Multi-Scale Based Context-Aware Net for Action Detection

Haijun Liu, Shiguang Wang, Wen Wang, and Jian Cheng

***Abstract*—We address the problem of action detection in continuous untrimmed video streams, based on the two-stage framework: one stage for action proposals generation and the other for proposals classification and refinement. The context features inside and outside a candidate region (proposal) are critical for classification in action detection. Therefore, effective integration of these features with different scales has become a fundamental problem. We contend that different action instances and candidate proposals may need different context features. To address this issue, we present a novel multiple scales based context-aware net (MSCA-Net) to effectively classify the action proposals for action detection in this paper. For each candidate action proposal, MSCA-Net takes its multiple regions with different temporal scales as input and then generates suitable context features. Based on the "candidate-control" mechanism of LSTM, the proposed MSCA-Net specially adopts the two-branch structure: Branch1 generates multi-scale context features for each candidate proposal, whereas Branch2 utilizes the context-aware gate function to control the message passing. Extensive experiments on THUMOS'14, Charades daily and ActivityNet action detection datasets, demonstrate the effectiveness of the designed structure and show how these context features influence the detection results.**

***Index Terms*—Action detection, multiple scales, context-aware, gate function.**

I. Introduction

ACTION detection, as a crucial problem in computer vision, has attracted a lot of attention. It provides basic information and reduces the effort required to browse through lengthy videos for many real-world tasks, e.g., video retrieval and intelligent video surveillance [6]. Differing from action recognition that only needs to recognize the category of a trimmed video clip, action detection requires not only recognizing, but also precisely localizing the start and end times of each action in an untrimmed video. It's challenging due to large variations in illumination, viewpoints, poses, duration, background, and expensive computation. Recently, much CNN-based works [2], [3], [6], [10], [18], [25], [26], [41], [44] have made their efforts to solve this problem and have been proved to be useful for action detection because of the powerful ability of feature learning .

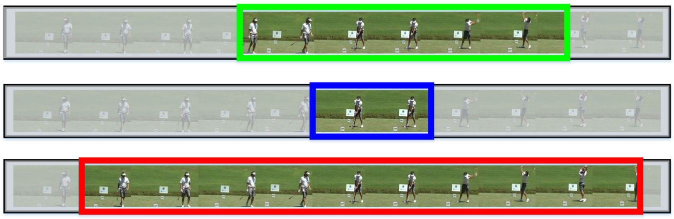


Fig. 1. Bad examples of the generated candidate proposals. The green box is the ground-truth interval of the action class "GolfSwing". The blue box denotes a proposal which has partial overlap with that ground-truth interval. The red box denotes a proposal which includes the ground-truth interval while involves too much background.

Current action detection pipeline is composed of two major components: proposals generation and classification. An intersection over union (IoU) detection threshold (e.g. $\alpha = 0.5$) is used to decide whether an action instance has been detected successfully. When the candidate proposal is generated exactly, we are prone to detect the action instance correctly following the framework of action recognition. However, it's hard to get the perfect candidate proposal. We usually meet the situations in Fig. 1, where the candidate region whether has partial overlap with the ground-truth interval or includes too much background. In that cases, we may fail to recognize the action class and localize the temporal boundary.

On the one hand, when a candidate proposal covers only a part of the ground-truth interval (the blue box in Fig. 1), it would be inadequate for action detection. In some cases, it may capture sufficient information (e.g. local details, scene information) for action recognition and answer the question "What is the action?", while fail to answer the question "When does it start and end?". Because classifying and precisely localizing an action instance in time not only require to look the evolution inner an action interval, but also demand to know what happened before and after the action procedure.

On the other hand, when a candidate proposal has much larger region than the ground-truth interval (the red box in Fig. 1), it would contain too much background. It leads the problem to return to the original point where both of the classification ("What is the action?") and the localization ("When does it start and end?") are not resolved. Therefore, the detector may not be able to distinguish the good proposals from the bad ones, as it always sees a significant amount of background [6]. What's

Manuscript received November 30, 2018; revised April 16, 2019; accepted July 4, 2019. Date of publication July 22, 2019; date of current version January 24, 2020. This work was supported in part by the National Natural Science Foundation of China (61671125, 61201271, and 61301269). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Elisa Ricci. *(Corresponding author: Jian Cheng.)*

The authors are with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: haijun_liu@126.com; xiaohu_wyyx@163.com; uestc_wangwen@163.com; chengjian@uestc.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2019.2929923

TABLE I
The Per-Class AP (Average Precision) at IoU Threshold $\alpha = 0.5$ on THUMOS'14 Dataset (%), When Integrating Candidate Proposal With Diverse Scaled Context Regions (S). The Generation of Scaled Context Regions Follows Section III-C, While With RoI-Pooled Feature Volume Size $1 \times 4 \times 4$ Following R-C3D [41]. The Original Proposal Features and Context Features are Concatenated Across Feature Channels and Subsequently Convolved ($F^{cat-conv}$)

Category	S			
	1.0(baseline)	1.0+0.8	1.0+1.5	1.0+2.1
	(with RoI-Pooled feature volume size $1 \times 4 \times 4$)			
BaseballPitch	10.7	15.5	**22.6**	19.0
HammerThrow	46.5	48.1	**55.7**	46.4
BasketballDunk	46.2	**49.4**	49.3	47.5
HighJump	40.6	40.5	**45.1**	43.0
Billiards	8.4	**14.5**	7.3	10.0
JavelinThrow	41.0	47.9	**50.0**	47.7
CleanAndJerk	27.1	**35.4**	31.9	33.6
LongJump	49.4	68.1	66.2	**70.3**
CliffDiving	53.7	**60.5**	59.9	58.1
PoleVault	41.8	**61.3**	46.7	48.9
CricketBowling	32.0	**35.7**	31.6	33.7
Shotput	20.0	19.0	**27.1**	20.8
CricketShot	16.9	**21.1**	18.2	19.4
SoccerPenalty	19.4	**26.7**	25.5	22.2
Diving	53.1	60.7	**62.3**	59.5
TennisSwing	13.3	**17.4**	16.9	12.6
FrisbeeCatch	**15.6**	9.6	8.0	13.7
ThrowDiscus	19.7	21.1	**27.1**	17.5
GolfSwing	23.5	**38.2**	32.2	25.2
VolleyballSpiking	8.6	9.2	**12.7**	11.1
mAP(α=0.5)	29.4	**35.0**	34.8	33.0

more, local details may be important for some action categories which involve many atomic action units. When CNN takes a large region as input, it sacrifices the capability of extracting local details.

It is well known to us that multiple scales play an important role in object detection [11], [46]. As to action detection, we argue that for each candidate action proposal, the corresponding multiple temporal regions, with different scales (resolutions), can generate valuable context features. To evaluate this claim, we conducted experiments on THUMOS'14 dataset [16] to make a sanity check, as shown in Table I. $S = 1.0$ denotes that we only use the original candidate proposals, the same with R-C3D [41] model. While others represent the original proposals additionally integrating the context region with scale value $S =$ 0.8, 1.5, 2.1, respectively, by our proposed MSCA-Net with the two-branch structure. Note that the RoI-Pooled feature volume size is still set $1 \times 4 \times 4$ following R-C3D [41]. From Table I, we can see that:

1) On the whole, the results by integration with scaled context features show improvements compared to the results with only original candidate proposal, demonstrating the effectiveness of context features from multiple temporal scales.
2) Different context scales show different effect on different actions.

Therefore, how to obtain appropriate context features for each candidate proposal from its corresponding multiple temporal scales is the key topic for action detection.

Each candidate action proposal, existing in two cases, whether has partial overlap (inside region) with the ground-truth interval or includes too much background (outside region). It is unknown to all of us, in which case the candidate proposal is, during testing. So how to effectively integrating those context features inside and outside a candidate region is the main research point in this paper.

There are two problems we should concentrate on: one is how to represent those multiple scaled temporal regions, and the other is how to automatically integrate them to perform the classification for action detection.

Inspired by the philosophy of LSTM [14], "candidate-control" mechanism,[1] we develop a novel network, multi-scale based context-aware network (MSCA-Net) to gather context features from multiple scaled temporal regions for each candidate action proposal for performing the action detection. MSCA-Net works with a two-branch structure: Branch1 generates multi-scale context features for each candidate proposal, while Branch2 utilizes the context-aware gate function to control the message passing. The candidate context features are obtained from multiple temporal regions with different scales (resolutions) using 3D RoI-Pooling^{+} layer, which focuses on "What is the candidate context features?". The gate function is controlled by the visual cues from all the temporal regions, which concentrates on "How much the candidate context features should be passed?". Context-aware gate filters are learned for each dimension and each channel. Experimental results on three action detection datasets demonstrated the effectiveness of our proposed MSCA-Net structure compared to the state-of-the-art action detection approaches.

To summarize, our main contributions are that.

- A multi-scale based context-aware network (MSCA-Net) is proposed to automatically integrate the context features of multiple temporal scales for each candidate action proposal. The two-branch structure is designed: one branch is for multiple context features generation, and the other branch is for message passing control.
- MSCA-Net structure achieves big promising results on THUMOS'14 [16] and Charades [28], and improves the results on ActivityNet [8] when using only C3D features.

II. Related Work

Recently action detection has achieved impressive improvements. In this section, we will briefly review some related works from the following aspects, action recognition, action detection and object detection.

Action Recognition: Action recognition has been extensively explored in recent years [15], [23], [32], [36], [38], [39], [45], and has achieved great improvements on the corresponding benchmarks UCF101 [31] and HMDB51 [17]. Here, we mainly focus on the models that could be used as feature extractor in action detection. Hand-crafted method DT [33] used dense trajectories and motion boundary descriptors to capture video information. iDT [34] further took camera motion into consideration.

[1] In LSTM, one branch generates the candidate values to be updated, while the other branch controls the update weights through a gate function.

These efficient video representation achieved state-of-the-art results on various of datasets while also required significant computing resources. Since deep learning based approaches achieved better performance with much less computation, hand-crafted features have become less popular. Deep learning-based methods can be mainly classified into two categories: two-stream and 3DCNN. Two-stream methods [9], [29] used a spatial stream ConvNet to capture appearance information from still frames and a temporal stream ConvNet to capture motion information between frames. 3DCNN [32] learned spatiotemporal features from raw video frames using deep 3-dimensional convolutional networks (C3D). It's efficient and could eliminate the burden of extracting motion features.

Action Detection: Action detection aims to classify the category of a segment proposal, while also needs to localize the start and end time of the action within an untrimmed video. It's very challenging since the length of actions and videos can be arbitrarily long, and the temporal border of some actions may be blurry compared to the spatial border in object detection, etc. Previous works mainly used sliding windows to generate candidates and focused on how to design hand-crafted features for action recognition. Recently, lots of works introduced deep learning into action detection framework [2]–[4], [6], [7], [13], [18], [26], [40]–[43], [47]. S-CNN [26] solved this problem via three segment-based 3D ConvNets. Nevertheless, since the classifiers are based on segment-level, they are unable to localize temporal boundary precisely. There are two mainstreams to solve this issue. Intuitively, some works [10], [18], [40] followed the philosophy of object detection by designing various temporal boundary regressors. Other works [25], [42] explored per-frame labeling to boost the precision of localizing. R-C3D [41] extended 2-dimensional object detection framework Faster R-CNN [24] to 3-dimensional temporal action detection. It learns an end-to-end network from raw video frames, which is more likely to avoid getting the sub-optimized solution. However, these features, extracted from two-stream or C3D network, may be insufficient to encode raw videos for action detection task. Alternatively, action detection approaches [2], [7] adopted recurrent neural network to handle this issue. The fully convolutional network to identify multi-scale temporal action proposals (FCNTAP) [13] was proposed that utilizes only the temporal convolutions to retrieve accurate action proposals for video sequences.

Some works focus on explore the context information to improve the performance of action detection. Semantic context cascade (SCC) [3] embraced the high-level semantic priors (action-object and action-scene) associated with human activities, generates high-quality class-specific action proposals and prunes unrelated activities in a cascade fashion. Temporal action localization network (TAL-Net) [4], based on Faster-RCNN framework, appropriately extended receptive fields to better exploit the temporal context of actions for both proposal generation and action classification in a multi-stream feature fusion manner. Temporal context network (TCN) [6] proposeed a novel representation, explicitly capturing context around a proposal, to rank these proposals for final action detection.

Object Detection: Since object detection approaches show great similarity with action detection, we will review some popular object detection frameworks in recent years, which are related to our work. Lots of excellent works [5], [20], [24], [46] have been done to push the limits of object detection. Faster R-CNN [24] followed the pipeline: proposals generation and classification, which is a general and robust object detection framework. Context has been shown useful in object detection [1], [11], [37], [46]. Gidaris [11] preliminarily explored the effectiveness of multi-region. Furthermore, Zeng [46] proposed a gated bi-directional CNN (GBD-Net) to integrate local and contextual regions. The full investigation of multiple regions in object detection gives prodigious inspiration for our work. However, compared to GBD-Net, our proposed MSCA-Net has at least three contributions beyond. (1) GDB-Net investigates the spatial context for object detection, while our work focus on the temporal context for action detection. Such a domain shift introduces several challenges. (2) We apply different approaches to effectively integrate the context features with different scales. GDB-Net proposed a gated bi-directional CNN to pass messages to make the neighboring support regions communicate with each other, while our work was based on the "candidate-control" mechanism of LSTM. (3) Moreover, GDB-Net is much more complex, directly applying it to action detection to build an end-to-end framework would bring enormous memory pressure.

III. Multi-Scale Based Context-Aware Network (MSCA-Net)

In this section, we first describe the baseline detector R-C3D [41] in Section III-A to make the paper self-contained. Then we provide a brief overview of our proposed model MSCA-Net in Section III-B. The generation of multiple scaled temporal context features is discussed in Section III-C, while Section III-D and Section III-E focus on the detailed design of MSCA-Net. Finally, we explain the details of the training scheme and parameters setting in Section III-F.

A. R-C3D Pipeline

R-C3D [41], inspired by the faster R-CNN [24] object detection method, computes fully-convolutional 3D ConvNet features and generates temporal region proposals likely to contain activities, then predicts the activity classes and refines the boundaries for these 3D regions. In order to utilize video features at any temporal granularity, it extends 2D region of interest (RoI) pooling to 3D which extracts a fixed-length feature representation for these proposals.

As illustrated in Fig. 2, R-C3D mainly consists of three components: a 3D ConvNet feature extractor, a temporal proposal subnet and an action classification subnet. The 3D ConvNet [32] is adopted to extract rich spatio-temporal features. The proposal subnet predicts variable length temporal segments that potentially contain activities. The classification subnet classifies these proposals into specific activity categories or background, and further refines the proposal segment boundaries.

B. MSCA-Net Framework Overview

Due to the excellent performance and the generalization ability of R-C3D [41] for action detection, we adopt the R-C3D as our action detection pipeline with three steps:

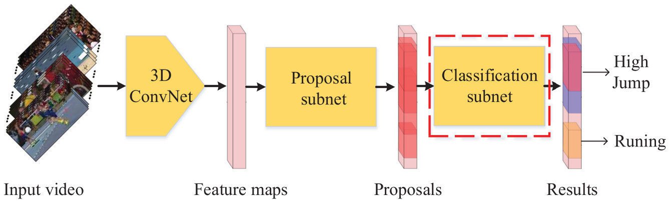


Fig. 2. The pipeline of R-C3D. The raw video frames are inputted into the 3D ConvNet, obtaining the convolutional features. Based on those features, the proposal subnet generates proposals (candidate activities) of variable length along with confidence scores. Finally based on the proposals and the convolutional features, the classification subnet predicts the activity labels and refines the segment boundaries. The classification subnet (in red square) is the main focus of our model.

- Step 1) Candidate regions generation. Numerous of candidate proposals are generated via a fully-convolution proposal network.
- Step 2) RoI features extraction. Given the candidate proposals generated in step 1 and the 3DCNN feature maps of the untrimmed video, a max-pooling function is operated on the 3DCNN feature maps to extract fixed-size RoI feature.
- Step 3) Classification. The extracted RoI features go through several layers to predict the class label and refine temporal locations of the candidate proposals.

Our proposed model MSCA-Net mainly focuses on how to effectively use (or integrate) those multi-scale temporal context features of each candidate action proposal. Fig. 3 shows the architecture of our proposed model. Based on the R-C3D pipeline, MSCA-Net takes an untrimmed video and those candidate proposals as input, and uses the 3D RoI-Pooling$^+$ layer to obtain fixed-size features from multiple temporal regions with different scales. Then, the two-branch structure is utilized to generate suitable context features for each candidate action proposal. Finally, classification and regression are conducted. If only the candidate proposal ($S = 1.0$) and one branch is considered, Fig. 3 becomes R-C3D network.

In order to make the best of visual cues in surrounding/inner proposals, we modify the R-C3D as follows.

1) In the 3D RoI-Pooling$^+$ step, temporal regions with the same center location but different length are pooled from the same feature maps. The temporal regions with different length get feature volumes with the same size after 3D RoI-Pooling$^+$. To make a balance on keeping spatial information and temporal information, the RoI-Pooled feature volume is modified from $1 \times 4 \times 4$ to $4 \times 2 \times 2$. Therefore, the filter of the $FC6$ layer of R-C3D could also adopt the pre-trained model to initialize, and we could obtain the context features with different resolutions.
2) Context features with different resolutions optionally go through several CNNs to get their high-level features.
3) The Branch1 takes the advantage of multiple temporal scale context regions to generate candidate context feature, while the Branch2 utilizes the gate functions to control the message transmission for each candidate proposal.

C. RoI-Pooling Features With Different Context Scales

We modify 3D RoI-Pooling layer [41] to obtain features with different scales and context regions, which could enlarge or narrow the temporal candidate region at a certain scale. We term it as 3D RoI-Pooling$^+$ layer.

Given a candidate temporal proposal $p^o = [s^o, e^o]$ with start location s^o and end location e^o, its scaled proposal is denoted by p^S. p^S is obtained by changing the original proposal p^o along temporal axis with scale value S as follows,

$$p^S = [s^o - (e^o - s^o)(S-1)/2, \quad e^o + (e^o - s^o)(S-1)/2]. \tag{1}$$

For a arbitrarily long candidate proposal p^o, with its corresponding 3D convolutional features f^o on feature maps, 3D RoI-Pooling$^+$ layer first extends p^o to p^S, and then conducts max-pooling on the feature maps to get fixed-size features. In our experiments, we set the scale values $S = 0.8$, 1.0, 1.5, 2.1, respectively, which represent the regions with different scales and contexts. These scaled regions corresponding to $p^{0.8}$, $p^{1.0}$, $p^{1.5}$ and $p^{2.1}$ are respectively warped into $f^{0.8}$, $f^{1.0}$, $f^{1.5}$ and $f^{2.1}$ by the 3D RoI-Pooling$^+$ layer. Here, a larger S value means a lower resolution for the original proposal but more contextual information, while a smaller S denotes a larger resolution for the original proposal with more local details. For simplicity, the features from the original temporal proposal ($S = 1.0$) and the corresponding multiple context regions ($S = 0.8$, 1.5, 2.1, we term them as the context features) are RoI-Pooled to the same dimensionality ($4 \times 2 \times 2 \times 512$).

D. Candidate Multiple Context Features Generation

Branch1 in Fig. 3 shows the procedure of candidate context features generation. It takes features $f^{0.8}$, $f^{1.5}$, $f^{2.1}$ as input and outputs the candidate context features $\mathbf{h}_x$. Fusion method cat is adopted to integrate features with different scales and context regions. It could be formulated as follows,

$$h^{0.8} = \delta(f^{0.8}\mathbf{w}_{0.8} + \mathbf{b}_{0.8}), \tag{2}$$

$$h^{1.0} = \delta(f^{1.0}\mathbf{w}_{1.0} + \mathbf{b}_{1.0}), \tag{3}$$

$$h^{1.5} = \delta(f^{1.5}\mathbf{w}_{1.5} + \mathbf{b}_{1.5}), \tag{4}$$

$$h^{2.1} = \delta(f^{2.1}\mathbf{w}_{2.1} + \mathbf{b}_{2.1}), \tag{5}$$

$$\mathbf{h}_a = cat(h^{0.8}, h^{1.5}, h^{2.1}), \tag{6}$$

$$\mathbf{h}_x = \tanh(\mathbf{h}_a \otimes \mathbf{w}_x + \mathbf{b}_x), \tag{7}$$

where δ refers to the RELU [22] function. $\mathbf{w}_s$ and $\mathbf{b}_s$ ($s = 0.8$, 1.0, 1.5, 2.1, respectively) are the parameters of optional CNNs to get high-level features of the context features. Fusion method cat stacks feature maps at the same spatio-temporal locations (l, h, w) across the feature channels c. The integrated multiple context feature $\mathbf{h}_a$ is more robust to match diverse kinds of visual pattern. $\mathbf{w}_x \in \mathbb{R}^{3\times3\times3\times NC\times C}$ and $\mathbf{b}_x \in \mathbb{R}^C$, are learned to reduce the dimensionality and get high-level features, where N is the number of sources of feature maps. $\otimes$ represents the convolution operation. Function $\tanh$ is applied to make the context features under control to prevent gradient explosion.

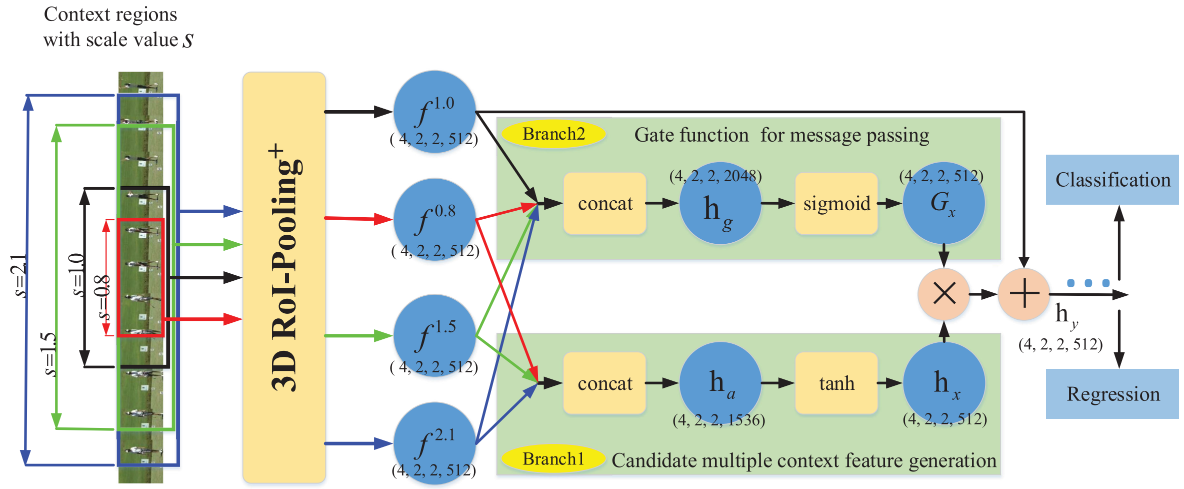


Fig. 3. The detailed architecture of our proposed model, which takes an untrimmed video and the candidate proposal ($S = 1.0$) as input, and outputs the category label and temporal location refinement. 3D RoI-Pooling^{+} operation is applied to get context features ($S = 0.8, 1.5, 2.1$) with different scales. To get suitable context features for individual candidate proposal, we propose the two-branch structure for action detection. Specifically, Branch1 generates candidate context features, while Branch2 utilizes the gate function to control message passing. The shapes (L, H, W, C) of these features are listed under the notation of the corresponding features, where L, H, W, C are the length, height, width and the number of channels of the feature maps.

Since the aim of the proposed structure is to refine the features $f^{1.0}$ of the original temporal proposal from other suitable context regions, we naturally add an identity mapping layer in the structure. The refined RoI features can be represented as follow,

$$\mathbf{h}_y = f^{1.0} + \gamma \mathbf{h}_x, \tag{8}$$

where γ is a constant to control the magnitude of messages from other context features. The exploration of γ is given in the ablation experiments in Section IV-A.

E. Gate Function for Message Passing

Instead of passing messages in the same way for all the candidate context features, the gate function is introduced to control message passing. Gate function is also implemented as convolution. The design of gate filter considers the following aspects:

1) Since a complex action can be regarded as a sequential composition of many "sub-actions" (atomic actions) [35], different "sub-action" may need different magnitude of context features. Furthermore, $\mathbf{h}_x$ has multiple feature channels. The gate filters should be learned for each dimension and each channel.
2) The message passing rates should be automatically controlled by the responses to all the context features and the features from the original temporal proposal. Moreover, it could be determined by visual cues from nearby regions, e.g. one "sub-action" could be influenced by the "sub-actions" around. Therefore, the size of gate filters should not be $1 \times 1 \times 1$, while $3 \times 3 \times 3$ is used in our implementation.

The considerations above constitute the idea of "context-aware".

The gate function in Branch2 and the proposed gated structure in Fig. 3 can be summarized as follows,

$$\mathbf{h}_g = cat(h^{1.0}, h^{0.8}, h^{1.5}, h^{2.1}), \tag{9}$$

$$G_x = \sigma(\mathbf{h}_g \otimes \mathbf{w}_g + \mathbf{b}_g), \tag{10}$$

where $\mathbf{h}_g$ denotes the features for encoding the message passing controller. $\sigma(x) = 1/[1 + \exp(-x)]$ is the element-wise sigmoid function that make the passing rate in the range of $(0, 1)$. G_x is the controlling gate function and is implemented by convolution layers with the sigmoid function. When $G_x(l, h, w, c) = 0$, there is no message passed. When $G_x(l, h, w, c) = 1$, all the candidate regions has the same passing rate which is only determined by γ. More generally, the learnable parameters $\mathbf{w}_g$ and $\mathbf{b}_g$ use features from co-located and all the temporal regions to decide the passing rate for each "sub-action". Thence, the changed features in Eq. (8) can be represented as follow,

$$\mathbf{h}_y = f^{1.0} + \gamma \mathbf{h}_x G_x. \tag{11}$$

F. Implementation Details

We follow all the setups of R-C3D [41], including detection loss, training procedure and action prediction. For a training sample with class label y and ground-truth interval $g = [g_s, g_e]$, the loss function for our framework can be summarized as a softmax loss function L_{cls} for classification, and a smooth L_1 loss function L_{reg} [12] for regression. Specifically, the objective function is given by,

$$L(y, y^t, g, g^t) = L_{cls}(y, y^t) + \lambda L_{reg}(g, g^t), \tag{12}$$

where y^t is the predicted probability of the proposal, and g^t is the predicted relative offset. λ is set as 1 in our implementation. In proposal subnetwork, the binary classification loss L_{cls} predicts whether the proposal contains an action or not, and L_{reg} optimizes the relative offset between proposals and ground truths. In classification subnetwork (MSCA-Net), the multiclass classification loss L_{cls} predicts the specific activity class for the proposal, and L_{reg} optimizes the relative offset between actions and ground truths. We optimize the network by jointly training the classification and regression task for both the proposal subnetwork and our MSCA-Net. The standard post-process NMS is performed to suppress some overlapped detection results in action prediction.

To save GPU memory and accelerate the training procedure, we freeze the first two convolutional layers of 3D ConvNet [32] in our model. For THUMOS'14 dataset, the learning rate is fixed at 0.0001 for 5 epoches. For Charades dataset, we fixed the learning rate at 0.0001 for the first 10 epoches and then decreased to 0.00001 for 5 further epoches. For ActivityNet dataset, we also fixed the learning rate at 0.0001 for the first 10 epoches and then decreased to 0.00001 for next 5 epoches. The optional CNNs ($\mathbf{w}_s$ and $\mathbf{b}_s$) are not added, while γ is set as 0.2 based on the ablation experiments. All the other hyper-parameters follows [41].

TABLE II
THE MAP RESULTS ON THUMOS'14 DATASET (%) COMPARING WITH STATE-OF-THE-ART METHODS AT VARIOUS IOU THRESHOLD α

α	0.1	0.2	0.3	0.4	0.5	0.6	0.7
S-CNN [26]	47.7	43.5	36.3	28.7	19.0	10.3	5.3
CDC [25]	49.1	46.1	40.1	29.4	23.3	13.1	7.9
SSAD [18]	50.1	47.8	43.0	35.0	24.6	-	-
Xiong et.al. [40]	64.1	57.7	48.7	39.8	28.2	-	-
SSN [47]	60.3	56.2	50.6	40.8	29.1	-	-
CBR-TS [10]	60.1	56.7	50.1	41.3	31.0	19.1	9.9
TPC+FGM [42]	-	-	44.1	37.1	28.2	20.6	12.7
TCN [6]	-	-	-	33.3	25.6	15.9	9.0
BSN [19]	-	-	53.5	45.0	36.9	28.4	20.0
TAL-Net (RGB) [4]	49.3	-	42.6	-	31.9		14.2
TAL-Net [4]	59.8	57.1	53.2	48.5	**42.8**	**33.8**	**20.8**
R-C3D [41]	54.5	51.5	44.8	35.6	28.9	-	-
R-C3D (baseline)	57.0	55.2	50.4	41.0	29.4	21.1	10.0
MSCA-Net (Ours)	61.6	**61.5**	**58.4**	**51.6**	41.8	31.5	18.7

IV. EXPERIMENTS

In this section, we evaluated the performance of our proposed model on three challenging action detection datasets: THUMOS'14 [16], Charades [28] and ActivityNet [8]. The ablation experiments were further given on THUMOS'14, considering its popularity and the possession of actions with diverse complexities: the subjects, objects and scenes involved; the duration and change rate of an action; the clarity of the border of an action.

A. THUMOS'14

The THUMOS'14 [16] detection dataset contains 20 action categories collected from over 24 hours of videos. The training set contains 2765 trimmed videos while the validation set and the test set contain 200 and 213 untrimmed videos respectively. The validation set was used as the training set, while the final results were reported on the test set following the standard evaluation metric (per-class AP and mAP over all the classes). Following the setup of R-C3D [41], we initialize the 3DConvNet part of our model with C3D weights [32] trained on Sports-1M and finetuned on UCF101.

Comparison with the state-of-the-art approaches: We reimplemented the R-C3D [41] model with PyTorch framework[2] as our baseline method. Some existing published approaches are adopted to compare with our proposed MSCA-Net. The results are reported in Table II. we can see that:

1) The baseline method (R-C3D (baseline)) obtains a little better results compared to the original R-C3D [41]. The difference may be come from the implementation with different frameworks, PyTorch vs Caffe.
2) Compared to R-C3D (baseline), our MSCA-Net performs much better at all the α, especially at the $\alpha = 0.5$ (41.8% vs 29.4%). It demonstrates the effectiveness of our MSCA-Net to integrate the context features of multiple temporal scales for each candidate action proposal.
3) When increasing the detection threshold, most methods exhibited dramatic performance drop, e.g. Xiong *et al.* [40] dropped from 64.1% (mAP($\alpha = 0.1$)) to 57.7% (mAP($\alpha = 0.2$)). Our MSCA-Net model dropped slowly and obtained significant performance with a better localization accuracy, which confirms the analysis in ablation experiments later that model with multiple context scales has the ability of reducing localization error.
4) MSCA-Net obtains the best results when $\alpha = 0.2, 0.3, 0.4$, and comparable results to the state-of-the-art under other α. TAL-Net [4] performs better than our MSCA-Net when $\alpha = 0.5, 0.6, 0.7$. However, this is because TAL-Net [4] adopts the two-stream feature fusion framework, including RGB and flow data. While our MSCA-Net only performs on the RGB data. When TAL-Net [4] only performs on RGB data, much worse results are obtained compared to our MSCA-Net.
5) All of the TCN [6], TAL-Net (RGB) [4] and our MSCA-Net methods focus on the context features exploring for action detection. The better performance of MSCA-Net demonstrates the effectiveness of our multiple-scale based context-aware method for context features integrating.

Moreover, the Average Precision (AP) for each class at IoU threshold $\alpha = 0.5$ is shown in Table III. For per-class AP, our MSCA-Net model outperforms the other two comparison methods in most classes and shows significant improvements. Fig. 5(a) shows some visualization results.

Investigation on combination with different context regions: To investigate the performance when combining our two-branch structure with different context regions, we enlarge the candidate proposal with a series of scale values S. The experimental results for these settings are shown in Table IV. It can be seen that.

1) Integrating context regions with multiple scales using our MSCA-Net substantially improves the detection performance as the number of context scales increases.
2) Moreover, the R-C3D (baseline) performs worse compared to MSCA-Net ($S = 1.0$). Both of them only adopt the original proposal region ($S = 1.0$). The only difference is that R-C3D (baseline) utilizes the 3D RoI-Pooling layer while MSCA-Net utilizes the modified 3D RoI-Pooling$^+$ layer, which modified the RoI-Pooled feature volume from $1 \times 4 \times 4$ to $4 \times 2 \times 2$ to make a balance on keeping spatial information and temporal information. It will be detailedly analyzed in the following subsection (Table V).

[2] https://github.com/sunnyxiaohu/R-C3D.pytorch.

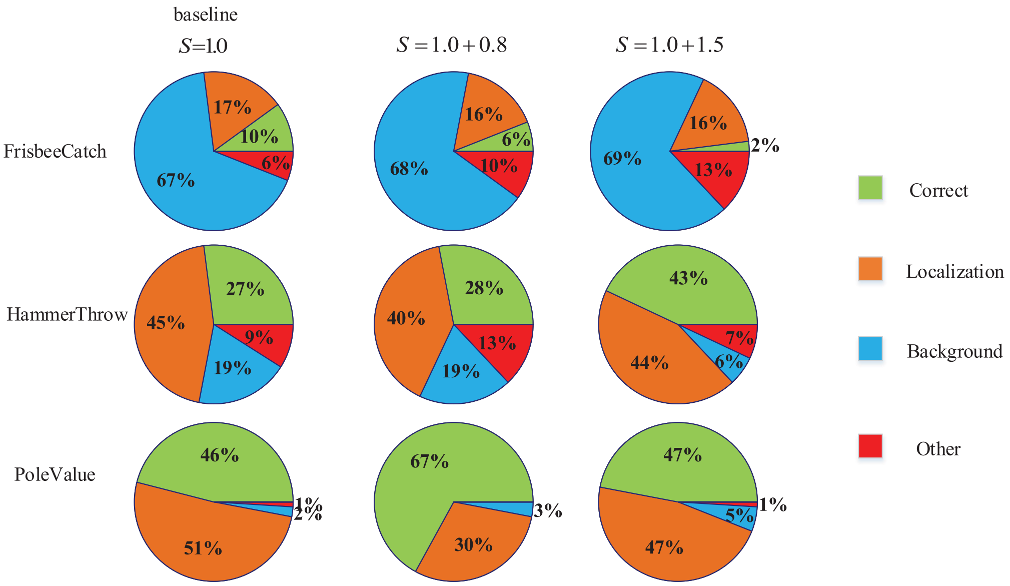


Fig. 4. Fraction of the detection results at 70% average recall rate that are correct (Correct), or false positive due to poor localization (Localization), confusion with background (Background), confusion with other action categories (Other). Left column: the results of R-C3D baseline with the original candidate region. Middle column: the results of integrating more local details (inside the proposal) $S = 1.0 + 0.8$. Right column: the results of integrating more context (surrounding the proposal) $S = 1.0 + 1.5$.

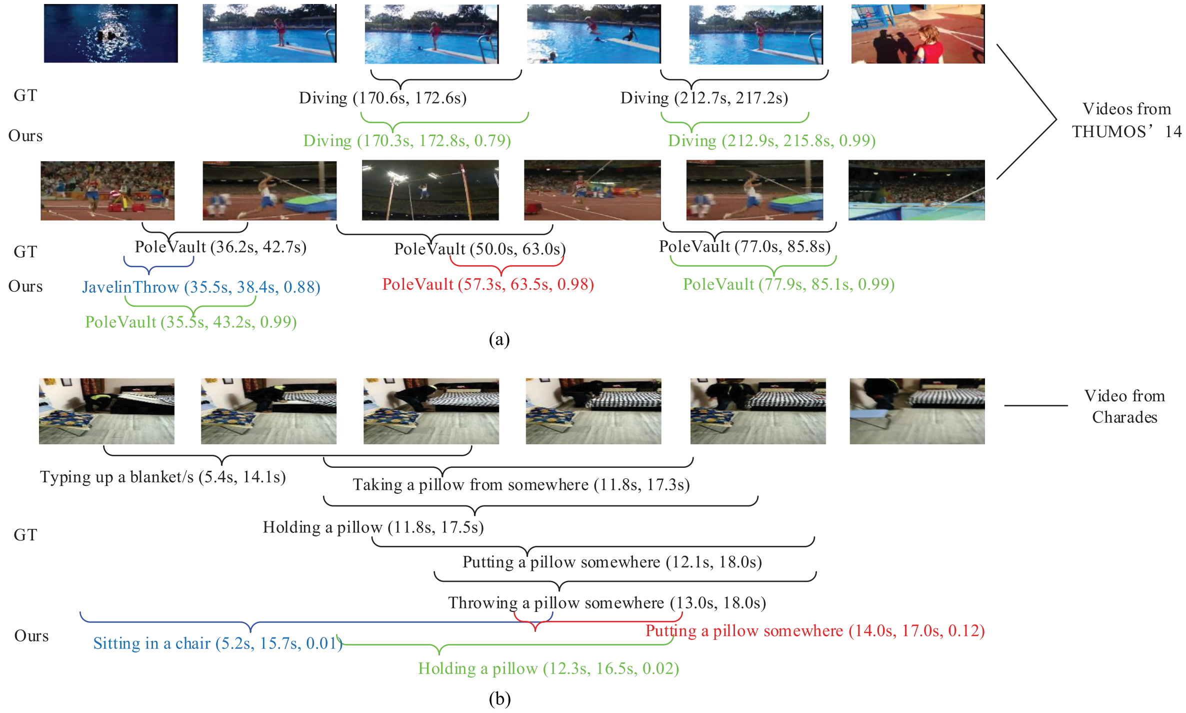

GT
Ours
Diving (170.6s, 172.6s)
Diving (170.3s, 172.8s, 0.79)
Diving (212.7s, 217.2s)
Diving (212.9s, 215.8s, 0.99)
Videos from
THUMOS’ 14
GT
Ours
PoleVault (36.2s, 42.7s)
JavelinThrow (35.5s, 38.4s, 0.88)
PoleVault (35.5s, 43.2s, 0.99)
PoleVault (50.0s, 63.0s)
PoleVault (57.3s, 63.5s, 0.98)
PoleVault (77.0s, 85.8s)
PoleVault (77.9s, 85.1s, 0.99)
(a)
Video from
Charades
Typing up a blanket/s (5.4s, 14.1s)
Taking a pillow from somewhere (11.8s, 17.3s)
GT
Holding a pillow (11.8s, 17.5s)
Putting a pillow somewhere (12.1s, 18.0s)
Throwing a pillow somewhere (13.0s, 18.0s)
Ours
Sitting in a chair (5.2s, 15.7s, 0.01)
Putting a pillow somewhere (14.0s, 17.0s, 0.12)
Holding a pillow (12.3s, 16.5s, 0.02)
(b)

Fig. 5. Visualization of the predicted actions by MSCA-Net (best viewed in color). (a) The results of two videos from THUMOS’14 dataset. (b) The results of one video from Charades dataset. Ground-truth (GT) action segments are marked in black. Predicted action segments are marked in green for correct predictions and in red and blue for wrong ones. Predicted action segments with IoU ≥ 0.5 are considered as correct. Corresponding start-end times and confidence score are shown inside brackets.

Investigation on using different feature volume sizes: In the 3D RoI-Pooling$^+$ step, regions with different temporal length, say $l \times h \times w$, are transformed into the same size volume ($l_s \times h_s \times w_s$). How to set the size volume is an important factor for describing the spatio-temporal information of the candidate proposal. To directly adopt the pretrained parameters of FC6 layer of R-C3D [41] model, we choose $1 \times 4 \times 4$, $16 \times 1 \times 1$ and $4 \times 2 \times 2$ for comparison because they have the identical number of parameters. In Table V, we explore the influence when different output feature volume sizes are adopted. The feature

TABLE III
THE PER-CLASS AP AT IOU THRESHOLD $\alpha = 0.5$ ON THUMOS'14 DATASET (%)

category	S-CNN [26]	R-C3D [41]	MSCA-Net
BaseballPitch	14.9	**26.1**	25.4
HammerThrow	19.1	43.2	**54.0**
BasketballDunk	20.1	54.0	**67.5**
HighJump	20.0	30.9	**42.5**
Billiards	7.6	**8.3**	7.8
JavelinThrow	18.2	47.0	**50.4**
CleanAndJerk	24.8	27.9	**55.7**
LongJump	34.8	57.4	**73.2**
CliffDiving	27.5	42.9	**74.4**
PoleVault	32.1	42.7	**73.7**
CricketBowling	15.7	30.6	**38.6**
Shotput	12.1	19.4	**21.5**
CricketShot	13.8	10.9	**20.1**
SoccerPenalty	19.2	15.8	**31.5**
Diving	17.6	26.2	**76.8**
TennisSwing	19.3	16.6	**26.6**
FrisbeeCatch	15.3	**20.1**	11.2
ThrowDiscus	24.4	**29.2**	28.4
GolfSwing	18.2	16.1	**43.8**
VolleyballSpiking	4.6	5.6	**14.1**
mAP($\alpha = 0.5$)	19.0	28.9	**41.8**

TABLE IV
THE MAP RESULTS OF OUR PROPOSED MSCA-NET MODEL ON THUMOS'14 DATASET (%) WITH INTEGRATING MULTIPLE CONTEXT REGIONS AT VARIOUS IOU THRESHOLD α

S	mAP(α=0.3)	mAP(α=0.5)	mAP(α=0.7)
R-C3D (baseline)	50.4	29.4	10.0
1.0	54.0	37.7	15.7
1.0+0.8	56.9	41.0	18.3
1.0+1.5	55.8	40.1	17.5
1.0+2.1	56.9	39.9	17.9
1.0+0.8+1.5	56.1	41.2	18.5
1.0+0.8+2.1	57.9	41.0	18.5
1.0+1.5+2.1	58.8	41.0	18.3
1.0+0.8+1.5+2.1	58.4	41.8	18.7

TABLE V
THE MAP RESULTS OF OUR PROPOSED MSCA-NET MODEL ON THUMOS'14 DATASET (%) WITH DIFFERENT FEATURE VOLUME SIZES AT VARIOUS IOU THRESHOLD α

	$(l_s \times h_s \times w_s)$		
	$(1 \times 4 \times 4)$	$(16 \times 1 \times 1)$	$(4 \times 2 \times 2)$
mAP(α=0.3)	52.9	58.0	58.4
mAP(α=0.5)	31.3	40.7	41.8
mAP(α=0.7)	12.2	18.5	18.7

volume with size $4 \times 2 \times 2$ performs the best, while $16 \times 1 \times 1$ outperforms the $1 \times 4 \times 4$. It demonstrates that the temporal information may be more important compared to spatial information in action detection. As a spatio-temporal tradeoff, we set the size of the feature volume as $4 \times 2 \times 2$.

Moreover, the results of MSCA-Net with feature volume size $1 \times 4 \times 4$ is better than the R-C3D (baseline), also demonstrating the effectiveness of combination with different context regions (investigation in the previous subsection, Table IV).

TABLE VI
THE MAP RESULTS OF OUR PROPOSED MSCA-NET MODEL ON THUMOS'14 DATASET (%) WITH THE OPTIONAL CNN BLOCKS AT VARIOUS IOU THRESHOLD α

	Optional CNN Blocks		
	0	1	2
mAP(α=0.3)	58.4	55.2	55.4
mAP(α=0.5)	41.8	37.1	37.6
mAP(α=0.7)	18.7	16.1	16.2

Investigation on the optional CNN blocks: Here, we investigate the optional CNNs (in Eqs. (2−5)) used to get high-level context features. The conv filters having a spatial size of $3 \times 3 \times 3$, a padding of 2, following with a RELU function, is considered as a CNN block. The results of our MSCA-Net with different number of the optional CNN blocks are shown in Table VI. We can find that, with the increasing of the number of CNN blocks, there are no improvements on the detection results. Instead, the performance declines to varying degrees. It's probably because that more parameters make the additional filter harder to learn without pretrained weight to initialize. Therefore, in our experiments, the optional CNNs (w_s and b_s, in Eqs. (2−5)) are not added.

Investigation on gate function: Gate function is introduced to adopt message passing for individual temporal regions. To evaluate the effectiveness of the proposed gate function in our MSCA-Net, we conducted the following experiments.

1) Without gate function (corresponding to Eq. (8)), all the candidate regions pass messages in the same way, which is denoted as v0.
2) Since an action could be divided into many "sub-actions", we design gate functions for different "sub-action" to get different magnitude of context features (corresponding to Eq. (11)), which is denoted as v3 (ours).
3) Moreover, v1 denotes the gate function that has the same influences for all the "sub-actions", by implementing the Eq. (10) with average-pooling.
4) In addition, we respectively get different gate functions for each scale, and generate each gate function from combination of its own scale and the normal scale, which is denoted as v2. The final refined RoI features for a proposal can be calculated as,

$$\mathbf{h}_y = f^{1.0} + \sum_{i=0.8,1.5,2.1} \gamma^i \mathbf{h}_x^i \sigma\big(conv\,(cat(h^i, h^{1.0}))\big), \tag{13}$$

where cat is the concatenation operation, $conv$ is the convolution operation, $\sigma(x) = 1/[1 + \exp(-x)]$ is the element-wise sigmoid function. We set $\gamma^i = 1$, for $i = 0.8, 1.5, 2.1$ in experiments.

As we can see from Table VII, v3, our proposed gate function, obtains the highest mAP at all the threshold α, which demonstrates the effectiveness of our gate function structure for automatically controlling the message passing.

Investigation on messages with different magnitudes: As mentioned in Section III-D, the magnitude of the messages (γ in Eq.

TABLE VII
THE MAP RESULTS OF OUR PROPOSED MSCA-NET MODEL ON THUMOS'14 DATASET (%) WITH DIFFERENT GATE FUNCTION AT VARIOUS IOU THRESHOLD α

	Gate Function			
	v0	v1	v2	v3 (ours)
mAP(α=0.3)	54.2	56.6	55.0	58.4
mAP(α=0.5)	38.2	38.9	39.2	41.8
mAP(α=0.7)	16.3	17.4	17.9	18.7

TABLE VIII
THE MAP RESULTS OF OUR PROPOSED MSCA-NET MODEL ON THUMOS'14 DATASET (%) WITH DIFFERENT MESSAGE MAGNITUDES γ AT VARIOUS IOU THRESHOLD α

	γ				
	0.1	0.2	0.3	0.5	0.8
mAP(α=0.3)	58.4	58.4	57.5	57.3	58.0
mAP(α=0.5)	41.7	41.8	41.9	39.6	41.0
mAP(α=0.7)	17.8	18.7	18.3	17.9	18.6

TABLE IX
THE DETECTION RESULTS ON THUMOS'14 DATASET (%), WHEN USING DIFFERENT FUSION METHODS TO INTEGRATE DIFFERENT CONTEXT REGIONS. THE ROI-POOLED FEATURE VOLUME SIZE IS SET $1 \times 4 \times 4$ FOLLOWING THE BASELINE R-C3D [41]

Model	0.3	0.5	0.7
R-C3D (baseline)	50.4	29.4	10.0
$F^{sum}(f^{1.0}, f^{0.8})$	51.3	31.1	11.8
$F^{sum}(f^{1.0}, f^{1.5})$	46.2	25.4	8.0
$F^{max}(f^{1.0}, f^{0.8})$	50.1	29.2	10.2
$F^{max}(f^{1.0}, f^{1.5})$	42.4	21.5	5.8
$F^{cat-conv}(f^{1.0}, f^{0.8})$	54.7	35.0	13.7
$F^{cat-conv}(f^{1.0}, f^{1.5})$	53.2	34.8	14.3

(11)) from other context features influences the detection accuracy. Table VIII shows the experimental results for messages with different magnitudes γ. It can be seen that the performance is stable when γ varies at a reasonable range. We choose $\gamma = 0.2$ for all the experiments.

Investigation on other integration methods: How to integrate those features of candidate temporal proposal and context regions with different scale values S is import to generate valuable features. In this section, we further explore some other integration methods to fuse the features of those multi-scale temporal regions. Our proposed $F^{cat-conv}$ method (Eqs. (6−7)) concatenates the original proposal features and context features across feature channels, subsequently following by convolution. For comparison, we also use other two methods, F^{sum} and F^{max}, which compute the sum and maximum of feature maps at the same temporal-spatial location, respectively. Based on R-C3D framework, we conduct experiments with different integration methods. Table IX summarizes the results. Surprisingly, not all the integration methods improve the baseline R-C3D detector on THUMOS'14 dataset. Specifically, the capability of fusing context features ($S = 0.8, 1.5$) of the integration methods are generally in line with the relationship: $F^{cat-conv} > F^{sum} > F^{max}$. $F^{cat-conv}$ outperforms the baseline model, while F^{sum} and F^{max} is slightly better or even worse than the baseline model. We conjecture the reason may be the differences between integration methods. F^{max} introduces a local competition between original region and context region. If either side fails in the competition, we will lose the corresponding feature information. F^{sum} simply defines an element-wise adding operation. It can not dynamically adjust the contribution for the fused feature, while $F^{cat-conv}$ combines original feature and context feature via the learnable parameters.

The detection results analysis of context information: To further investigate how context regions with different scales influence the detection results, we count all kinds of detection results at 70% average recall rate and show the percentage of detections in Fig. 4. All the results are based on the R-C3D baseline model with different context information integration by our $F^{cat-conv}$ method. Because of space limitations, we present the pie charts only for three representative classes. “FrisbeeCatch”: with original candidate proposal ($S = 1.0$) achieving the highest AP; “HammerThrow”: integrating with more context ($S = 1.0 + 1.5$) achieving the highest AP; while “PoleVault”: integrating with more local details ($S = 1.0 + 0.8$) achieving the highest AP.

From the statistic data we can see that, the model with more local details can reduce localization error ($S = 1.0$(baseline): 17%, 45%, 51% vs. $S = 1.0 + 0.8$: 16%, 40%, 30%), while lacking the ability of handling classification error (Other+Background). The model with more context is also can contribute to reducing the localization error ($S = 1.0$(baseline): 17%, 45%, 51% vs. $S = 1.0 + 1.5$: 16%, 44%, 47%) while it is inferior to the model with more local details. Therefore, the contribution of those context information inside (or outside) a proposal for action detection among different action classes is uncertain, which depends on both of the proposals and the action instances.

B. Charades

Charades [28] dataset is introduced for action classification and detection, which mainly focus on daily dynamic scenes. In total, the dataset consists of 7,985 train and 1,863 test videos with an average length of 30s. It provides 66,500 temporally localized intervals for 157 action classes. The videos are recorded by Amazon Mechanical Turk users based on provided scripts. The dataset is challenging because of low illumination, diversity and casual nature of the videos, etc. But the most challenge of this dataset is the abundance of overlapping activities, sometimes multiple activities having exactly the same start and end times (typical examples include pairs of activities like “holding a phone” and “playing with a phone” or “holding a towel” and “tidying up a towel”). Following the setup of R-C3D [41], we finetune the Sports-1M pretrained C3D model on the Charades training set at the same 5 fps and initialize the 3D ConvNet part of our model with these finetuned weights.

The evaluation metric estimates the prediction performance on 25 equidistant frames by making a multi-label prediction for each of these frames, while the action localization result is

TABLE X
THE mAP RESULTS ON CHARADES DATASET (%). WE REPORT THE RESULTS USING THE SAME EVALUATION METRICS AS IN [27]

	mAP
Random [27]	4.2
RGB [27]	8.8
Two-Stream [27]	10.0
Two-Stream+LSTM [27]	8.8
Sigurdsson et al. [27]	12.1
R-C3D [41]	12.7
MSCA-Net(Ours)	**14.4**

TABLE XI
THE mAP ($\alpha = 0.5$) RESULTS ON ACTIVITYNET DATASET (%). WE REPORT THE RESULTS ON THE VALIDATION SET

	mAP
UPC [21]	22.5
R-C3D [41]	26.8
MSCA-Net (Ours)	**30.2**
Singh et. al. [30]	34.5
TCN [6]	36.2
TAL-Net [4]	38.2
SCC [3]	40.0
SSN [47]	43.3
CDC [25]	45.3
BSN [19]	52.5

reported in terms of mAP metric on these frames. For a fair comparison, we map our action segment prediction to 25 equidistant frames and the evaluation uses the same mAP evaluation metric.

As shown in Table X, our model outperforms the R-C3D model [41] and the asynchronous temporal fields model proposed in [27]. The presence of a large number of temporally overlapping actions is one of the major challenges of this dataset. The results show that our MSCA-Net model is capable of handling such scenarios.

C. ActivityNet

The ActivityNet v1.3 [8] dataset consists of untrimmed videos, which has 10024, 4926 and 5044 videos containing 200 different types of actions in the train, validation and test sets respectively. Most videos contain activity instances of a single class covering a great deal of the video. Compared to THUMOS'14, this is a large-scale dataset both in terms of the number of actions involved and the amount of videos. Following the setup of R-C3D [41], we finetuned the Sports-1M pretrained C3D model on the ActivityNet training set, and initialize the 3D ConvNet part of our model with these finetuned weights. Then we report the action detection results on the validation set.

Table XI shows the results of our MSCA-Net method and some other recent published results. Among them, UPC [21], R-C3D [41] and our MSCA-Net use only C3D features. Our proposed MSCA-Net performs better than UPC and R-C3D. The 3.4% (= 30.2% – 26.8%) improvement of MSCA-Net compared to R-C3D, demonstrates the effectiveness of automatically extracting valuable context features from multiple scales of each candidate proposal. However, the other methods whether need perform more object detection tasks (SCC [3]) or are based on the two-stream CNN framework (TCN [6], TAL-Net [4], SSN [47], BSN [19]). Moreover, compared to THUMOS'14, ActivityNet maybe not the better choice for evaluating action localization [4]. THUMOS'14 has more action instances per video and each video contains a larger portion of background: on average, the THUMOS'14 training set has 15 instances per video and each video has 71% background, while the ActivityNet training set has only 1.5 instances per video and each video has only 36% background.

TABLE XII
THE ACTION DETECTION SPEED AND STORAGE REQUIREMENT AT INFERRING STAGE

	Speed (fps)	Storage Requirement (MB)
S-CNN [26]	60	936
R-C3D [41]	1030	303
MSCA-Net(Ours)	950	500

D. Efficiency Analysis

In this section, we compare our MSCA-Net model with two others, S-CNN [26] and R-C3D [41], in terms of detection speed and storage requirement. S-CNN [26] uses a time-consuming sliding window strategy. R-C3D [41] constructs the proposal and classification pipeline in an end-to-end fashion and these two stages share the features. Compared to R-C3D, our MSCA-Net additionally uses the two-branch structure (in Fig. 3) to automatically extract context features from multiple temporal regions of each candidate proposal. As shown in Table XII, our MSCA-Net model is also efficient in both speed and storage. The result is obtained using single Titan X Pascal GPU for testing.

E. Visualization of Action Detection

To visually show the performance of our MSCA-Net for action detection, we list the predicted results in Fig. 5. Each consists of a sequence of frames sampled from a full test video, the ground-truth (GT) are marked in black. It shows all kinds of detection cases. The correct predictions, those intervals with correct action labels and the IoU ≥ 0.5, are marked in green.

There are two wrong prediction cases. Case 1: poor localization, those intervals predicted with wrong start or end time (e.g. PoleVault (57.3 s, 63.5 s, 0.98) in (a) and Putting a pillow somewhere (14.0 s, 17.0 s, 0.12) in (b)), are marked in red. Case 2: wrong classification, those intervals predicted with wrong labels (e.g. JavelinThrow (35.5 s, 38.4 s, 0.88) in (a) and Sitting in a chair (5.2 s, 15.7 s, 0.01) in (b)), are marked in blue.

V. CONCLUSIONS

In this paper, we present a novel multiple scales based context-aware net (MSCA-Net) for action detection based on the framework of R-C3D. MSCA-Net aims to automatically extract valuable context features from multi-scale temporal regions of each candidate action proposal. The proposed MSCA-Net specially utilizes a two-branch structure to capture context information inner and outer the candidate proposal, one branch for generating the multiple scaled context features of each action proposal and the other branch for controlling message passing with the the context-aware gate function. Through extensive experiments on three action detection datasets, we evaluated the effectiveness of our designed two-branch structure and explored how these context features influence the detection results.

REFERENCES

[1] S. Bell, C. Lawrence Zitnick, K. Bala, and R. Girshick, "Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks," in *Proc. Comput. Vis. Pattern Recognit.*, 2016, pp. 2874–2883.

[2] S. Buch, V. Escorcia, C. Shen, B. Ghanem, and J. C. Niebles, "Sst: Single-stream temporal action proposals," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, pp. 6373–6382.

[3] F. Caba Heilbron, W. Barrios, V. Escorcia, and B. Ghanem, "Scc: Semantic context cascade for efficient action detection," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, pp. 1454–1463.

[4] Y.-W. Chao *et al.*, "Rethinking the faster R-CNN architecture for temporal action localization," in *Proc. Comput. Vis. Pattern Recognit.*, 2018, pp. 1130–1139.

[5] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

[6] X. Dai, B. Singh, G. Zhang, L. S. Davis, and Y. Q. Chen, "Temporal context network for activity localization in videos," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 5727–5736.

[7] V. Escorcia, F. C. Heilbron, J. C. Niebles, and B. Ghanem, "Daps: Deep action proposals for action understanding," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 768–784.

[8] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *Proc. Comput. Vis. Pattern Recognit.*, 2015, pp. 961–970.

[9] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proc. Comput. Vis. Pattern Recognit.*, 2016, pp. 1933–1941.

[10] J. Gao, Z. Yang, and R. Nevatia, "Cascaded boundary regression for temporal action detection," 2017, arXiv:1705.01180.

[11] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware CNN model," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 1134–1142.

[12] R. Girshick, "Fast R-CNN," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[13] D. Guo, W. Li, and X. Fang, "Fully convolutional network for multiscale temporal action proposals," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3428–3438, Dec. 2018.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[15] J. Hou, X. Wu, Y. Sun, and Y. Jia, "Content-attention representation by factorized action-scene network for action recognition," *IEEE Trans. Multimedia*, vol. 20, no. 6, pp. 1537–1547, Jun. 2018.

[16] Y.-G. Jiang *et al.*, "THUMOS challenge: Action recognition with a large number of classes," 2014. [Online]. Available: http://crcv.ucf.edu/THUMOS14/

[17] H. Kuehne, H. Jhuang, R. Stiefelhagen, and T. Serre, "Hmdb51: A large video database for human motion recognition," in *High Performance Computing in Science and Engineering'12*. Berlin, Germany: Springer, 2013, pp. 571–582.

[18] T. Lin, X. Zhao, and Z. Shou, "Single shot temporal action detection," in *Proc. 25th ACM int. conf. Multimedia*, 2017, pp. 988–996.

[19] T. Lin, X. Zhao, H. Su, C. Wang, and M. Yang, "Bsn: Boundary sensitive network for temporal action proposal generation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[20] W. Liu *et al.*, "Ssd: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.

[21] A. Montes, A. Salvador, S. Pascual, and X. Giro-i Nieto, "Temporal activity detection in untrimmed videos with recurrent neural networks," 2016, *arXiv:1608.08128*.

[22] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 807–814.

[23] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3D residual networks," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 5534–5542.

[24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2016.

[25] Z. Shou, J. Chan, A. Zareian, K. Miyazawa, and S.-F. Chang, "Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, pp. 1417–1426.

[26] Z. Shou, D. Wang, and S.-F. Chang, "Temporal action localization in untrimmed videos via multi-stage CNNs," in *Proc. Comput. Vis. Pattern Recognit.*, 2016, pp. 1049–1058.

[27] G. A. Sigurdsson, S. Divvala, A. Farhadi, and A. Gupta, "Asynchronous temporal fields for action recognition," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, vol. 2, pp. 585–594.

[28] G. A. Sigurdsson *et al.*, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 510–526.

[29] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 568–576.

[30] G. Singh and F. Cuzzolin, "Untrimmed video classification for activity detection: Submission to activitynet challenge," 2016, *arXiv:1607.01979*.

[31] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," 2012, *arXiv:1212.0402*.

[32] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 4489–4497.

[33] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *Int. J. Comput. Vis.*, vol. 103, no. 1, pp. 60–79, 2013.

[34] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 3551–3558.

[35] L. Wang, Y. Qiao, and X. Tang, "Latent hierarchical model of temporal structure for complex activity classification," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 810–822, Feb. 2014.

[36] L. Wang *et al.*, "Temporal segment networks: Towards good practices for deep action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 20–36.

[37] S. Wang, J. Cheng, H. Liu, F. Wang, and H. Zhou, "Pedestrian detection via body part semantic and contextual information with DNN," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3148–3159, Nov. 2018.

[38] X. Wang, L. Gao, P. Wang, X. Sun, and X. Liu, "Two-stream 3-D convnet fusion for action recognition in videos with arbitrary size and length," *IEEE Trans. Multimedia*, vol. 20, no. 3, pp. 634–644, Mar. 2018.

[39] Y. Wang, M. Long, J. Wang, and P. S. Yu, "Spatiotemporal pyramid network for video action recognition," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, pp. 2097–2106.

[40] Y. Xiong, Y. Zhao, L. Wang, D. Lin, and X. Tang, "A pursuit of temporal accuracy in general activity detection," 2017, *arXiv:1703.02716*.

[41] H. Xu, A. Das, and K. Saenko, "R-c3D: Region convolutional 3D network for temporal activity detection," in *Proc. Int. Conf. Comput. Vis.*, 2017, vol. 6, pp. 5783–5792.

[42] K. Yang, P. Qiao, D. Li, S. Lv, and Y. Dou, "Exploring temporal preservation networks for precise temporal action localization," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 7477–7484.

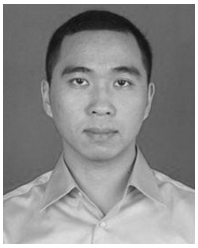
[43] J. Yuan, B. Ni, X. Yang, and A. A. Kassim, "Temporal action localization with pyramid of score distribution features," in *Proc. Comput. Vis. Pattern Recognit.*, 2016, pp. 3093–3102.

[44] Z. Yuan, J. C. Stroud, T. Lu, and J. Deng, "Temporal action localization by structured maximal sums," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, pp. 3684–3692.

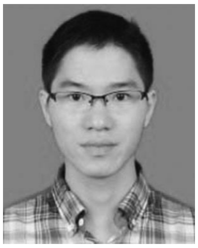
[45] J. Y.-H. Ng *et al.*, "Beyond short snippets: Deep networks for video classification," in *Proc. Comput. Vis. Pattern Recognit.*, 2015, pp. 4694–4702.

[46] X. Zeng *et al.*, "Crafting GBD-net for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 9, pp. 2109–2123, Sep. 2018.

[47] Y. Zhao *et al.*, "Temporal action detection with structured segment networks," in *Proc. Int. Conf. Comput. Vis.*, 2017, vol. 8, pp. 2933–2942.



Haijun Liu received the B.Eng. and M.Eng. degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 2011 and 2014, respectively. He is currently working toward the Ph.D. degree with the School of Information and Communication Engineering, University of Electronic Science and Technology of China. His main research interests include manifold learning, metric learning, deep learning, subspace clustering and sparse representation in computer vision and machine learning, with focuses on human action detection and recognition, face detection and recognition, person detection and re-identification, and remote sensing image processing. He was a Reviewer for many journals, including IEEE TIP, TMM, TCSVT, SPL, etc., and also the Editorial Board member of *Journal of Artificial Intelligence and Systems*.



Shiguang Wang received the bachelor's degree from Southwest Jiaotong University, Chengdu, China, in 2016. He is currently working toward the master's degree with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, China. His research interests include computer vision and deep learning.

Wen Wang received the B.E. degree in electronic and information engineering from the Harbin University of Commerce, Harbin, China, in 2016. She is currently working toward the Ph.D. degree in signal and information processing from the University of Electronic Science and Technology of China, Chengdu, China. Her research interests include human action recognition and video analysis.

Jian Cheng received the Ph.D. degree in pattern recognition and intelligent system from Shanghai Jiao Tong University, Shanghai, China, in 2006. From 2006 to 2007, he was an Assistant Researcher with the Chengdu Information Technology of Chinese Academy of Sciences Co., Ltd. He is currently a Professor with the School of Information and Communication Engineering, University of Electronic Science and Technology of China. His main research interests include machine learning, computer vision, remote sensing image analysis, multimodal image classification, video surveillance and scene understanding, human behavior analysis, etc.