to predict proposal score (background or activity), the second is used to predict a relative offset $\{\delta c_i, \delta l_i\}$ to the center location and the length of each anchor segment $\{c_i, l_i\}$, $i \in \{1, 2, \ldots, K_l\}$.

Training: Each level of spatiotemporal feature hierarchy and its corresponding proposal detector are considered an activity proposal network (APN). Typically, for training each APN, we assign a binary class label (of being an object or not) to each anchor segment. We assign an anchor segment with a positive label if it has the highest Temporal Intersection-over-Union (tIoU) for a given ground-truth activity or it has a

tIoU higher than 0.7 with any ground-truth activity. If the anchor segment has tIoU overlap lower than 0.3 with all ground-truth activities, given a negative label. We sample balanced batches with a positive/negative ratio of 1:1.

3.4 Activity Classification Network

Our STMP is a typical "detection by classification" network. Therefore, ACN have two main jobs: (1) Selecting high quality activity proposals generated from every feature map and getting fixed-size features for each proposal. (2) Activity classification and temporal boundaries refinement. For the first job, similar to the object detection [2], we employ a greedy Non-Maximum Suppression (NMS) strategy to eliminate highly overlapping and low confidence proposals from each proposal detector (the NMS threshold is set as 0.7). Then, we stack all the proposals (after NMS) from every proposal detector and employ a highly NMS thresh (such as 0.9 or 0.999). After that, following the standard practice in activity detection, a 3D RoI pooling layer is used to extract the fixed-size volume features for each variable-length proposal from the shared convolution features $C_{conv5b} \in \mathbb{R}^{512\times\frac{L}{8}\times\frac{H}{16}\times\frac{W}{16}}$. For the second job, we design two simple full-connected layers.

Training: Similar to APN, we need to assign activity labels to each proposal for training the classifier. Our tIoU thresh is set to 0.5, that means we assign an anchor segment with an activity (positive) label if it has the highest tIoU for a given ground-truth activity or it has a tIoU higher than 0.5 with any ground-truth activity. If the anchor segment has tIoU overlap lower than 0.5 with all ground-truth activities, given a background (negative) label. We sample balanced batches with an activity/background ratio of 1:3. And, the batch size is set to 64.

3.5 Loss Function

For each activity proposal network (there are 4 APN), softmax loss is used for classification (activity or not), and smooth L1 loss is used for regression. Specifically, our loss function for an APN is defined as:

$$\mathrm{L}(\{a_i\},\ \{t_i\}) = \frac{1}{N_{cls}}\sum\nolimits_i L_{cls}\left(a_i,\ a_i^*\right) + \lambda\frac{1}{N_{reg}}\sum\nolimits_i a_i^* L_{reg}\left(t_i,\ t_i^*\right) \tag{1}$$

Here, i is the index of an anchor segment in a batch and a_i is the predicted probability of anchor segment i being an activity. The ground-truth label a_i^* is 1 if the anchor segment is positive, and is 0 if the anchor segment is negative. $t_i = \left\{\delta\widehat{c_i},\ \delta\widehat{l_i}\right\}$ is the predicted relative offset to anchor segments. $t_i^* = \{\delta c_i, \delta l_i\}$ is the coordinate transformation of ground-truth segments to anchor segments. λ is the loss trade-off parameter. By default, we set $\lambda = 5$, and thus both *cls* and *reg* terms are roughly equally weighted.

Above is the single loss for a subnet. In our approach, there are 4 sub activity proposal network (APN) and one activity classification network (ACN). Thus, our joint loss function for a video is defined as:

$$\text{Loss} = \sum_{k=1}^{K} \gamma_k L(\{a_{ki}\}, \{t_{ki}\}) \tag{2}$$

Where K is the number of subnets (here is 5). γ_k balances the importance of models at different branch, here is set to 1 for each γ.

4 Experiments and Analysis

For studying the influence of multi-level spatial information on detection, we add an experiment (SMP) with temporal stride of each layer in STFH as 8. SMP denotes Spatial Multi-level Proposal network. We evaluate SMP and STMP on two challenging activity detection datasets: THUMOS'14 [16] and ActivityNet1.3 [17]. For both datasets, Average Precision (AP) and mean AP (mAP) are adopt for evaluation. More details are introduced from the following aspects: (1) implementation details of two experiments. (2) Experimental settings and evaluation on these public benchmarks.

4.1 Implementation Details

Experiments Settings. Table 1 shows the APN architecture (Spatiotemporal Feature Hierarchy and Multi-level Proposal Detector) of SMP and STMP. Here, each term of STFH and MPD denote the kernel size and filters of the convolutional layer.

Table 1. APNs architecture of SMP and STMP

#	Layer name	Output size	STFH	MPD
	Conv5b	$512 \times L/8 \times 7 \times 7$		$1 \times 7 \times 7, 2k$
SMP	APN_conv1_x	$256 \times L/8 \times 5 \times 5$	$1 \times 1 \times 1, 256$ $3 \times 3 \times 3, 256$	$1 \times 5 \times 5, 2k$
	APN_conv2_x	$256 \times L/8 \times 3 \times 3$	$3 \times 3 \times 3, 256$	$1 \times 3 \times 3, 2k$
	APN_conv3_x	$256 \times L/8 \times 1 \times 1$	$3 \times 3 \times 3, 256$	$1 \times 1 \times 1, 2k$
STMP	APN_conv1_x	$256 \times L/16 \times 5 \times 5$	$1 \times 1 \times 1, 256$ $3 \times 3 \times 3, 256$	$1 \times 5 \times 5, 2k$
	APN_conv2_x	$256 \times L/32 \times 3 \times 3$	$3 \times 3 \times 3, 256$	$1 \times 3 \times 3, 2k$
	APN_conv3_x	$256 \times L/64 \times 1 \times 1$	$3 \times 3 \times 3, 256$	$1 \times 1 \times 1, 2k$

Training Setup. We create a video buffer of 512 frames for THUMOS'14 and 768 frames for ActivityNet1.3, each frame in a video is resized to 172×128 (width $\times$ height) pixels, and we randomly crop regions of 112×112 from each frame. These buffers of frames act as input, and are generated by a sliding window.

Hyper-parameters. The weights of the filters of ACN and APNs are initialized by randomly drawing from a zero-mean Gaussian distribution with standard deviation 0.01. Biases are set to 0.1. All other layers are initialized from C3D model pre-trained on UCF-101. SGD algorithm with a momentum of 0.9 and a weight decay of 5×10^{-4} was adopted to train our model. Most importantly, we divided the whole network into two parts: backbone network and the rest (APNs and ACN), and take turns training the two parts alternately. The learning rate is initially set to 10^{-4} and then reduced by a factor of 10 after every 80k.

4.2 Experiments on THUMOS'14

THUMOS'14 is a widely used benchmark. The training set is the UCF-101 [23] dataset including 13320 trimmed videos of 101 categories while the validation and the test sets contain 200 and 213 untrimmed videos. In our experiments, all 200 videos are used as the training set and the results are reported on 213 test videos.

Experiments Setup. Since the GPU memory is limited, we create a video buffer of 512 frames and sample the frames at 25 fps to fit it in the GPU memory. As shown in Table 2, the number of anchor segments K in each level of STFH chosen for SMP (STMP) is 26 (7) with scale range 1:56 (1:7, 3:8, 4:8, 4:8). At 25 fps, the anchor segments of SMP (STMP) correspond to segments of duration between 0.64 and 17.92 s ([0.32, 2.24), [1.92, 5.12), [5.12, 10.24), [10.24, 17.92)).

Table 2. Anchor segments settings on THUMOS'14 for SMP and STMP.

Layer name	SMP		STMP		
	Strides	Anchor segments scale	Strides	Anchor segments scale	Temporal scale ranges
Conv5b	8	1:56	8	1:7	8–56
APN_conv1_x	8	1:56	16	3:8	48–128
APN_conv2_x	8	1:56	32	4:8	128–256
APN_conv3_x	8	1:56	64	4:8	256–512

Results. In Table 3, we present a superior activity detection performance of our SMP and STMP with existing state-of-the-art approaches. Our SMP (STMP) model shows about 8.4% (9.3%) absolute improvement @mAP 0.5 over R-C3D model, which clearly confirm that our model can encode effective spatiotemporal information simultaneously. Moreover, in Table 4, we present the Average Precision (AP) for each class in THUMOS'14 at tIoU threshold 0.5. Our STMP outperforms all the methods in most classes and achieves significant improvement (by more than 10% absolute AP over the R-C3D) for activities *e.g.* Crick Bowling, High Jump, Long Jump and Volleyball Spiking, which indicates the robustness of our model to multi-scale activities.

4.3 Experiments on ActivityNet

ActivityNet [17] is a recently released large-scale activity detection benchmark. We use the latest release (1.3) which has 10024, 4029 and 5044 videos containing 200 different types of activities in the training, validation and test respectively. Compared to THUMOS'14, ActivityNet1.3 is a large-scale dataset with longer activity instances and more classes.

Experimental Setup. Considering the long duration of activity instances of ActivityNet1.3, we create a video buffer of 768 frames and sample the frames at 3 fps to fit the GPU memory. The duration of the buffer is approximately 256 s covering 99.99% training activities. Similar to THUMOS'14, Table 5 shows the anchor segments settings on ActivityNet1.3.

Table 3. Activity detection results on THUMOS'14 test dataset (in percentage), measured by the mean average precision (mAP) of different tIoU thresholds α.

Method	α				
	0.1	0.2	0.3	0.4	0.5
Oneata et al. [24]	36.6	33.6	27.0	20.8	14.4
Richard et al. [25]	39.7	35.7	30.0	23.2	15.2
Yeung et al. [20]	48.9	44.0	36.0	26.4	17.1
Yuan et al. [21]	51.4	42.6	33.6	26.1	18.8
S-CNN [22]	47.7	43.5	36.3	28.7	19.0
CDC [26]	–	–	40.1	29.4	23.3
SSAD [4]	50.1	47.8	43.0	35.0	24.6
TCN [27]	–	–	–	33.3	25.6
R-C3D [1]	54.5	51.5	44.8	35.6	28.9
SSN [6]	**66.0**	59.4	51.9	41.0	29.8
SMP (ours)	60.4	58.8	55.7	48.7	37.3
STMP (ours)	62.5	**60.8**	**56.9**	**50.5**	**38.2**

Results. The comparison results between our SMP/STMP and other state-of-the-art methods [1, 19, 28, 29] published recently are shown in Table 6. Our SMP and STMP model achieve a significant improvement (about 2.8% and 3.5% absolute improvement in the average mAP of tIoU thresholds from 0.5:0.05:0.95) over R-C3D [1], which demonstrates the effectiveness of our method. Our STMP shows inferior performance over MSN [19], which using a deeper two-stream (RGB and optical flow) network. However, C3D is a simple 3D ConvNet, only uses low resolution RGB information. In Table 7, we compare detection speed of our model with R-C3D and two other state-of-the-art methods. S-CNN is similar to MSN and uses two-stream network to extract features. Despite the comparable results on ActivityNet1.3, our model is dozens of times faster than other framework (about 16x faster than S-CNN and 7x faster than DAP), which demonstrates the great potential of our model in future applications. Furthermore, our backbone network is relatively independent and can be replaced by other action recognition networks, e.g. I3D or P3D.

Table 4. Per-class AP at tIoU threshold $\alpha = 0.5$ on THUMOS'14 test dataset (in percentage)

	[24]	[20]	[21]	R-C3D	SMP (ours)	STMP (ours)
Baseball pitch	8.6	14.6	14.9	**26.1**	16.8	25.7
Basketball dunk	1.0	6.3	20.1	54.0	**56.1**	55.3
Billiards	2.6	9.4	7.6	8.3	20.6	**23.9**
Clean and Jerk	13.3	**42.8**	24.8	27.9	35.5	30.4
Cliff diving	17.7	15.6	27.5	49.2	52.2	**57.1**
Crick bowling	9.5	10.8	15.7	30.6	42.2	**44.9**
Cricket shot	2.6	3.5	13.8	10.9	**21.0**	**21.0**
Diving	4.6	10.8	17.6	26.2	28.1	**29.4**
Frisbee catch	1.2	10.4	15.3	20.1	19.6	**21.3**
Golf swing	**22.6**	13.8	18.2	16.1	18.4	15.3
Hammer throw	34.7	28.9	19.1	43.2	45.9	**51.8**
High jump	17.6	33.3	20.0	30.9	46.3	**48.8**
Javelin throw	22.0	20.4	18.2	47.0	63.9	**66.7**
Long jump	47.6	39.0	34.8	57.4	72.8	**74.8**
Pole vault	19.6	16.3	32.1	42.7	**48.2**	44.2
Shotput	11.9	16.6	12.1	19.4	34.0	**35.1**
Soccer penalty	8.7	8.3	19.2	15.8	**32.4**	25.2
Tennis swing	3.0	5.6	19.3	16.6	23.4	**23.9**
Throw discus	36.2	29.5	24.4	29.2	**44.9**	42.3
Volleyball spiking	1.4	5.2	4.6	5.6	23.7	**25.6**
mAP@0.5	14.4	17.1	19.0	28.9	37.3	**38.2**

Table 5. Anchor segments settings on ActivityNet1.3 for SMP and STMP

Layer name	SMP		STMP		
	Strides	Anchor segments scale	Strides	Anchor segments scale	Temporal scale ranges
Conv5b	8	1:64	8	1:16	8–128
APN_conv1_x	8	1:64	16	8:12	128–192
APN_conv2_x	8	1:64	32	6:8	192–256
APN_conv3_x	8	1:64	64	4:8	256–512

Table 6. Activity detection results on ActivityNet1.3 validation dataset. The performance are measured by mean average precision (mAP) at different tIoU thresholds α and the average mAP of tIoU thresholds from 0.5:0.05:0.95.

Method	α			
	0.5	0.75	0.95	Average
UPC [28]	22.5	–	–	–
R-C3D [1]	26.45	11.47	1.69	13.3
Wang et al. [29]	**42.48**	2.88	0.06	14.62
MSN [19]	28.67	**17.78**	**2.88**	**17.68**
SMP (ours)	27.30	14.70	1.45	15.10
STMP (ours)	34.23	13.96	2.40	16.88

Table 7. Activity detection speed during inference.

Methods	FPS
S-CNN [22]	60
DAP [30]	134.1
R-C3D (Titan X Pascal)	**1030**
SMP (ours on Titan X Pascal)	719
STMP (ours on Titan X Pascal)	972

5 Conclusion

In this paper, we propose a spatial temporal multi-level proposal (STMP) network for activity detection. We evaluate our approach on two benchmark datasets: THUMOS'14 and ActivityNet1.3. Experimental results demonstrate that STMP outperforms other approaches in terms of detection and computation on THUMOS'14. However, our method is superior to R-C3D on ActivityNet1.3, but inferior to MSN because C3D and 3D RoI pooling cannot encode long-term spatiotemporal information. Our future research will focus on developing a better video representation network for improving the performance of detecting on large multi-scale activities.

Acknowledgement. This paper was partially supported by the Shenzhen Science & Technology Fundamental Research Program (No: JCYJ20160330095814461) & Shenzhen Key Laboratory for Intelligent Multimedia and Virtual Reality (ZDSYS201703031405467). Special acknowledgements are given to Aoto-PKUSZ Joint Research Center of Artificial Intelligence on Scene Cognition & Technology Innovation for its support.

References

1. Xu, H., Das, A., Saenko, K.: R-C3D: Region convolutional 3D network for temporal activity detection. In: The IEEE International Conference on Computer Vision (ICCV), p. 8. (2017)
2. Girshick, R.: Fast R-CNN. arXiv preprint arXiv:1504.08083 (2015)
3. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
4. Lin, T., Zhao, X., Shou, Z.: Single shot temporal action detection. In: Proceedings of the 2017 ACM on Multimedia Conference, pp. 988–996. ACM (2017)
5. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 4489–4497. IEEE (2015)
6. Zhao, Y., Xiong, Y., Wang, L., Wu, Z., Tang, X., Lin, D.: Temporal action detection with structured segment networks. In: The IEEE International Conference on Computer Vision (ICCV) (2017)
7. Roerdink, J.B., Meijster, A.: The watershed transform: definitions, algorithms and parallelization strategies. Fundamenta informaticae **41**, 187–228 (2000)

8. Buch, S., Escorcia, V., Shen, C., Ghanem, B., Niebles, J.C.: SST: Single-stream temporal action proposals. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6373–6382. IEEE (2017)
9. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: 2013 IEEE International Conference on Computer Vision (ICCV), pp. 3551–3558. IEEE (2013)
10. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
12. Feichtenhofer, C., Pinz, A., Wildes, R.P.: Spatiotemporal multiplier networks for video action recognition. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7445–7454. IEEE (2017)
13. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Systems, pp. 568–576 (2014)
14. Qiu, Z., Yao, T., Mei, T.: Learning spatio-temporal representation with pseudo-3D residual networks. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 5534–5542. IEEE (2017)
15. Liu, W., et al.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
16. Jiang, Y., et al.: THUMOS challenge: action recognition with a large number of classes (2014)
17. Caba Heilbron, F., Escorcia, V., Ghanem, B., Carlos Niebles, J.: ActivityNet: a large-scale video benchmark for human activity understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 961–970 (2015)
18. Ma, S., Sigal, L., Sclaroff, S.: Learning activity progression in LSTMs for activity detection and early detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1942–1950 (2016)
19. Singh, B., Marks, T.K., Jones, M., Tuzel, O., Shao, M.: A multi-stream bi-directional recurrent neural network for fine-grained action detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1961–1970. IEEE (2016)
20. Yeung, S., Russakovsky, O., Mori, G., Fei-Fei, L.: End-to-end learning of action detection from frame glimpses in videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2678–2687 (2016)
21. Yuan, J., Ni, B., Yang, X., Kassim, A.A.: Temporal action localization with pyramid of score distribution features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3093–3102 (2016)
22. Shou, Z., Wang, D., Chang, S.-F.: Temporal action localization in untrimmed videos via multi-stage CNNs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1049–1058 (2016)
23. Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
24. Oneata, D., Verbeek, J., Schmid, C.: The LEAR submission at Thumos 2014 (2014)
25. Richard, A., Gall, J.: Temporal action detection using a statistical language model. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3131–3140 (2016)

26. Shou, Z., Chan, J., Zareian, A., Miyazawa, K., Chang, S.-F.: CDC: convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1417–1426. IEEE (2017)
27. Dai, X., Singh, B., Zhang, G., Davis, L.S., Chen, Y.Q.: Temporal context network for activity localization in videos. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 5727–5736. IEEE (2017)
28. Montes, A., Salvador, A., Pascual, S., Giro-i-Nieto, X.: Temporal activity detection in untrimmed videos with recurrent neural networks. arXiv preprint arXiv:1608.08128 (2016)
29. Wang, R., Tao, D.: UTS at activitynet 2016. AcitivityNet Large Scale Activity Recognition Challenge 2016, 8 (2016)
30. Escorcia, V., Caba Heilbron, F., Niebles, J.C., Ghanem, B.: DAPs: deep action proposals for action understanding. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 768–784. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_47