Adaptive Two-Stream Consensus Network for Weakly-Supervised Temporal Action Localization

Yuanhao Zhai, *Member, IEEE,* Le Wang, *Senior Member, IEEE,* Wei Tang, *Member, IEEE,* Qilin Zhang, *Member, IEEE,* Nanning Zheng, *Fellow, IEEE,* David Doermann, *Fellow, IEEE,* Junsong Yuan, *Fellow, IEEE,* and Gang Hua, *Fellow, IEEE*

Abstract—Weakly-supervised temporal action localization (W-TAL) aims to classify and localize all action instances in untrimmed videos under only video-level supervision. Without frame-level annotations, it is challenging for W-TAL methods to clearly distinguish actions and background, which severely degrades the action boundary localization and action proposal scoring. In this paper, we present an adaptive two-stream consensus network (A-TSCN) to address this problem. Our A-TSCN features an iterative refinement training scheme: a frame-level pseudo ground truth is generated and iteratively updated from a late-fusion activation sequence, and used to provide frame-level supervision for improved model training. Besides, we introduce an adaptive attention normalization loss, which adaptively selects action and background snippets according to video attention distribution. By differentiating the attention values of the selected action snippets and background snippets, it forces the predicted attention to act as a binary selection and promotes the precise localization of action boundaries. Furthermore, we propose a video-level and a snippet-level uncertainty estimator, and they can mitigate the adverse effect caused by learning from noisy pseudo ground truth. Experiments conducted on the THUMOS14, ActivityNet v1.2, ActivityNet v1.3, and HACS datasets show that our A-TSCN outperforms current state-of-the-art methods, and even achieves comparable performance with several fully-supervised methods.

Index Terms—Temporal action localization, weakly-supervised learning.

1 Introduction

The task of weakly-supervised temporal action localization (W-TAL) aims at simultaneously localizing and classifying all action instances in a long untrimmed video given only video-level categorical labels in the learning phase. Compared to its fully-supervised counterpart, which requires frame-level annotations of all action instances during training, W-TAL greatly simplifies the procedure of data collection and avoids annotation bias of human annotators, and therefore has been widely studied [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14] in recent years.

Several previous W-TAL methods [2], [4], [5], [6], [9], [10], [11], [12], [13], [14] adopt a multiple instance learning (MIL) framework, where a video is treated as a bag of snippets to perform video-level action classification. During testing, the trained model slides over time and generates a temporal-class activation map (T-CAM) [4], [15] (*i.e.*, a sequence of probability distributions over action classes at each time step) and an attention sequence that measures the relative importance of each snippet. The action proposals are generated by thresholding the attention value and/or the T-CAM. This MIL framework is usually built on two feature modalities, *i.e.*, RGB frames and optical flow, which are fused in two mainstream ways. Early-fusion methods [3], [5], [6], [8], [12] concatenate the RGB and optical flow features before they are fed into the network, and late-fusion methods [4], [6], [9], [10] compute a weighted sum of their respective outputs before generating action proposals. An example of late fusion is shown in Fig. 1.

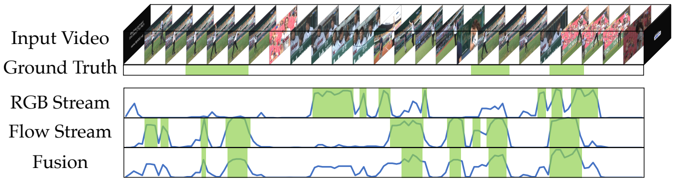


Fig. 1. Visualization of two-stream outputs and their late-fusion result. The five rows show the input video, the ground truth action instances and attention sequences (scaled from 0 to 1) predicted by the RGB stream, the flow stream and their weighted sum (*i.e.*, the fusion result), respectively. The horizontal and vertical axes denote the time and the intensity of attention values, respectively. The green boxes denote the localization results generated by thresholding the attention at the value of 0.5. By properly combining the two different attention distributions predicted by the RGB and flow streams, the late-fusion result achieves better localization performance.

Despite these recent developments, one major challenge remains to be solved: the lack of frame-level supervision makes W-TAL methods hard to distinguish action from the background clearly. This problem degrades the localization performance in two major ways. First, the detected

- *Y. Zhai, L. Wang, and N. Zheng are with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China. Y. Zhai is also with the Computer Science and Engineering Department, State University of New York at Buffalo, Buffalo, NY 14260, USA. E-mail: yzhai6@buffalo.edu, {lewang,nnzheng}@mail.xjtu.edu.cn. (Corresponding author: Le Wang.)*
- *W. Tang is with the Department of Computer Science, University of Illinois at Chicago, Chicago, IL 60607, USA. E-mail: tangw@uic.edu.*
- *Q. Zhang is an independent researcher, Menlo Park, CA 94025, USA. E-mail: samqzhang@gmail.com.*
- *D. Doermann and J. Yuan are with the Computer Science and Engineering Department, State University of New York at Buffalo, Buffalo, NY 14260, USA. E-mail: {doermann,jsyuan}@buffalo.edu.*
- *G. Hua is with Wormpex AI Research, Bellevue, WA 98004, USA. E-mail: ganghua@gmail.com.*

action instance may not necessarily correspond to the video-level labels, *e.g.*, the detector may falsely recognize frames including a pool as the swimming action. Second, the ambiguity between actions and background will influence the activation sequences. This not only makes thresholding methods generate incomplete or over-complete action proposals but also leads to unreliable action proposal confidence scores. Therefore, it is necessary to exploit more fine-grained supervision to guide the learning process.

In this paper, we introduce an adaptive two-stream consensus network (A-TSCN) to address this problem. First, we present an adaptive attention normalization loss to better differentiate actions and background. Inspired by Otsu's method in image binarization [16], the adaptive attention normalization loss automatically distinguishes the action snippets and the background snippets according to the video attention distribution. By maximizing the difference between attention values of the action snippets and background snippets, the adaptive attention normalization loss promotes precise localization of action boundaries. Besides, inspired by two-stream late fusion [17], we introduce a frame-level pseudo ground truth to provide more fine-grained supervision. As shown in Fig. 1, with a proper fusion parameter (*e.g.*, the hyperparameter controlling the relative importance of the two modalities), the late-fusion activation sequence is of higher quality compared with each single stream. Therefore, we propose to generate a frame-level pseudo ground truth based on the late-fusion activation sequence, which is then used to iteratively refine the two-stream base models. To alleviate the adverse effect caused by learning from noisy pseudo labels, we propose a video-level and a snippet-level uncertainty estimator. They respectively compute a video-level confidence score and a snippet-level confidence score for the pseudo labels based on the agreement of two-stream outputs. By applying larger weights to confident pseudo labels and smaller weights to ambiguous pseudo labels, the model can avoid learning from possibly wrong pseudo labels, and gradually generate more precise pseudo labels.

Given an input video, snippet-level features are first extracted with pre-trained backbones from RGB frames and optical flow, respectively. Then the two-stream base models are trained with video-level labels on RGB and optical flow features, respectively, where the adaptive attention normalization loss is used to learn the attention distribution. After obtaining two-stream attention sequences, a frame-level pseudo ground truth is generated based on their weighted sum (*i.e.*, the late-fusion attention sequence). Meanwhile, the video-level uncertainty estimator and the snippet-level uncertainty estimator compute the pseudo label confidence given two-stream outputs. The pseudo ground truth in turn provides frame-level supervision to improve the two-stream base models. We iteratively update the pseudo ground truth and refine the two-stream base models, where the adaptive attention normalization loss simultaneously forces the predicted attention to act as a binary selector. The final localization result is obtained by thresholding the late-fusion attention sequence.

To summarize, our contribution is threefold:

- We introduce an adaptive two-stream consensus network (A-TSCN) for W-TAL. The proposed A-TSCN features an iterative refinement training method. The pseudo ground truth generated from the late-fusion attention sequence at the previous iteration can provide more precise frame-level supervision at the current iteration, and iteratively refine base models. In addition, we propose a video-level uncertainty estimator and a snippet-level uncertainty estimator to mitigate the adverse effect caused by learning noisy pseudo ground truth.
- We propose an adaptive attention normalization loss to differentiate actions and background. The proposed loss function adaptively distinguishes the action snippets and the background snippets based on the video attention distribution, leading to more training signals. The adaptive attention normalization loss promotes precise action boundary localization and accurate action proposal scoring.
- Extensive experiments are conducted on four datasets (*i.e.*, THUMOS14, ActivityNet v1.2, ActivityNet v1.3, and HACS) to demonstrate the effectiveness of the proposed method. Our A-TSCN significantly outperforms previous state-of-the-art W-TAL methods, and even achieves comparable performance to some recent fully-supervised TAL methods.

We note a conference version of this paper appears in [18]. This paper extends our previous version in three significant aspects.

- We improve the original attention normalization loss by adaptively selecting the action snippets and the background snippets according to the attention distribution for each video, rather than using a fixed portion. The improved adaptive attention normalization loss provides more training signals and improve the performance.
- To mitigate the adverse effect caused by learning from noisy pseudo labels, we introduce a video-level uncertainty estimator and a snippet-level uncertainty estimator. They estimate the confidence scores for the pseudo labels at the video level and snippet level, respectively, thus reducing the weights of possibly wrong pseudo labels.
- More ablation studies are conducted to validate the effectiveness of the proposed method. In addition, we compare it with more state-of-the-art methods, and also include comparison on the new HACS dataset [19]. The results reveal that the proposed A-TSCN outperforms previous state-of-the-art methods on all benchmarks.

This paper is organized as follows: Section 2 briefly reviews related work. We present the technical details of the proposed method in Section 3. Experimental results and discussions are presented in Section 4. Finally, we conclude in Section 5.

2 Related Work

We briefly review related work in action recognition, fully-supervised temporal action localization, weakly-supervised temporal action localization and self-training.

Action recognition. Traditional methods [20], [21], [22], [23] aim to model spatio-temporal information via hand-crafted features. Recently, Two-Stream Convolutional Networks [17] use two separate Convolutional Neural Networks (CNNs)

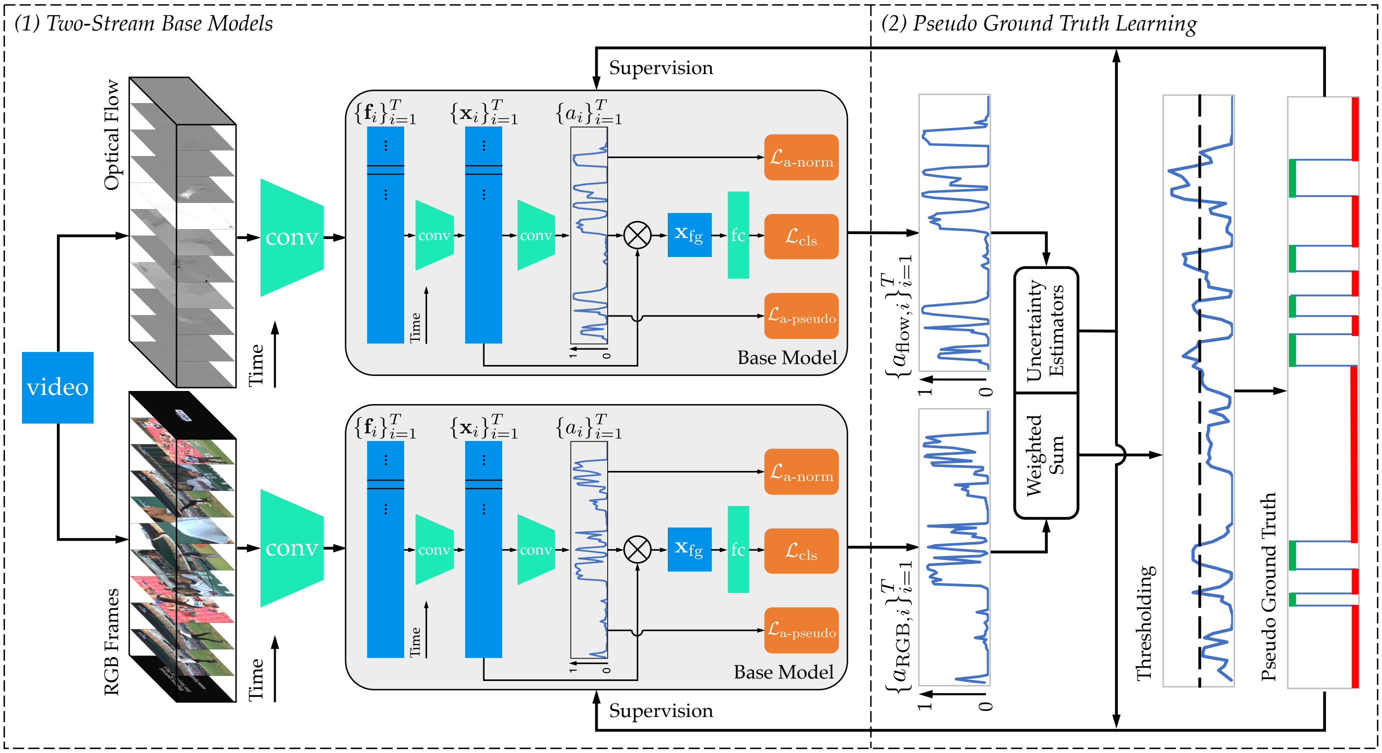

(1) Two-Stream Base Models
(2) Pseudo Ground Truth Learning
Supervision
Optical Flow
Time
video
conv
RGB Frames
Time
$\{\mathbf{f}_i\}_{i=1}^T$
$\{\mathbf{x}_i\}_{i=1}^T$
$\{a_i\}_{i=1}^T$
conv
conv
Time
1
0
$\mathbf{x}_{\text{fg}}$
fc
$\mathcal{L}_{\text{a-norm}}$
$\mathcal{L}_{\text{cls}}$
$\mathcal{L}_{\text{a-pseudo}}$
Base Model
$\{a_{\text{Flow},i}\}_{i=1}^T$
$\{a_{\text{RGB},i}\}_{i=1}^T$
Uncertainty Estimators
Weighted Sum
Thresholding
Pseudo Ground Truth
Supervision

Fig. 2. An overview of the proposed adaptive two-stream consensus network, which consists of two parts. (1) Two-stream base models, where RGB and optical flow snippet-level features are first extracted with pre-trained models, then action recognition is performed on the two modalities with two-stream base models, respectively. (2) Pseudo ground truth learning, where a frame-level pseudo ground truth is generated from the two-stream late-fusion attention sequence, along with video-level and snippet-level uncertainty estimators computing the confidence of the generated pseudo ground truth. The pseudo ground truth in turn provides frame-level supervision to two-stream base models.

to exploit appearance and motion clues from RGB frames and optical flow respectively, and use a late-fusion method to reconcile the two-stream outputs. Feichtenhofer *et al.* [24] focuses on studying different ways to fuse the two streams. The Inflated 3D ConvNet (I3D) [25] expands the 2D CNNs in two-stream convolutional networks to 3D CNNs, and further improves the performance. Several recent methods [26], [27], [28], [29], [30] focus on directly learning motion clues from RGB frames instead of calculating optical flow. Besides, some works [31], [32], [33], [34] also try to model long-term temporal information in videos.

Fully-supervised temporal action localization methods require frame-level annotations of all action instances during training. Several large-scale datasets have been created for this task, such as THUMOS [35], [36], ActivityNet [37], and Charades [38]. Many methods [39], [40], [41], [42], [43], [44], [45], [46] adopt a two-stage pipeline, *i.e.*, action proposal generation followed by action classification. Several methods [43], [44], [46], [47] adopt the Faster R-CNN [48] framework to TAL. Most recently, some methods [45], [49], [50] focus on generating action proposals with a more flexible duration. Several methods [51], [52], [53] apply the Graph Convolutional Networks (GCN) [54], [55] to TAL to incorporate more contextual information and exploit proposal-proposal relations. MS-TCN++ [56] proposes a smooth loss to address the over-segmentation error. Different from theirs, our smooth loss is proposed to smooth the attention sequence and remove fragmentary action proposals.

Weakly-supervised temporal action localization, which only requires video-level supervision during training, significantly reduces the data annotation efforts, and draws increasing attention from the community. Hide-and-Seek [1] randomly hides part of the input video to guide the network to discover other relevant parts. UntrimmedNet [2] consists of a selection module to select the important snippets and a classification module to perform per snippet classification. Sparse Temporal Pooling Network (STPN) [4] improves UntrimmedNet by adding a sparse loss to enforce the sparsity of selected segments. W-TALC [5] jointly optimizes a co-activity similarity loss and a multiple instance learning loss to train the network. AutoLoc [3] and CleanNet [8] adopt a two-stage pipeline, where they first generate initial action proposals, and then regress the action proposal boundaries based on prior knowledge: the action area should have higher activation than its surrounding background area. Liu *et al.* [6] propose a multi-branch network to model different stages of action. Besides, several methods [9], [12] focus on modeling the background to better differentiate actions and background. DGAM [14] proposes to separate action and context with a conditional Variational Auto-Encoder. A2CL-PT [57] uses two parallel branches in an adversarial way to generate complete action proposals. EM-MIL [58] also leverages pseudo labels, where the class-agnostic attention and the class-specific activation sequence are alternately trained to supervise each other.

Previously, RefineLoc [59] also proposes an iterative refinement framework to help the model capture a complete action instance. Our method is distinct from RefineLoc in three main aspects. (1) We adopt a late fusion framework while RefineLoc uses an early fusion framework. Note that the amount of parameters in the late fusion framework is only half of that in the early fusion framework, which also has the potential overfitting problem according to recent study [6], [60]. (2) Our pseudo ground truth is generated by fusing two-stream attention sequences, which provides better localization performance than individual streams, while RefineLoc generates the pseudo ground truth by expanding previous localization results, which might result in coarser and over-complete action proposals. (3) In addition to the classification loss, we also introduce an (adaptive) attention normalization loss, which explicitly avoids the ambiguity of attention, while RefineLoc does not have explicit constraints on attention values. As will be shown in Section 4.4, all three distinctions contribute to our performance superiority.

Self-training. In semi-supervised learning, self-training [61], [62], [63], [64], [65] is a widely-used training scheme, which mainly contains three steps: (1) train a student model with labeled data, (2) generate pseudo labels on unlabeled data with the trained model, and (3) train the student model with both labeled data and pseudo-labeled data. Our pseudo ground truth learning is similar to self-training by regarding each video snippet as a data point.

3 Proposed Method

In this section, we first formulate the task of weakly-supervised temporal action localization (W-TAL), and then describe the proposed adaptive two-stream consensus network (A-TSCN) in detail. As illustrated in Fig. 2, our A-TSCN consists of two parts, *i.e.*, two-stream base models and a pseudo ground truth generation module. Given an input video, two-stream base models are first used to perform action recognition on RGB snippets and optical flow snippets respectively, and get respective initial attention sequences. To facilitate action and background distinguishment, an adaptive attention normalization loss forces the attention to act like binary selection. Then, a frame-level pseudo ground truth is generated based on the late-fusion attention sequence, which in turn provides frame-level supervision to two-stream base models. Meanwhile, a video-level and a snippet-level uncertainty estimator dynamically compute the weights for the pseudo ground truth learning. Finally, the pseudo ground truth is iteratively updated and refines the base models.

3.1 Problem Formulation

Assume we are given a set of training videos. For each video, we only have its video-level categorical label $\mathbf{y}$, where $\mathbf{y} \in \mathbb{R}^C$ is a normalized multi-hot vector, and C is the number of action categories. The goal of temporal action localization is to detect a set of action instances $\{(t_s, t_e, c, \psi)\}$ for each testing video, where t_s, t_e, c, ψ denote the start time, the end time, the predicted action category, and the confidence score of the action instance, respectively.

3.2 Two-Stream Base Models

We follow recent W-TAL methods [3], [4], [5], [6], [8], [9], [10], [11], [12], [13], [14] to construct two-stream base models upon snippet-level feature sequences extracted from the raw video volume. After that, we use two-stream base models to perform action classification with only video-level labels, and then iteratively refine the base models with a frame-level pseudo ground truth.

Feature Extraction. The RGB and optical flow snippet-level features are extracted with pre-trained networks (*e.g.*, I3D [25]) from non-overlapping fixed-length RGB and optical flow snippets, respectively. They provide high-level appearance and motion information of the corresponding snippets. Formally, given a video with T non-overlapping snippets, we denote the extracted RGB feature and optical flow feature as $\mathbf{F}_{\text{RGB}} = \{\mathbf{f}_{\text{RGB},i}\}_{i=1}^T$ and $\mathbf{F}_{\text{flow}} = \{\mathbf{f}_{\text{flow},i}\}_{i=1}^T$, respectively, where $\mathbf{f}_{\text{RGB},i}, \mathbf{f}_{\text{flow},i} \in \mathbb{R}^D$ are the feature representations of the i-th RGB snippet and the i-th optical flow snippet, respectively, and D denotes the channel dimension.

The features of the two modalities are fed into two separate base models respectively, and the two base models use the same architecture but do not share parameters. Therefore, in the rest of this section, for conciseness, we omit the subscript RGB and flow to indicate a general operation for both modalities.

Feature Embedding. Since the feature-extraction backbones are not originally trained for the W-TAL task, we embed the extracted feature $\mathbf{F}$ with two layers of temporal convolutional layer interleaved with LeakyReLU activation. We denote the output feature as $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^T$, where $\mathbf{x}_i \in \mathbb{R}^D$. The embedding temporal convolutional layer consists of D convolutional kernels with a temporal size of 3 and a stride of 1. Besides, zero padding is used to retain the temporal dimension.

Action Recognition. As untrimmed videos may contain background snippets, to perform the video-level classification, we need to select snippets that are likely to contain action instances and meanwhile filter out snippets that are likely to contain background. To this end, an attention value $a_i \in (0, 1)$ to measure the likelihood of the i-th snippet containing an action is given by an attention module:

$$a_i = \sigma\left(g_{\text{att}}(\mathbf{x}_i; \mathbf{\Psi}_{\text{att}})\right), \tag{1}$$

where $\sigma(\cdot)$, $g_{\text{att}}(\cdot)$ and $\mathbf{\Psi}_{\text{att}}$ are the sigmoid function, the forward pass of the attention module and learnable parameters of the attention module, respectively. The attention module is implemented as a single temporal convolutional layer with a kernel size 3.

With the obtained attention sequence, we then perform attention-weighted pooling over the feature sequence to generate a single foreground feature $\mathbf{x}_{\text{fg}}$, and feed it to a classification module to get the video-level prediction $\hat{\mathbf{y}}$:

$$\mathbf{x}_{\text{fg}} = \frac{1}{\sum_{i=1}^T a_i} \sum_{i=1}^T a_i \mathbf{x}_i, \tag{2}$$

$$\hat{\mathbf{y}} = \text{softmax}\left(g_{\text{cls}}\left(\mathbf{x}_{\text{fg}}; \mathbf{\Psi}_{\text{cls}}\right)\right), \tag{3}$$

where softmax$(\cdot)$ is a softmax function along the class dimension, $g_{\text{cls}}(\cdot)$ is the forward pass of the classification module, and $\mathbf{\Psi}_{\text{cls}}$ is the learnable parameters of the classification module. The classification module share a similar structure with the attention module, except that the output layer consists of C convolutional kernels. The classification loss function $\mathcal{L}_{\text{cls}}$ is defined as the standard cross entropy loss:

$$\mathcal{L}_{\text{cls}} = -\sum_{c=1}^{C} y_c \log\left(\hat{y}_c\right), \tag{4}$$

where y_c and $\hat{y}_c$ denote the values of the label vector $\mathbf{y}$ and the action prediction result $\hat{\mathbf{y}}$ at index c, respectively.

In addition, the temporal-class activation map (T-CAM) [4], [15] $\mathbf{S} = \{\mathbf{s}_i\}_{i=1}^{T}$, $\mathbf{s}_i \in \mathbb{R}^C$, which is used to measure the action proposal confidence score in Section 3.4, is generated by sliding the classification module over all snippet-level features:

$$\mathbf{s}_i = \text{softmax}\left(g_{\text{cls}}(\mathbf{x}_i; \mathbf{\Psi}_{\text{cls}})\right). \tag{5}$$

Adaptive attention normalization loss. Ideally, the attention values are expected to be binary, where 1 indicates actions while 0 indicates background. To this end, the original TSCN [18] uses an attention normalization loss to maximize the difference between the top-l and bottom-l average attention values:

$$\mathcal{L}_{\text{norm}} = \frac{1}{l}\min_{\substack{a\subset\{a_i\}\\|a|=l}}\sum_{\phi\in a}\phi - \frac{1}{l}\max_{\substack{a\subset\{a_i\}\\|a|=l}}\sum_{\phi\in a}\phi, \tag{6}$$

where $l = \max\left(1, \lfloor\frac{T}{s}\rfloor\right)$ and s is set to 8 empirically.

One problem with this loss function is that it only applies on $1/4$ of the whole video, leading to limited training samples. However, in the weakly-supervised setting, the portions of actions and background are unknown. To address this problem, inspired by Otsu's method in image binarization [16], we dynamically determine an attention threshold θ_{otsu} via Otsu's method, which is further used to separate action snippets and background snippets. Otsu's method searches for a threshold that minimizes a weighted sum of action and background attention variances:

$$\begin{aligned}\theta_{\text{otsu}} = \arg\min_{\theta\in\{a_i\}} &|\{a_i|a_i<\theta\}|\text{var}(\{a_i|a_i<\theta\})+ \\ &|\{a_i|a_i\geq\theta\}|\text{var}(\{a_i|a_i\geq\theta\}),\end{aligned} \tag{7}$$

where $\text{var}(\cdot)$ denotes the variance, and $|\cdot|$ denotes the cardinality of a set. In this way, a background set $\mathbb{A}_{\text{bg}}$ and an action set $\mathbb{A}_{\text{act}}$ can be generated as $\mathbb{A}_{\text{bg}} = \{a_i|a_i < \theta_{\text{otsu}}\}$ and $\mathbb{A}_{\text{act}} = \{a_i|a_i \geq \theta_{\text{otsu}}\}$, respectively. Then, the final adaptive attention normalization loss is defined as:

$$\mathcal{L}_{\text{a-norm}} = \frac{1}{l_{\text{bg}}}\min_{\substack{a\subset\{a_i\}\\|a|=l_{\text{bg}}}}\sum_{\phi\in a}\phi - \frac{1}{l_{\text{act}}}\max_{\substack{a\subset\{a_i\}\\|a|=l_{\text{act}}}}\sum_{\phi\in a}\phi, \tag{8}$$

where $l_{\text{bg}} = \max\left(\lfloor\frac{T}{s}\rfloor, \lfloor\frac{|\mathbb{A}_{\text{bg}}|}{s'}\rfloor\right)$, and $l_{\text{act}} = \max\left(\lfloor\frac{T}{s}\rfloor, \lfloor\frac{|\mathbb{A}_{\text{act}}|}{s'}\rfloor\right)$. We note the lower bound $\lfloor\frac{T}{s}\rfloor$ is indispensable, otherwise the loss function will converge to only a few snippets being regarded as actions. Besides, under the weakly-supervised setting, a single stream is prone to make incorrect snippet-level action-background classification. Thus, a hyperparameter s' is used to make the loss function only focus on a portion of the most confident part. $s' = 2$ is empirically determined so that it doubles the training snippets than the original attention normalization loss.

Smooth Loss. As a minor improvement, we introduce a smooth loss to enforces temporally proximate snippets to give similar attention predictions, and thus helps generate a more smooth attention sequence [66]:

$$\mathcal{L}_{\text{smooth}} = \frac{1}{T-1}\sum_{t=1}^{T-1}|a_t - a_{t+1}|. \tag{9}$$

Total Loss. The overall loss for the base model training is a weighted sum of the classification loss, the adaptive attention normalization term, and the smooth loss:

$$\mathcal{L}_{\text{base}} = \mathcal{L}_{\text{cls}} + \alpha\mathcal{L}_{\text{a-norm}} + \beta\mathcal{L}_{\text{smooth}}, \tag{10}$$

where α and β are hyperparameters to control the weight of the adaptive attention normalization loss and the smooth loss.

3.3 Pseudo Ground Truth Learning

After training the base models with only video-level labels, we then iteratively refine the two-stream base models with a novel frame-level pseudo ground truth.

Specifically, we divide the whole training process into several refinement iterations. At refinement iteration 0, only video-level labels are leveraged for training. And at refinement iteration $n+1$, a frame-level pseudo ground truth is generated at refinement iteration n, and provides frame-level supervision for the current refinement iteration. However, without true frame-level ground truth annotation, we can neither measure the quality of the pseudo ground truth, nor guarantee the pseudo ground truth can help the base models achieve higher performance.

Inspired by two-stream late fusion [4], [6], [9], [10], [17], we introduce a simple yet effective method to generate the pseudo ground truth. Intuitively, the late fusion is a voting ensemble of two streams: locations at which both streams have high activations are likely to contain ground truth action instances; locations at which only one stream has high activations are likely to be either false positive action proposals or true action instances that only one stream can detect; locations at which both streams both have low activations are likely to be the background. Therefore, late fusion can effectively combine knowledge learned by two streams, and generates a more reliable attention sequence than each individual stream.

Following this intuition, we use the fusion attention sequence $\{a^{(n)}_{\text{fuse},i}\}_{i=1}^{T}$ at refinement iteration n to generate pseudo ground truth $\{\mathcal{G}_i^{(n+1)}\}_{i=1}^{T}$ for refinement iteration $n+1$, where $a^{(n)}_{\text{fuse},i} = \lambda a^{(n)}_{\text{RGB},i} + (1-\lambda)a^{(n)}_{\text{flow},i}$, and $\lambda \in [0,1]$ is a fusion hyperparameter to control the relative importance of RGB and flow attentions. We then refine the base models by forcing the attention sequence predicted by *each* stream to fit the pseudo ground truth. In this paper, we introduce two pseudo ground truth generation methods.

- **Soft pseudo ground truth** directly uses the fusion attention values as pseudo labels: $\mathcal{G}_i^{(n+1)} = a^{(n)}_{\text{fuse},i}$. The soft pseudo labels contain the probability of a snippet being the foreground action, but also add uncertainty to the model.

- **Hard pseudo ground truth** thresholds the attention sequence to generate a binary sequence:

$$\mathcal{G}_i^{(n+1)} = \begin{cases} 1, & a_{\text{fuse},i}^{(n)} > \theta; \\ 0, & a_{\text{fuse},i}^{(n)} \leq \theta, \end{cases} \tag{11}$$

where θ is the threshold value. Setting a large value of θ will eliminate the action proposals that only one stream has high activations, reducing the false positive rate. In contrast, setting a small value of θ will help models to generate more and longer action proposals and achieve a higher recall. Hard pseudo labels remove the ambiguity and provide stronger supervision, but introduce a hyperparameter.

After generating the pseudo ground truth, the attention sequences of each single stream are forced to fit the pseudo ground truth with a mean square error loss:

$$\mathcal{L}_{\text{pseudo}}^{(n+1)} = \frac{1}{T}\sum_{i=1}^{T}\left(a_i^{(n+1)} - \mathcal{G}_i^{(n+1)}\right)^2. \tag{12}$$

However, under only video-level supervision, the pseudo labels are prone to be noisy. To alleviate this issue, we introduce a video-level uncertainty estimator w_{video} and a snippet-level uncertainty estimator $w_{\text{snippet},i}$ for pseudo ground truth learning. The video-level and snippet-level uncertainty estimators leverage the agreement of two-stream outputs at video-level and snippet-level, respectively. Specifically, the video-level uncertainty estimator measures the confidence of the pseudo ground truth for a given video, and assigns larger/smaller weights to confident/ambiguous pseudo ground truth in a batch. The snippet-level uncertainty estimator measures the snippet-level confidence for the pseudo ground truth, and assigns larger/smaller weights to confident/ambiguous snippets in a video. The adaptive pseudo ground truth learning loss with video-level and snippet-level uncertainty estimators is formulated as

$$\mathcal{L}_{\text{a-pseudo}}^{(n+1)} = w_{\text{video}}\frac{1}{T}\sum_{i=1}^{T} w_{\text{snippet},i}\left(a_i^{(n+1)} - \mathcal{G}_i^{(n+1)}\right)^2. \tag{13}$$

For the video-level uncertainty estimator, we consider two different implementations.

- **Attention difference**: the difference between two-stream average attention values is leveraged to measure the uncertainty, where the video-level uncertainty is defined as $w_{\text{video}} = 1 - |\frac{1}{T}\sum_1^T a_{\text{RGB},i} - \frac{1}{T}\sum_1^T a_{\text{flow},i}|$.
- **Symmetric KL divergence on attention distribution**: this estimator considers the two-stream attention distribution, and measures the video-level consensus with symmetric KL divergence. To simplify computation, we approximate the attention distribution by dividing the attention into b bins, where b is a hyperparameter. The i-th bin contains snippets with an attention value in the range $[\frac{i-1}{b}, \frac{i}{b}]$. In this way, we can denote the attention distribution as a vector $\tilde{\mathbf{a}} \in \mathbb{R}^b$, where its i-th value $\tilde{a}_i$ denotes the ratio of the number of snippet contained in the i-th bin to the total number of snippets. Therefore, the two-stream attention distributions can be obtained as $\tilde{\mathbf{a}}_{\text{RGB}}$ and $\tilde{\mathbf{a}}_{\text{flow}}$, respectively, and the video-level uncertainty is estimated by symmetric KL divergence $w_{\text{video}} = \exp(-\text{KL}(\tilde{\mathbf{a}}_{\text{RGB}}\|\tilde{\mathbf{a}}_{\text{flow}}) - \text{KL}(\tilde{\mathbf{a}}_{\text{flow}}\|\tilde{\mathbf{a}}_{\text{RGB}}))$, where $\text{KL}(\cdot\|\cdot)$ denotes the KL divergence.

For the snippet-level uncertainty estimator, we also consider two different implementations.

- **Attention difference**: this estimator measures the snippet-level uncertainty via the difference between two-stream attention values, where $w_{\text{snippet},i} = 1 - |a_{\text{RGB},i} - a_{\text{flow},i}|$.
- **Symmetric KL divergence on T-CAM**: this estimator computes the symmetric KL divergence between the two-stream T-CAM: $w_{\text{snippet},i} = \exp(-\text{KL}(\mathbf{s}_{\text{RGB},i}\|\mathbf{s}_{\text{flow},i}) - \text{KL}(\mathbf{s}_{\text{flow},i}\|\mathbf{s}_{\text{RGB},i}))$.

To avoid the impact of numerical differences of different uncertainty estimators, we further normalize the uncertainties with min-max normalization, and add a bias so that the video-level/snippet-level uncertainties have a batch-/video-wise average uncertainties of 1.

Note that we only apply the pseudo ground truth learning to the attention sequence, while no constraint is applied to the classification module. This is because the classification module is primarily guided by the attention: the classification module uses an attention-weighted pooled feature to perform action recognition, and thus its activation resembles the attention.

Finally, at refinement iteration $n+1$, the total loss for each stream is

$$\mathcal{L}_{\text{total}}^{(n+1)} = \mathcal{L}_{\text{base}} + \gamma\mathcal{L}_{\text{a-pseudo}}^{(n+1)}, \tag{14}$$

where γ is a hyperparameter to control the relative importance of the pseudo ground truth learning.

3.4 Action Localization

During testing, following recent methods [4], [12], we first temporally upsample the attention sequence and T-CAM by a factor of 8 via linear interpolation. Since a video may contain action instances from different categories, we then select top-k action categories from the fusion video-level prediction $\hat{\mathbf{y}}_{\text{fuse}}$ to perform action localization, where $\hat{\mathbf{y}}_{\text{fuse}} = \lambda\hat{\mathbf{y}}_{\text{RGB}} + (1-\lambda)\hat{\mathbf{y}}_{\text{flow}}$. For each of these categories, following common practice [4], [9], [12], we generate action proposals by progressively thresholding the attention values, and concatenating consecutive snippets. The action proposals are scored via a variant of the Outer-Inner-Constrastive score [3]: instead of using average T-CAM, we use attention-weighted T-CAM to measure the temporal contrast between the action proposal and its surrounding areas. Formally, given an action proposal (t_s, t_e, c), a fusion attention sequence $\{a_{\text{fuse},i}\}_{i=1}^T$ and a fusion T-CAM $\{\mathbf{s}_{\text{fuse},i}\}_{i=1}^T$, where $\mathbf{s}_{\text{fuse},i} = \lambda\mathbf{s}_{\text{RGB},i} + (1-\lambda)\mathbf{s}_{\text{flow},i}$, the confidence score ψ is computed as

$$\begin{aligned}\psi =& \frac{\sum_{i=t_s}^{t_e} a_{\text{fuse},i}s_{\text{fuse},i,c}}{t_e - t_s} - \\ & \frac{\sum_{i=T_s}^{T_e} a_{\text{fuse},i}s_{\text{fuse},i,c} - \sum_{i=t_s}^{t_e} a_{\text{fuse},i}s_{\text{fuse},i,c}}{T_e - T_s - (t_e - t_s)},\end{aligned} \tag{15}$$

where $T_s = t_s - \frac{L}{4}$, $T_e = t_e + \frac{L}{4}$, $L = t_e - t_s$, and $s_{\text{fuse},i,c}$ is the fusion T-CAM value of i-th snippet for category c. Finally, non-maximum suppression is used within each class to remove duplicated detections.

4 Experiments and Discussions

In this section, we first introduce four standard benchmarks, the evaluation metrics, and the implementation details. Then, we compare the proposed A-TSCN with state-of-the-art methods, followed by a set of ablation studies. Note that only video-level categorical labels are leveraged to train the proposed A-TSCN.

4.1 Dataset and Evaluation

THUMOS14 dataset [35] contains 200 validation videos and 213 testing videos within 20 categories for the TAL task. We use the 200 validation videos to train, and use the 213 testing videos to evaluate. Following BaS-Net [12], we remove testing video #270, #1292 and #1496 as they are incorrectly annotated. Each video averagely contains 15.5 action instances in the THUMOS14 dataset.

ActivityNet dataset [37] has two release versions, *i.e.*, ActivityNet v1.3 and ActivityNet v1.2. ActivityNet v1.3 covers 200 action categories, with a training set of $10,024$ videos and a validation set of $4,926$ videos. ActivityNet v1.2 is a subset of ActivityNet v1.3, and covers 100 action categories, with $4,819$ and $2,383$ videos in the training and validation set, respectively.[1] We use the training set and the validation set for training and testing, respectively. Each video averagely contains 1.5 action instances in ActivityNet datasets.

HACS dataset [19] is a recently released dataset for the TAL task. To our knowledge, it is so far the largest TAL benchmark, and covers 200 action classes, with a training set of $37,612$ videos, and a validation set of $5,981$ videos. We use the HACS v1.1.1 to conduct the experiments. Each video in this dataset contains 2.5 action instances on average.

Evaluation Metrics. Following the standard protocol on temporal action localization, we evaluate our method with mean Average Precision (mAP) under different Intersection-over-Union (IoU) thresholds. We use the evaluation code provided by ActivityNet to measure the performance.

4.2 Implementation Details

The optical flow is estimated via the TV-L1 algorithm [71]. Two off-the-shelf feature-extraction backbones are used in our experiments, *i.e.*, UntrimmedNet [2] and I3D [25], with snippet lengths of 15 frames and 16 frames, respectively. The two backbones are pre-trained on ImageNet [72] and Kinetics-400 [25] respectively, and are not fine-tuned for a fair comparison. The RGB and optical flow snippet-level features are extracted at the `global_pool` layer as 1024-D vectors.

The network is implemented in PyTorch [73]. We use the AdamW optimizer with a fixed learning rate 0.0001 during the whole training process. For the pseudo ground truth generation, we simply select models in the last epoch of each refinement iterations as the teacher model. The batch size is set to 16. During testing, we choose top-2 action categories and reject categories whose fusion classification prediction scores are lower than 0.1 to perform action localization. To remove fragmentary action proposals in ActivityNet datasets, we downsample the input at a rate of $1/30$. The numbers of epochs are set to 80, 20, and 20 for refinement iteration 0 for THUMOS14, ActivityNet and HACS, respectively; and for later refinement iterations, the numbers of epochs are set to 40, 10 and 10, respectively. The number of training epochs is largely affected by the number of training videos per class in each dataset. For all datasets, we train the model for 8 refinement iterations.

Hyperparameters. For the s in the original attention normalization loss, we follow similar weakly-supervised classification loss functions [5], [12] to set $s = 8$. And s' in the adaptive attention normalization loss is set to 2, which doubles the training snippets than our conference version. The weights for loss functions are set by only adjusting their magnitudes: $\alpha = \beta = 0.1$, and $\gamma = 1$. And the fusion parameter λ is set to 0.5, so that the two modalities are equally weighted. According to our intuition that the attention performs binary classification, the thresholding parameter θ is set to 0.5. For the symmetric KL divergence in the uncertainty estimators, we set $b = 10$. As will be shown in the ablation study, our method is robust to most of the hyperparameters.

4.3 Comparisons with the State-of-the-art

THUMOS14. Table 1 summarizes the performance comparison between the proposed A-TSCN and state-of-the-art fully-supervised and weakly-supervised TAL methods on the THUMOS14 testing set. With UntrimmedNet features, A-TSCN outperforms other W-TAL methods at most IoU thresholds by a large margin, and even achieves comparable results to some recent W-TAL methods with I3D features (*e.g.*, BaS-Net [12] and DGAM [14]) at several IoU thresholds.

The proposed A-TSCN achieves higher performance with I3D features, and outperforms all of the previous W-TAL methods at the average mAP between 0.3 and 0.7. Furthermore, our A-TSCN achieves similar performance to some recent fully-supervised methods (*e.g.*, SSN [40]), and even outperforms TAL-net [46] at IoU thresholds 0.1 and 0.2. However, as the IoU threshold increases, the performance of A-TSCN drops significantly, because localizing more precise action boundaries needs true frame-level ground truth supervision.

ActivityNet. The performance comparisons on ActivityNet v1.2 and v1.3 are shown in Table 2 and Table 3 respectively, where our models are trained with I3D features. The proposed A-TSCN outperforms previous W-TAL methods at the average mAP at IoU threshold $0.5 : 0.05 : 0.95$ on both release versions of ActivityNet, verifying the efficacy of our design intuition.

HACS. The performance comparison on the HACS validation set is presented in Table 4, where all methods are trained with I3D features. Our A-TSCN outperforms the previous fully-supervised method SSN [40], the weakly-supervised method BaS-Net [12] and our baseline model TSCN [18] at all IoU thresholds and the average mAP. To our knowledge, the HACS dataset is the largest dataset for the TAL task, and it is a realistic and challenging one due to its fine-grained annotation. Thus, our performance superiority on this dataset indicates its applicability to real scenarios.

To summarize, on the above four datasets, the proposed A-TSCN outperforms state-of-the-art W-TAL methods,

[1] In our experiments, there are $9,937$ and $4,575$ videos in the training and validation set of ActivityNet v1.3 respectively, and $4,471$ and $2,211$ videos in the training and validation set of ActivityNet v1.2 respectively, because the rest of the videos are inaccessible from YouTube.

TABLE 1
Comparison of our method with state-of-the-art TAL methods on the THUMOS14 testing set. Recent fully-supervised and weakly-supervised methods are reported. UNT and I3D are abbreviations for UntrimmedNet feature and I3D feature, respectively. The Avg column indicates the average mAP at IoU thresholds 0.3:0.1:0.7.

Method	Supervision	Feature	mAP@IoU (%)									Avg (%)
			0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.3:0.1:0.7
Richard *et al.* [67]	Full	-	39.7	35.7	30.0	23.2	15.2	-	-	-	-	-
Yuan *et al.* [68]	Full	-	51.0	45.2	36.5	27.8	17.8	-	-	-	-	-
CDC [39]	Full	-	-	-	40.1	29.4	23.3	13.1	7.9	-	-	22.8
R-C3D [44]	Full	-	54.5	51.5	44.8	35.6	28.9	-	-	-	-	-
SSN [40]	Full	-	66.0	59.4	51.9	41.0	29.8	-	-	-	-	-
BSN [45]	Full	-	-	-	53.5	45.0	36.9	28.4	20.0	-	-	36.8
BMN [50]	Full	-	-	-	56.0	47.4	38.8	29.7	20.5	-	-	38.5
GTAN [49]	Full	-	69.1	63.7	57.8	47.2	38.8	-	-	-	-	-
G-TAD [52]	Full	-	-	-	54.5	47.6	40.2	30.8	23.4	-	-	39.3
TAL-Net [46]	Full	I3D	59.8	57.1	53.2	48.5	42.8	33.8	20.8	-	-	39.8
P-GCN [51]	Full	I3D	69.5	67.8	63.6	57.8	49.1	-	-	-	-	-
UntrimmedNet [2]	Weak	-	44.4	37.7	28.2	21.1	13.7	-	-	-	-	-
STPN [4]	Weak	UNT	45.3	38.8	31.1	23.5	16.2	9.8	5.1	2.0	0.3	17.1
W-TALC [5]	Weak	UNT	49.0	42.8	32.0	26.0	18.8	10.9	6.2	-	-	18.8
Liu *et al.* [6]	Weak	UNT	53.5	46.8	37.5	29.1	19.9	12.3	6.0	-	-	21.0
AutoLoc [3]	Weak	UNT	-	-	35.8	29.0	21.2	13.4	5.8	-	-	21.0
TSM [11]	Weak	UNT	-	-	37.3	-	21.9	-	6.0	-	-	-
RefineLoc [59]	Weak	UNT	-	-	36.1	29.6	22.6	12.1	5.8	-	-	21.2
Huang *et al.* [13]	Weak	UNT	54.2	47.1	37.8	29.4	21.2	13.9	6.8	-	-	21.8
CleanNet [8]	Weak	UNT	-	-	37.0	30.9	23.9	13.9	7.1	-	-	22.6
BaS-Net [12]	Weak	UNT	56.2	50.3	42.8	34.7	25.1	17.1	9.3	3.7	**0.5**	25.8
EM-MIL [58]	Weak	UNT	59.0	50.4	42.7	34.5	27.2	18.9	10.2	-	-	26.7
TSCN [18]	Weak	UNT	58.9	52.9	45.0	36.6	27.6	18.8	10.2	**4.0**	**0.5**	27.6
A-TSCN (Ours)	Weak	UNT	**60.5**	**54.0**	**46.3**	**37.4**	**28.8**	**19.2**	**10.3**	3.8	0.4	**28.4**
STPN [4]	Weak	I3D	52.0	44.7	35.5	25.8	16.9	9.9	4.3	1.2	0.1	18.5
W-TALC [5]	Weak	I3D	55.2	49.6	40.1	31.1	22.8	14.5	7.6	-	-	23.2
Liu *et al.* [6]	Weak	I3D	57.4	50.8	41.2	32.1	23.1	15.0	7.0	-	-	23.7
TSM [11]	Weak	I3D	-	-	39.5	-	24.5	-	7.1	-	-	-
RefineLoc [59]	Weak	I3D	-	-	40.8	32.7	23.1	13.3	5.3	-	-	23.0
BaS-Net [12]	Weak	I3D	58.2	52.3	44.6	36.0	27.0	18.6	10.4	3.9	0.5	27.3
Nguyen *et al.* [9]	Weak	I3D	60.4	56.0	46.6	37.5	26.8	17.6	9.0	3.3	0.4	27.5
Huang *et al.* [13]	Weak	I3D	62.3	57.0	48.2	37.2	27.9	16.7	8.1	-	-	27.6
DGAM [14]	Weak	I3D	60.0	54.2	46.4	38.2	28.8	19.8	11.4	3.6	0.4	29.0
A2CL-PT [57]	Weak	I3D	61.2	56.1	48.1	39.0	30.1	19.2	10.6	**4.8**	**1.0**	29.4
EM-MIL [58]	Weak	I3D	59.1	52.7	45.5	36.8	30.5	22.7	**16.4**	-	-	30.4
TSCN [18]	Weak	I3D	63.4	57.6	47.8	37.7	28.7	19.4	10.2	3.9	0.7	28.8
AUMN [69]	Weak	I3D	66.2	61.9	54.9	44.4	33.3	20.5	9.0	-	-	32.4
TSCN+UGCT [70]	Weak	I3D	**67.5**	**62.1**	**55.3**	**45.2**	33.3	20.7	9.5	-	-	32.8
A-TSCN (Ours)	Weak	I3D	65.3	59.0	52.1	42.5	**33.6**	**23.4**	12.7	4.5	0.5	**32.9**

TABLE 2
Comparison of our method with state-of-the-art TAL methods on the ActivityNet v1.2 validation set. The Avg column indicates the average mAP at IoU thresholds 0.5:0.05:0.95.

Method	Sup.	mAP@IoU (%)			Avg (%)
		0.5	0.75	0.95	0.5:0.05:0.95
SSN [40]	Full	41.3	27.0	6.1	26.6
UntrimmedNet [2]	Weak	7.4	3.2	0.7	3.6
AutoLoc [3]	Weak	27.3	15.1	3.3	16.0
TSM [11]	Weak	28.3	17.0	3.5	17.1
W-TALC [5]	Weak	37.0	12.7	1.5	18.0
Liu *et al.* [6]	Weak	36.8	22.0	5.6	22.4
Huang *et al.* [13]	Weak	37.6	23.9	5.4	23.3
BaS-Net [12]	Weak	38.5	24.2	5.6	24.3
DGAM [14]	Weak	41.0	23.5	5.3	24.4
EM-MIL [58]	Weak	37.4	-	2.0	20.3
TSCN [18]	Weak	37.6	23.7	5.7	23.6
AUMN [69]	Weak	**42.0**	25.0	5.6	25.5
TSCN+UGCT [70]	Weak	40.0	23.6	5.6	24.3
A-TSCN (Ours)	Weak	39.6	**25.1**	**5.8**	**25.6**

TABLE 3
Comparison of our method with state-of-the-art W-TAL methods on the ActivityNet v1.3 validation set. The Avg column indicates the average mAP at IoU thresholds 0.5:0.05:0.95.

Method	mAP@IoU (%)			Avg (%)
	0.5	0.75	0.95	0.5:0.05:0.95
STPN [4]	29.3	16.9	2.6	-
TSM [11]	30.3	19.0	4.5	-
Liu *et al.* [6]	34.0	20.9	**5.7**	21.2
Nguyen *et al.* [9]	36.4	19.2	2.9	-
BaS-Net [12]	34.5	22.5	4.9	22.2
A2CL-PT [57]	36.8	22.0	5.2	22.5
TSCN [18]	35.3	21.4	5.3	21.7
AUMN [69]	**38.3**	**23.5**	5.2	23.5
TSCN+UGCT [70]	38.1	21.2	5.4	22.8
A-TSCN (Ours)	37.9	23.1	5.6	**23.6**

including TSCN proposed in our conference paper [18]. Surprisingly, our A-TSCN achieves similar or even higher performance than some recent fully-supervised methods on the four benchmarks. The clear performance superiority demonstrates the effectiveness of the proposed A-TSCN.

4.4 Ablation Study

In this subsection, to better analyze the contribution of each component, we conduct ablation studies on the THUMOS14

TABLE 4
Comparison of our method with state-of-the-art TAL methods on the HACS validation set. The Avg column indicates the average mAP at IoU thresholds 0.5:0.05:0.95. * denotes our reproduced results.

Method	Sup.	mAP@IoU (%)			Avg (%)
		0.5	0.75	0.95	0.5:0.05:0.95
SSN [40]	Full	28.82	18.80	5.32	18.97
BaS-Net [12]*	Weak	30.12	16.69	6.13	18.63
TSCN [18]	Weak	33.40	19.97	6.45	20.80
A-TSCN (Ours)	Weak	**34.86**	**20.89**	**6.60**	**21.71**

TABLE 5
Ablation study on the adaptive attention normalization loss $\mathcal{L}_{\text{a-norm}}$. #Act and #Bg denote the average number of positive and negative snippets participated in the loss function computation in the testing set, respectively. Performances are reported w/o pseudo ground truth learning.

Loss	s	s'	mAP@IoU (%)			Avg (%)	#Act	#Bg
			0.3	0.5	0.7	0.3:0.1:0.7		
-	-	-	33.1	19.0	5.7	18.2	-	-
$\mathcal{L}_{\text{norm}}$	2	-	42.8	24.3	8.0	23.3	196.4	196.4
	4	-	44.4	27.1	9.7	26.6	98.2	98.2
	8	-	45.7	29.3	10.6	28.4	49.1	49.1
	16	-	44.0	28.1	9.7	27.3	24.6	24.6
$\mathcal{L}_{\text{a-norm}}$	-	1	44.6	27.3	10.4	26.8	157.5	239.2
	-	2	40.1	22.9	6.0	21.9	35.2	162.7
	-	4	37.4	18.1	4.5	17.6	14.6	83.9
	-	8	32.2	14.4	3.8	13.9	6.3	42.6
	8	1	45.4	27.8	10.3	27.5	168.8	216.0
	8	2	47.9	30.3	10.7	29.6	71.1	130.9
	8	4	46.9	30.0	10.5	29.2	50.2	66.6

testing set. The ablation studies are conducted with I3D features. To improve readability, we use gray color to mark the final setting used to compared with the state-of-the-art.

Ablation study on the adaptive attention normalization loss $\mathcal{L}_{\text{a-norm}}$. To reduce the ambiguity between foreground and background, we introduce an adaptive attention normalization loss to differentiate them in attention values. Compared with the original version, where the action and background portions are fixed, the new adaptive version dynamically determines the action and background portions according to the attention distribution, increasing the training samples and improving the performance. Table 5 compares the performance of the original attention normalization loss and its adaptive version. We make the following observations. (1) For the original attention normalization loss $\mathcal{L}_{\text{norm}}$, the performance first raises as s increases from 2 to 8, indicating manually setting a large portion of action or background does not conform to the real action and background distribution (*e.g.*, setting $s = 4$ assumes action and background each account for 25% of the whole video). The performance drops at $s = 16$, which might attribute to the decrease of training samples. (2) For the adaptive version without a lower bound, the performance significantly drops as s' increases. Besides, the number of action snippets decreases much more quickly than the number of background snippets. Without a lower bound constraint, we speculate that the model only focuses on the most discriminative part of actions for classification, while ignoring the completeness of action instances. (3) With the lower bound constraint (the last group), the performance improves significantly for $s' = 2$ and $s' = 4$, which demonstrates the effectiveness of our adaptive attention normalization loss. Besides, setting $s' = 1$ slightly decreases the performance, indicating that it is unreliable to determine the action and background for the whole video for a single modality.

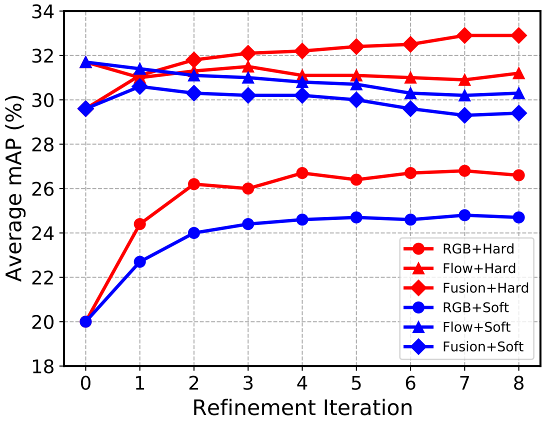


Fig. 3. Comparison between models trained with different pseudo ground truth at different refinement iterations on the THUMOS14 testing set. “Hard” denotes models trained with hard pseudo ground truth, and “Soft” denotes models trained with soft pseudo ground truth.

Ablation study on the pseudo ground truth type. Fig. 3 plots the performance comparison between different pseudo ground truth methods at different refinement iterations. The results reveal that the hard pseudo ground truth dramatically improves the performance for the RGB stream and the fusion result. Despite the slight performance drop for the flow stream, the fusion result outperforms both streams after the hard pseudo label learning. In contrast, the soft pseudo ground truth degrades the performance of the flow stream and the fusion result. As for the RGB stream, though the soft labels improve its performance, the improvement requires more refinement iterations and is still lower than that trained with hard labels. These results reveal the importance of removing ambiguity in the pseudo ground truth.

In the following discussion, if not explicitly stated, the pseudo ground truth denotes the hard pseudo ground truth.

Table 6 lists the detailed performance comparison between models trained with only video-level labels and those trained with pseudo ground truth. The results show that the pseudo ground truth improves the localization performance for the RGB stream and the fusion result at all IoU thresholds, and improves the flow stream at high IoU thresholds. Also, the pseudo ground truth dramatically improves the precision and recall for the RGB stream, and improves the precision for the flow stream and the fusion result with a slight loss of recall. The pseudo ground truth improves the F-measure for all three results. This demonstrates that the pseudo ground truth can help eliminate false positive action proposals.

The per category precision-recall (PR) curve is presented in Fig. 4. The category-wise PR curve indicates that the pseudo ground truth improves precision for most categories (*i.e.*, higher in the y axis), and thus achieves a larger area enclosed by the PR curve, the x axis and the y axis (*i.e.*, average precision, AP). However, for several categories (*e.g.*, *Cricket Shot* and *Tennis Swing*), the performance slightly drops. The reason is that some false positive action proposals are wrongly reinforced in the iterative refinement, and we will further illustrate this problem in qualitative analysis.

TABLE 6
Comparison between the models trained with only video-level labels and the model trained with hard pseudo ground truth on the THUMOS14 testing set. The "Label" column denotes the supervision used in training, where "Video" indicates only video-level labels are leveraged, and "Frame" indicates the hard pseudo ground truth is also leveraged during training. Precision, recall and F-measure are calculated under IoU threshold 0.5.

Modality	Label	mAP@IoU (%)									Avg (%)	Recall (%)	Precision (%)	F-measure
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.3:0.1:0.7			
RGB	Video	54.5	47.2	38.0	28.3	18.4	10.5	4.7	1.0	0.1	20.0	48.0	6.0	0.1067
RGB	Frame	58.5	53.0	45.1	35.9	26.9	17.8	8.3	2.9	0.4	26.8	53.4	9.9	0.1670
Flow	Video	63.5	58.2	51.0	41.8	32.2	21.6	11.9	3.7	0.4	31.7	61.0	7.3	0.1304
Flow	frame	62.8	56.7	50.2	40.7	31.1	21.4	11.6	4.3	0.5	31.0	54.8	10.2	0.1720
Fusion	Video	61.6	55.2	47.9	39.5	30.3	19.8	10.7	3.0	0.3	29.6	67.0	7.3	0.1316
Fusion	Frame	65.3	59.0	52.1	42.5	33.6	23.4	12.7	4.5	0.5	32.9	63.2	9.4	0.1636

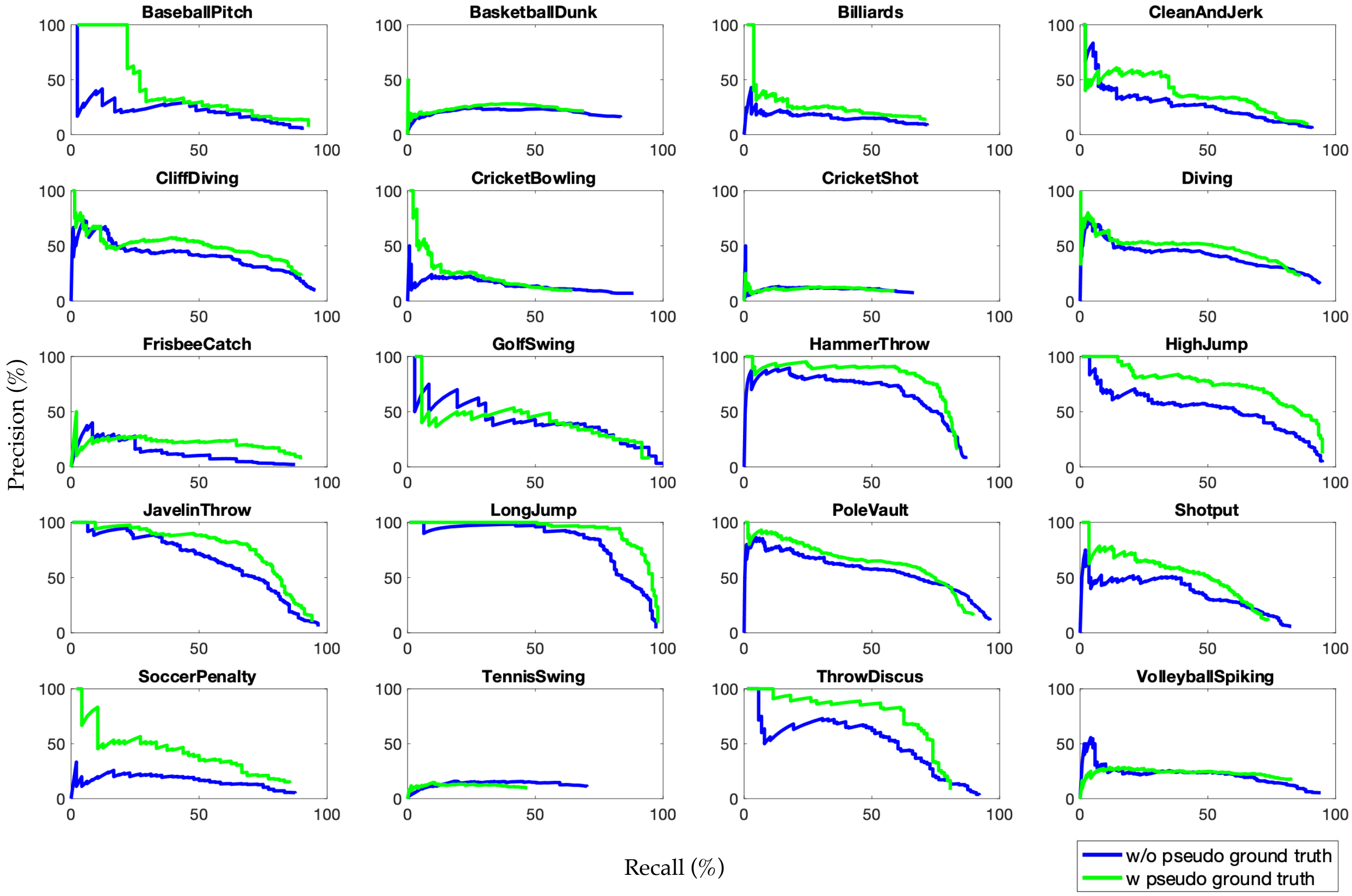
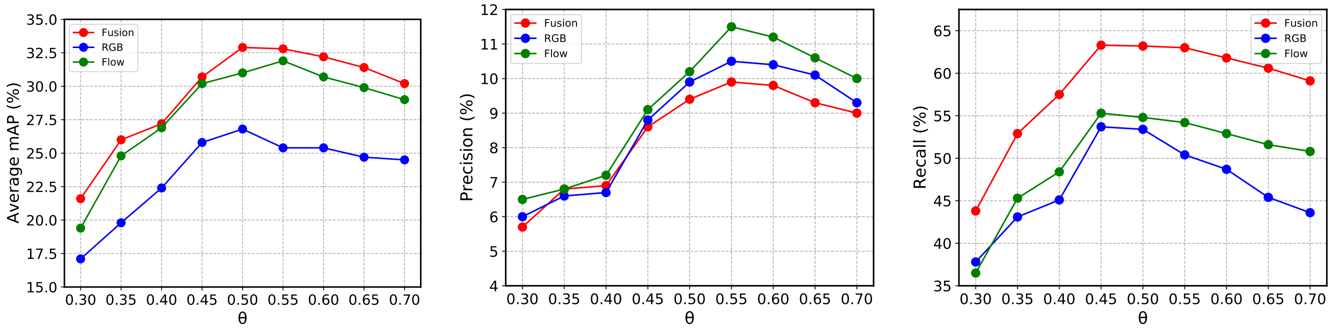


Fig. 4. Per category precision-recall (PR) curves on the THUMOS14 testing set. The PR curve is plotted under IoU threshold 0.3. The area enclosed by the PR curve, *x* axis and *y* axis is average precision (AP) of each category.

Ablation study on the uncertainty estimators. To mitigate the adverse effect caused by the noise of pseudo ground truth, we introduce a video-level uncertainty estimator and a snippet-level uncertainty estimator. They estimate the reliability of pseudo ground truth in a batch and in a video respectively, and thus decrease the weight for uncertain pseudo ground and increase the weight for confident ones. Table 7 summarizes the results, which demonstrate the usage of either uncertainty estimator improves the performance, and their combination leads to even higher performance. Specifically, the snippet-level uncertainty estimator has more impact than the video-level one. Moreover, symmetric KL divergence-based uncertainty estimators perform better than those using attention difference.

Sensitivity analysis on the thresholding parameter θ. The thresholding parameter θ in the hard pseudo ground truth generation has significant impact on the quality of the pseudo ground truth. Fig. 5(a), Fig. 5(b) and Fig. 5(c) plot the localization performance, precision and recall changes under different θ values, respectively. In Fig. 5(b), a relative large θ (*e.g.*, 0.55 and 0.6) helps remove false positive action proposals, and improve the precision, while a too large or small θ decreases the precision. In Fig. 5(c), a relatively small θ (*e.g.*, 0.45) helps retain more action proposals, which may contain some false negatives, and improve the recall, while a too small or large θ decreases the recall. Therefore, the localization performance in Fig. 5(a) shows a trade-off result between precision and recall, where the best performance



(a) Localization performance comparison under different θ values.

(b) Precision comparison under different θ values.

(c) Recall comparison under different θ values.

Fig. 5. Comparison between models trained with hard pseudo ground truth under different thresholding θ values.

TABLE 7
Ablation study on video-level and snippet-level uncertainty estimators.

Video Uncertainty	Snippet Uncertainty	mAP@IoU (%) 0.3	0.5	0.7	Avg (%) 0.3:0.1:0.7
None	None	50.9	32.8	11.9	32.1
	Diff	51.2	33.1	12.2	32.3
	KLD	51.5	33.2	12.3	32.4
Diff	None	51.1	32.8	11.8	32.1
	Diff	51.4	33.2	12.3	32.4
	KLD	51.7	33.3	12.4	32.6
KLD	None	51.0	33.0	12.0	32.2
	Diff	51.7	33.2	12.3	32.5
	KLD	52.1	33.6	12.7	32.9

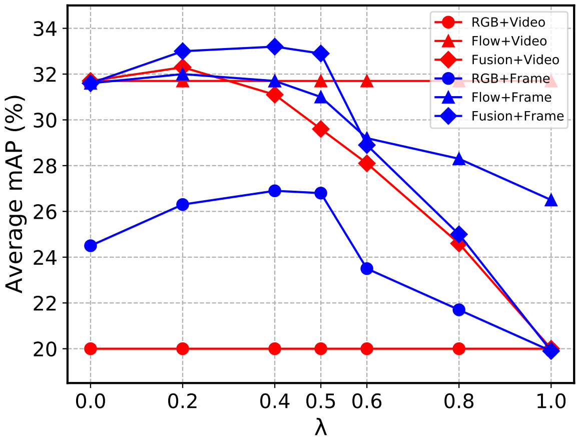


Fig. 6. Comparison between models under different fusion parameter λ on the THUMOS14 testing set.

is achieved under $\theta = 0.5$. To summarize, the localization performance, precision and recall show the same tendency: the performance is positively correlated with the distance between θ and 0.5.

Sensitivity analysis on the fusion parameter λ. λ is an important hyperparameter controlling the relative importance between the RGB stream and the flow stream at late fusion, and thus influences the fusion result and the pseudo ground truth. As shown in Fig. 6, with only video-level supervision, the late-fusion result outperforms both individual streams only when the stream that has higher performance dominates (*e.g.*, $\lambda = 0.2$). Under frame-level pseudo supervision, the localization performances of the RGB stream and the fusion result are greatly improved compared with those under only video-level supervision. However, when the noisy RGB stream predominates the pseudo ground truth (*i.e.*, $\lambda > 0.5$), the performance of the flow stream and the fusion result corrupt significantly. We also note that performances for $\lambda = 0.2$ and $\lambda = 0.4$ exceed the performance for $\lambda = 0.5$, as the noisy RGB prediction has lower weights than the more precise flow stream. That said, to demonstrate the generalization ability of our method, we use $\lambda = 0.5$ in later experiments.

TABLE 8
Results of the proposed method in the early-fusion framework.

Loss	Pseudo Label	mAP@IoU (%) 0.3	0.5	0.7	Avg (%) 0.3:0.1:0.7
-	-	31.6	16.8	5.4	16.7
$\mathcal{L}_{\text{norm}}$	-	37.6	22.2	6.0	22.1
$\mathcal{L}_{\text{a-norm}}$	-	39.3	23.7	7.1	23.3
$\mathcal{L}_{\text{a-norm}}$	Soft	40.2	24.1	7.3	23.7
	Hard	41.2	25.0	7.9	24.4

Interestingly, under frame-level pseudo supervision with $\lambda = 1$, (*i.e.*, only the RGB stream is used for pseudo ground truth generation), the flow stream still outperforms the RGB stream by a large margin, which demonstrates the RGB stream is insensitive to actions and lacks generalization ability.

Ablation study on the early-fusion framework. As we reviewed in Section 1, there are two mainstream two-stream fusion methods, *i.e.*, early fusion and late fusion. To demonstrate the effectiveness of the proposed method, we implement our method in the early-fusion framework, where the concatenation of the RGB and optical flow features on the feature dimension is fed into a single base model. The pseudo ground truth is generated from the single base model's attention sequence and used for iterative refinement.

The performance comparison in the early-fusion network is summarized in Table 8. Without pseudo ground truth, the results show the same tendency with the late-fusion framework: the original attention normalization loss greatly improves the baseline performance, and its adaptive version further boosts the performance, demonstrating its effectiveness in both early- and late-fusion frameworks. In the early-fusion framework, the pseudo ground truth requires the base model to output previous results under

TABLE 9
Hyperparameter sensitivity analysis.

(a) Sensitivity analysis on the attention normalization loss weight α. Results are reported w/o pseudo ground truth learning.

α	mAP@IoU (%)			Avg (%)
	0.3	0.5	0.7	0.3:0.1:0.7
0	33.1	19.0	5.7	18.2
0.05	47.8	29.8	10.3	28.9
0.1	47.9	30.3	10.7	29.6
0.2	48.3	30.2	10.4	29.5

(b) Sensitivity analysis on the smooth loss weight β. Results are reported w/o pseudo ground truth learning.

β	mAP@IoU (%)			Avg (%)
	0.3	0.5	0.7	0.3:0.1:0.7
0	46.8	29.8	10.5	29.2
0.05	47.5	30.1	10.5	29.5
0.1	47.9	30.3	10.7	29.6
0.2	47.0	29.7	10.5	29.2

(c) Sensitivity analysis on the pseudo ground truth learning loss weight γ.

γ	mAP@IoU (%)			Avg (%)
	0.3	0.5	0.7	0.3:0.1:0.7
0.1	51.7	33.3	12.6	32.5
0.2	52.0	33.3	12.4	32.6
0.5	51.9	33.4	12.8	32.8
1	52.1	33.6	12.7	32.9
2	52.0	33.7	12.3	32.8

different stochastic model noise (*e.g.*, dropout), and thus it improves the generalization ability and robustness of the base model. Therefore, both soft and hard pseudo ground truths improve the performance in the early-fusion framework, demonstrating their effectiveness. Furthermore, the hard pseudo ground truth also achieves higher performance than its soft counterpart, which agrees with the results in the late-fusion framework.

Hyperparameter sensitivity. To demonstrate the robustness of our method to hyperparameters, we present a set of ablation study in Table 9. The results reveal that our method is robust to loss weights for the adaptive attention normalization loss (Table 9(a)), the smooth loss (Table 9(b)), and the adaptive pseudo ground truth learning loss (Table 9(c)). Specifically, our smooth loss improves the performance at low loss weights, as it involves the temporal relationship in attention learning.

Qualitative analysis. Four representative examples of TAL results are plotted in Fig. 7 to illustrate the efficacy of the proposed pseudo supervision. In the first example, with only video-level labels, the RGB stream provides a worse localization result than the flow stream, and thus leads to a noisy fusion attention sequence. The pseudo ground truth guides the RGB stream to identify false positive action proposals and discover true action instances. It furthermore leads to a cleaner fusion attention sequence, where high activations correspond better to the ground truth. In the second example, with only video-level supervision, both streams have some non-overlapping false positive action proposals at the beginning of the video. In this case, the pseudo ground truth helps remove such false positives. In the third example, with only video-level supervision, the RGB stream can only distinguish certain scenes, and fails to separate proximate action instances. In contrast, the flow stream can precisely detect the ground truth action instance. Therefore, the pseudo ground truth helps the RGB stream to separate consecutive action instances. The last example shows a classic case of performance degradation. Both streams exhibit numerous false positives in the middle of the video. The false positives are mostly overlapped, and are reinforced in the pseudo ground truth, making the models trained with the pseudo ground truth more confident about the false positive. Eliminating such false positive action proposals, however, requires true ground truth supervision.

To summarize, the two modalities have their own strengths and limitations. The RGB stream is sensitive to appearance. Thus, it fails in scene shot from unusual angles or separating proximate action instances; the flow stream is sensitive to motion, and provides more accurate results, but it fails in slow or occluded motion. Qualitative results reveal that the pseudo ground truth helps two streams reach a consensus at most temporal locations. Therefore, the fusion attention sequence becomes cleaner and helps generate more precise action proposals and more reliable confidence scores.

5 Conclusion

In this paper, we propose an adaptive two-stream consensus network (A-TSCN) for W-TAL, which benefits from an adaptive attention normalization loss and an iterative refinement training approach. The adaptive attention normalization loss dynamically selects the action and background snippets in a video, and forces the attention to perform a binary selection, thus reducing the ambiguity between the foreground and background. The iterative refinement training scheme uses a novel frame-level pseudo ground truth as fine-grained supervision, and iteratively improves the two-stream base models. Meanwhile, a video-level uncertainty estimator and a snippet-level uncertainty estimator dynamically determine the learning weights for each video and snippet, thus mitigating the adverse effect caused by learning from noisy pseudo labels. Experiments on four benchmarks demonstrate the proposed A-TSCN outperforms current state-of-the-art methods, and verify our design intuition.

Acknowledgments

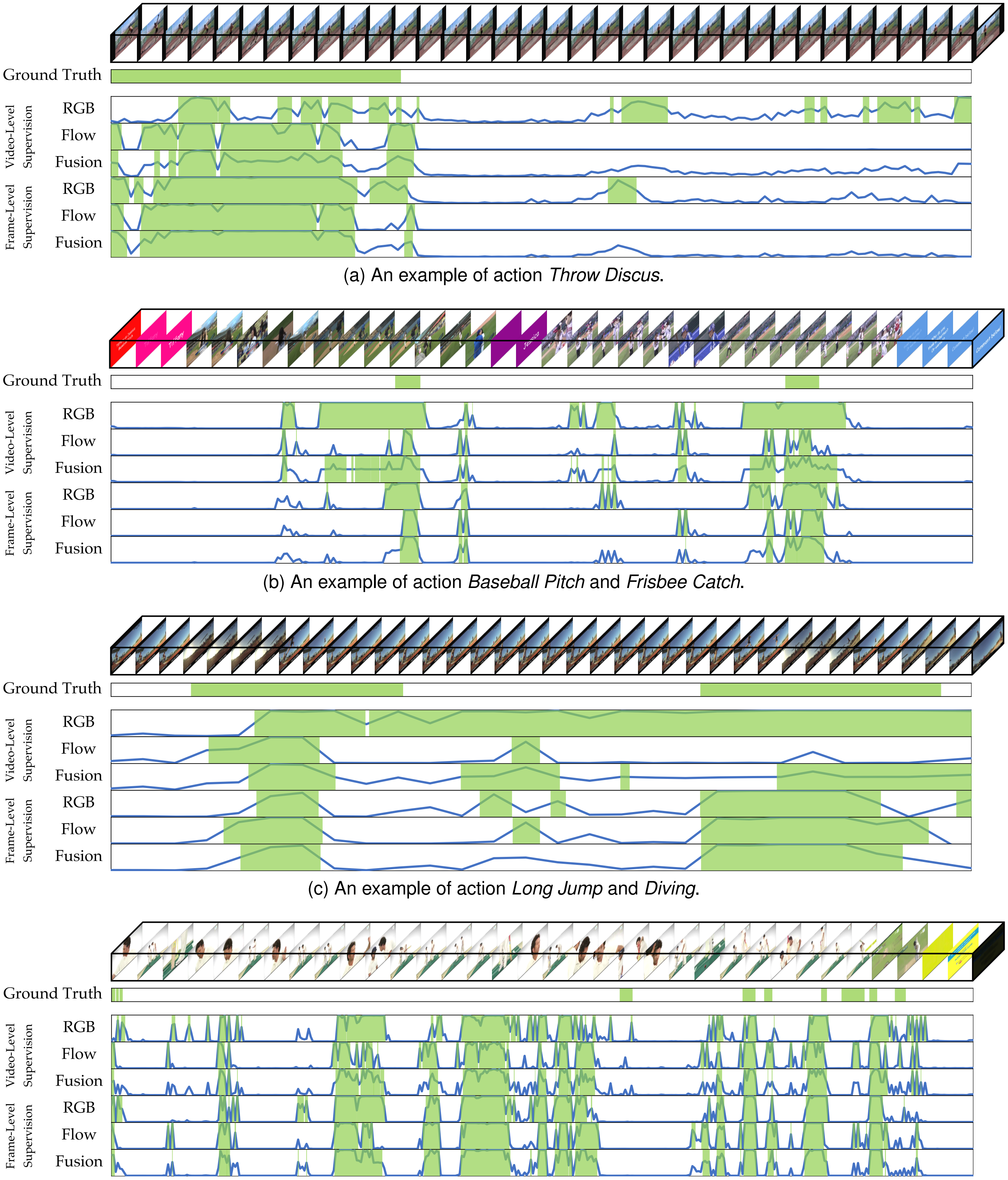
This work was supported partly by National Key R&D Program of China Grant 2018AAA0101400, NSFC Grants 62088102 and 61976171, Natural Science Foundation of Shaanxi Province under Grant 2022JC-41, and Fundamental Research Funds for the Central Universities under Grant XTR042021005.

This work is supported in part by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001120C0124. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Projects Agency (DARPA).

We thank valuable suggestions and feedback from Ziyi Liu at Wormpex AI Research, Beijing, China.

References

[1] K. Kumar Singh and Y. Jae Lee, "Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3524–3533. 1, 3



(a) An example of action *Throw Discus*.

(b) An example of action *Baseball Pitch* and *Frisbee Catch*.

(c) An example of action *Long Jump* and *Diving*.

(d) An example of action *Cricket Shot* and *Crickt Bowling*.

Fig. 7. Qualitative results on the THUMOS14 testing set. The eight rows in each example are input video, ground truth action instance, RGB stream, flow stream, and fusion attention sequences from the model trained with only video-level labels and frame-level pseudo ground truth, respectively. Green box denotes area whose attention activation is higher than 0.5. The horizontal and vertical axes are time and intensity of attention, respectively.

[2] L. Wang, Y. Xiong, D. Lin, and L. Van Gool, "UntrimmedNets for weakly supervised action recognition and detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4325–4334. 1, 3, 7, 8

[3] Z. Shou, H. Gao, L. Zhang, K. Miyazawa, and S.-F. Chang,

"AutoLoc: Weakly-supervised temporal action localization in untrimmed videos," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 154–171. 1, 3, 4, 6, 8

[4] P. Nguyen, T. Liu, G. Prasad, and B. Han, "Weakly supervised action localization by sparse temporal pooling network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6752–6761. 1, 3, 4, 5, 6, 8

[5] S. Paul, S. Roy, and A. K. Roy-Chowdhury, "W-TALC: Weakly-supervised temporal activity localization and classification," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 563–579. 1, 3, 4, 7, 8

[6] D. Liu, T. Jiang, and Y. Wang, "Completeness modeling and context separation for weakly supervised temporal action localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1298–1307. 1, 3, 4, 5, 8

[7] Y. Zhai, L. Wang, Z. Liu, Q. Zhang, G. Hua, and N. Zheng, "Action coherence network for weakly supervised temporal action localization," in *Proc. IEEE Int. Conf. Image Process.*, 2019, pp. 3696–3700. 1

[8] Z. Liu, L. Wang, Q. Zhang, Z. Gao, Z. Niu, N. Zheng, and G. Hua, "Weakly supervised temporal action localization through contrast based evaluation networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 3899–3908. 1, 3, 4, 8

[9] P. X. Nguyen, D. Ramanan, and C. C. Fowlkes, "Weakly-supervised action localization with background modeling," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 5502–5511. 1, 3, 4, 5, 6, 8

[10] S. Narayan, H. Cholakkal, F. S. Khan, and L. Shao, "3C-Net: Category count and center loss for weakly-supervised action localization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 8679–8687. 1, 4, 5

[11] T. Yu, Z. Ren, Y. Li, E. Yan, N. Xu, and J. Yuan, "Temporal structure mining for weakly supervised action detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 5522–5531. 1, 4, 8

[12] P. Lee, Y. Uh, and H. Byun, "Background suppression network for weakly-supervised temporal action localization," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 11 320–11 327. 1, 3, 4, 6, 7, 8, 9

[13] L. Huang, Y. Huang, W. Ouyang, L. Wang *et al.*, "Relational prototypical network for weakly supervised temporal action localization," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 11 053–11 060. 1, 4, 8

[14] B. Shi, Q. Dai, Y. Mu, and J. Wang, "Weakly-supervised action localization by generative attention modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1009–1019. 1, 3, 4, 7, 8

[15] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2921–2929. 1, 5

[16] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. on Syst., man, and Cybern.*, vol. 9, no. 1, pp. 62–66, 1979. 2, 5

[17] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 568–576. 2, 5

[18] Y. Zhai, L. Wang, W. Tang, Q. Zhang, J. Yuan, and G. Hua, "Two-stream consensus network for weakly-supervised temporal action localization," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 37–54. 2, 5, 7, 8, 9

[19] H. Zhao, A. Torralba, L. Torresani, and Z. Yan, "HACS: Human action clips and segments dataset for recognition and temporal localization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 8668–8678. 2, 7

[20] I. Laptev, "On space-time interest points," *Int. J. Comput. Vis.*, vol. 64, no. 2, pp. 107–123, 2005. 2

[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 886–893. 2

[22] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 428–441. 2

[23] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 3169–3176. 2

[24] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1933–1941. 3

[25] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6299–6308. 3, 4, 7

[26] Y. Zhao, Y. Xiong, and D. Lin, "Recognize actions by disentangling components of dynamics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6566–6575. 3

[27] N. Crasto, P. Weinzaepfel, K. Alahari, and C. Schmid, "MARS: Motion-augmented rgb stream for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7882–7891. 3

[28] Z. Shou, X. Lin, Y. Kalantidis, L. Sevilla-Lara, M. Rohrbach, S.-F. Chang, and Z. Yan, "DMC-Net: Generating discriminative motion cues for fast compressed video action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1268–1277. 3

[29] L. Wang, P. Koniusz, and D. Q. Huynh, "Hallucinating IDT descriptors and I3D optical flow features for action recognition with CNNs," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 8698–8708. 3

[30] A. Piergiovanni and M. S. Ryoo, "Representation flow for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9945–9953. 3

[31] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 20–36. 3

[32] C. Luo and A. L. Yuille, "Grouped spatial-temporal aggregation for efficient action recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 5512–5521. 3

[33] C. Yang, Y. Xu, J. Shi, B. Dai, and B. Zhou, "Temporal pyramid network for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 591–600. 3

[34] Y. Li, B. Ji, X. Shi, J. Zhang, B. Kang, and L. Wang, "TEA: Temporal excitation and aggregation for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 909–918. 3

[35] Y.-G. Jiang, J. Liu, A. R. Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar, "THUMOS challenge: Action recognition with a large number of classes," 2014. 3, 7

[36] A. Gorban, H. Idrees, Y.-G. Jiang, A. R. Zamir, I. Laptev, M. Shah, and R. Sukthankar, "THUMOS challenge: Action recognition with a large number of classes," 2015. 3

[37] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles, "ActivityNet: A large-scale video benchmark for human activity understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 961–970. 3, 7

[38] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 510–526. 3

[39] 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. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5734–5743. 3, 8

[40] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, X. Tang, and D. Lin, "Temporal action detection with structured segment networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2914–2923. 3, 7, 8, 9

[41] J. Gao, Z. Yang, and R. Nevatia, "Cascaded boundary regression for temporal action detection," in *Proc. Br. Mach. Vis. Conf.*, 2017. 3

[42] F. C. Heilbron, W. Barrios, V. Escorcia, and B. Ghanem, "SCC: Semantic context cascade for efficient action detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3175–3184. 3

[43] X. Dai, B. Singh, G. Zhang, L. S. Davis, and Y. Qiu Chen, "Temporal context network for activity localization in videos," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5793–5802. 3

[44] H. Xu, A. Das, and K. Saenko, "R-C3D: Region convolutional 3D network for temporal activity detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5783–5792. 3, 8

[45] 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. 3, 8

[46] Y.-W. Chao, S. Vijayanarasimhan, B. Seybold, D. A. Ross, J. Deng, and R. Sukthankar, "Rethinking the faster R-CNN architecture for temporal action localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1130–1139. 3, 7, 8

[47] J. Gao, Z. Yang, K. Chen, C. Sun, and R. Nevatia, "TURN TAP: Temporal unit regression network for temporal action proposals," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3628–3636. 3

[48] 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, 2017. 3

[49] F. Long, T. Yao, Z. Qiu, X. Tian, J. Luo, and T. Mei, “Gaussian temporal awareness networks for action localization,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 344–353. 3, 8

[50] T. Lin, X. Liu, X. Li, E. Ding, and S. Wen, “BMN: Boundary-matching network for temporal action proposal generation,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 3889–3898. 3, 8

[51] R. Zeng, W. Huang, M. Tan, Y. Rong, P. Zhao, J. Huang, and C. Gan, “Graph convolutional networks for temporal action localization,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7094–7103. 3, 8

[52] M. Xu, C. Zhao, D. S. Rojas, A. Thabet, and B. Ghanem, “G-TAD: Sub-graph localization for temporal action detection,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10 156–10 165. 3, 8

[53] Z. Zhu, W. Tang, L. Wang, N. Zheng, and G. Hua, “Enriching local and global contexts for temporal action localization,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 13 516–13 525. 3

[54] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *Proc. Int. Conf. Learn. Rep.*, 2017. 3

[55] M. Tan, Q. Shi, A. van den Hengel, C. Shen, J. Gao, F. Hu, and Z. Zhang, “Learning graph structure for multi-label image classification via clique generation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4100–4109. 3

[56] S.-J. Li, Y. AbuFarha, Y. Liu, M.-M. Cheng, and J. Gall, “MS-TCN++: Multi-stage temporal convolutional network for action segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020. 3

[57] K. Min and J. J. Corso, “Adversarial background-aware loss for weakly-supervised temporal activity localization,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 283–299. 3, 8

[58] Z. Luo, D. Guillory, B. Shi, W. Ke, F. Wan, T. Darrell, and H. Xu, “Weakly-supervised action localization with expectation-maximization multi-instance learning,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 729–745. 3, 8

[59] A. Pardo, H. Alwassel, F. Caba, A. Thabet, and B. Ghanem, “Refine-Loc: Iterative refinement for weakly-supervised action localization,” in *Proc. Winter Conf. Appl. Comput. Vis.*, 2021, pp. 3319–3328. 3, 8

[60] W. Wang, D. Tran, and M. Feiszli, “What makes training multi-modal classification networks hard?” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12 695–12 705. 4

[61] H. Scudder, “Probability of error of some adaptive pattern-recognition machines,” *IEEE Trans. Inf. Theory*, vol. 11, no. 3, pp. 363–371, 1965. 4

[62] D. Yarowsky, “Unsupervised word sense disambiguation rivaling supervised methods,” in *Proc. Annu. Meet. Assoc. Comput Linguist.*, 1995, pp. 189–196. 4

[63] E. Riloff and J. Wiebe, “Learning extraction patterns for subjective expressions,” in *Conf. Empir. Methods Nat. Lang. Process.*, 2003, pp. 105–112. 4

[64] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, “Self-training with noisy student improves ImageNet classification,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10 687–10 698. 4

[65] J. He, J. Gu, J. Shen, and M. Ranzato, “Revisiting self-training for neural sequence generation,” in *Proc. Int. Conf. Learn. Rep.*, 2020. 4

[66] Y. Zhai, L. Wang, D. Doermann, and J. Yuan, “Two-stream consensus network: Submission to HACS challenge 2021 weakly-supervised learning track,” *arXiv preprint arXiv:2106.10829*, 2021. 5

[67] A. Richard and J. Gall, “Temporal action detection using a statistical language model,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3131–3140. 8

[68] Z. Yuan, J. C. Stroud, T. Lu, and J. Deng, “Temporal action localization by structured maximal sums,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3684–3692. 8

[69] W. Luo, T. Zhang, W. Yang, J. Liu, T. Mei, F. Wu, and Y. Zhang, “Action unit memory network for weakly supervised temporal action localization,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 9969–9979. 8

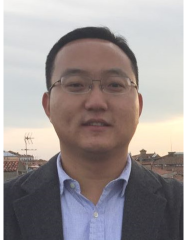
[70] W. Yang, T. Zhang, X. Yu, T. Qi, Y. Zhang, and F. Wu, “Uncertainty guided collaborative training for weakly supervised temporal action detection,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 53–63. 8

[71] C. Zach, T. Pock, and H. Bischof, “A duality based approach for realtime tv-l 1 optical flow,” *Pattern Recognit.*, pp. 214–223, 2007. 7

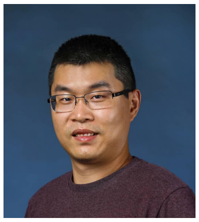
[72] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and F. F. Li, “ImageNet: A large-scale hierarchical image database,” *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 248–255, 2009. 7

[73] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “PyTorch: An imperative style, high-performance deep learning library,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8024–8035. 7

Yuanhao Zhai (Member, IEEE) received the B.Eng degree in Computer Science from Xi’an Jiaotong University, Xi’an, China in 2020. He is now a Ph.D. student with the State University of New York at Buffalo, USA. From 2018 to 2021, he was a research intern at the Institute of Artificial Intelligence and Robotics of Xi’an Jiaotong University, Xi’an, China. His research interests lie in computer vision and machine learning.



Le Wang (Senior Member, IEEE) received the B.S. and Ph.D. degrees in Control Science and Engineering from Xi’an Jiaotong University, Xi’an, China, in 2008 and 2014, respectively. From 2013 to 2014, he was a visiting Ph.D. student with Stevens Institute of Technology, Hoboken, New Jersey, USA. From 2016 to 2017, he is a visiting scholar with Northwestern University, Evanston, Illinois, USA. He is currently a Professor with the Institute of Artificial Intelligence and Robotics of Xi’an Jiaotong University, Xi’an, China. His research interests include computer vision, pattern recognition, and machine learning. He is an associate editor of Pattern Recognition Letters. He is an area chair of CVPR’2022, ICME’2022 and IAPR’2022, and a senior program committee member of AAAI’2022. He holds 8 China patents and has 22 more China patents pending. He is the author of more than 60 peer reviewed publications in prestigious international journals and conferences.



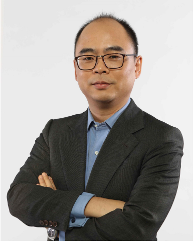
Wei Tang (Member, IEEE) received his Ph.D. degree in Electrical Engineering from Northwestern University, Evanston, Illinois, USA in 2019. He received the B.E. and M.E. degrees from Beihang University, Beijing, China, in 2012 and 2015 respectively. He is currently an Assistant Professor in the Department of Computer Science at the University of Illinois at Chicago. His research interests include computer vision, pattern recognition and machine learning.

Qilin Zhang (Member, IEEE) received the B.E. degree in Electrical Information Engineering from the University of Science and Technology of China, Hefei, China, in 2009, and the M.S. degree in Electrical and Computer Engineering from the University of Florida, Gainesville, Florida, USA, in 2011, and the Ph.D. degree in Computer Science from Stevens Institute of Technology, Hoboken, New Jersey, USA, in 2016. He is currently an Independent Researcher. He was a Senior Research Scientist (2020-2021) with ABB Corporate Research Center, Raleigh, NC, USA. Before that, he was a Senior Research Engineer (2016-2018) and then Lead Research Engineer (2018-2020) with HERE Technologies, Chicago, IL, USA. His research interests include computer vision and signal processing.

Nanning Zheng (Fellow, IEEE) graduated from the Department of Electrical Engineering of Xi'an Jiaotong University (XJTU), Xi'an, China, in 1975, received the M.E. degree in Information and Control Engineering from Xi'an Jiaotong University, Xi'an, China, in 1981, and the Ph.D. degree in electrical engineering from Keio University, Keio, Japan, in 1985. He is currently a Professor and the Director with the Institute of Artificial Intelligence and Robotics of Xi'an Jiaotong University, Xi'an, China. His research interests include computer vision, pattern recognition, computational intelligence, and hardware implementation of intelligent systems. Since 2000, he has been the Chinese representative on the Governing Board of the International Association for Pattern Recognition. He became a member of the Chinese Academy of Engineering in 1999.

David Doermann (Fellow, IEEE) is Professor of Empire Innovation and the Director of the Artificial Intelligence Institute the University at Buffalo (U.B.). Before coming to U.B. he was a Program Manager with the Information Innovation Office at the Defense Advanced Research Projects Agency (DARPA), where he developed, selected and oversaw research and transition funding in the areas of computer vision, human language technologies and voice analytics. From 1993 to 2018, David was a member of the research faculty at the University of Maryland, College Park. In his role in the Institute for Advanced Computer Studies, he served as Director of the Laboratory for Language and Media Processing, and as an adjunct member of the graduate faculty for the Department of Computer Science and the Department of Electrical and Computer Engineering. He and his group of researchers focus on many innovative topics related to analysis and processing of document images and video including triage, visual indexing and retrieval, enhancement and recognition of both textual and structural components of visual media. David has over 250 publications in conferences and journals, is a fellow of the IEEE and IAPR, has numerous awards, including an honorary doctorate from the University of Oulu, Finland and is a founding Editor-in-Chief of the International Journal on Document Analysis and Recognition.

Junsong Yuan (Fellow, IEEE) is Professor and Director of Visual Computing Lab at Department of Computer Science and Engineering (CSE), State University of New York at Buffalo, USA. Before joining SUNY Buffalo, he was Associate Professor (2015-2018) and Nanyang Assistant Professor (2009-2015) at Nanyang Technological University (NTU), Singapore. He obtained his Ph.D. from Northwestern University in 2009 (advised by Professor Ying Wu), M.Eng. from National University of Singapore in 2005, and B.Eng. from Huazhong University of Science Technology (HUST) in 2002. His research interests include computer vision, pattern recognition, video analytics, human action and gesture analysis, large-scale visual search and mining. He received Best Paper Award from IEEE Trans. on Multimedia, Nanyang Assistant Professorship from NTU, and Outstanding EECS Ph.D. Thesis award from Northwestern University. He served as Associate Editor of IEEE Trans. on Image Processing (T-IP), IEEE Trans. on Circuits and Systems for Video Technology (T-CSVT), Machine Vision and Applications (MVA), and Senior Area Editor of Journal of Visual Communications and Image Representation (JVCI). He was Program Co-Chair of IEEE Conf. on Multimedia Expo (ICME'18), and Area Chair for CVPR, ICCV, ECCV, and ACM MM. He was elected senator at both NTU and UB. He is a Fellow of IEEE and IAPR.



Gang Hua (Fellow, IEEE) was enrolled in the Special Class for the Gifted Young of Xi'an Jiaotong University (XJTU), Xi'an, China, in 1994 and received the B.S. degree in Automatic Control Engineering from XJTU in 1999. He received the M.S. degree in Control Science and Engineering in 2002 from XJTU, and the Ph.D. degree in Electrical Engineering and Computer Science at Northwestern University, Evanston, Illinois, USA, in 2006. He is currently the Vice President and Chief Scientist of Wormpex AI Research. Before that, he served in various roles at Microsoft (2015-18) as the Science/Technical Adviser to the CVP of the Computer Vision Group, Director of Computer Vision Science Team in Redmond and Taipei ATL, and Principal Researcher/Research Manager at Microsoft Research. He was an Associate Professor at Stevens Institute of Technology (2011-15). During 2014-15, he took an on leave and worked on the Amazon-Go project. He was an Visiting Researcher (2011-14) and a Research Staff Member (2010-11) at IBM Research T. J. Watson Center, a Senior Researcher (2009-10) at Nokia Research Center Hollywood, and a Scientist (2006-09) at Microsoft Live labs Research. He is an associate editor of TPAMI, TIP, TCSVT, CVIU, IEEE Multimedia, TVCJ and MVA. He also served as the Lead Guest Editor on two special issues in TPAMI and IJCV, respectively. He is a General Chair of ICCV'2025. He is a program chair of CVPR'2019&2022. He is an area chair of CVPR'2015&2017, ICCV'2011&2017, ICIP'2012&2013&2016, ICASSP'2012&2013, and ACM MM 2011&2012&2015&2017. He is the author of more than 190 peer reviewed publications in prestigious international journals and conferences. He holds 19 US patents and has 15 more US patents pending. He is the recipient of the 2015 IAPR Young Biometrics Investigator Award for his contribution on Unconstrained Face Recognition from Images and Videos, and a recipient of the 2013 Google Research Faculty Award. He is an IAPR Fellow and an ACM Distinguished Scientist.