Progressive Knowledge Distillation From Different Levels of Teachers for Online Action Detection

Md Moniruzzaman, *Graduate Student Member, IEEE*, and Zhaozheng Yin, *Senior Member, IEEE*

***Abstract*—In this paper, we explore the problem of Online Action Detection (OAD), where the task is to detect ongoing actions from streaming videos without access to video frames in the future. Existing methods achieve good detection performance by capturing long-range temporal structures. However, a major challenge of this task is to detect actions at a specific time that arrive with insufficient observations. In this work, we utilize the additional future frames available at the training phase and propose a novel Knowledge Distillation (KD) framework for OAD, where a teacher network looks at more frames from the future and the student network distills the knowledge from the teacher for detecting ongoing actions from the observation up to the current frames. Usually, the conventional KD regards a high-level teacher network (i.e., the network after the last training iteration) to guide the student network throughout all training iterations, which may result in poor distillation due to the large knowledge gap between the high-level teacher and the student network at early training iterations. To remedy this, we propose a novel progressive knowledge distillation from different levels of teachers (PKD-DLT) for OAD, where in addition to a high-level teacher, we also generate several low- and middle-level teachers, and progressively transfer the knowledge (in the order of low- to high-level) to the student network throughout training iterations, for effective distillation. Evaluated on two challenging datasets THUMOS14 and TVSeries, we validate that our PKD-DLT is an effective teacher-student learning paradigm, which can be a plug-in to improve the performance of the existing OAD models and achieve a state-of-the-art.**

***Index Terms*—Online action detection (OAD), knowledge distillation (KD), progressive knowledge distillation.**

I. Introduction

ACTION detection in untrimmed videos has been widely explored under offline settings [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], where the entire video is available for the detection at any moment. But, many real-time applications of computer vision such as human-robot collaboration, autonomous driving, and video surveillance require online action detection. Unlike offline action detection, *Online Action Detection (OAD)* aims at the task of detecting ongoing actions from streaming videos without access to video frames in the future.

Prior works [16], [17], [18] employed Recurrent Neural Networks (RNN) (e.g., Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU)) to encode the temporal dependencies of the observed frames for online action detection. But, the RNN-based methods have the problem of non-parallelism and are prone to forgetting informative history. To remedy this, recent works [19], [20] utilized transformer-like architectures to encode the observed frames in parallel with the multi-head self-attention mechanism. However, since online action detection detects what is happening at each frame based on the available observations up to the current time, a major challenge of this task is to detect actions at a specific time that arrive with insufficient observations.

Since detecting ongoing actions only from observations of the past and current frames (i.e., online action detection) is more challenging than the detection from the observations of past, current, and future frames (i.e., offline action detection), we utilize the additional future frames available at the training phase and formulate a knowledge distillation framework for OAD. Ordinarily, knowledge distillation is an effective technique for many computer vision tasks [21], [22], [23], [24], [25], [26], [27], [28], where a powerful teacher network with a large number of parameters transfers knowledge to a less-parameterized student network. However, different from the ordinary KD, in KD for online action detection, the difference between the teacher and student networks lies in their corresponding observations, i.e., the input data rather than the network architecture. As shown in Fig. 1(a), the KD framework in OAD involves: a *teacher network* that examines the past, current and future frames for detecting the action of the current frame; a *student network* for detecting the action of the current frame from the observations of the past and current frames; and a knowledge distillation mechanism to transfer the knowledge from the teacher network to the student network.

Usually, in the KD framework, a teacher network fully trained after the last training iteration (defined as *high-level teacher* in short) is naturally considered to guide the training of a student network throughout all iterations, as shown in Fig. 1(b). In other words, the student network at every training iteration distills the same knowledge from the high-level teacher. But, during the training of the student network, in the early training iterations, there is a large knowledge gap between the student network and the high-level teacher network. Therefore, only using a

Received 30 March 2024; revised 9 July 2024; accepted 14 August 2024. Date of publication 24 December 2024; date of current version 10 March 2025. This work was supported by the National Science Foundation under Grant CMMI-1954548, Grant ECCS-2025929, and Grant CMMI-2246673. The associate editor coordinating the review of this article and approving it for publication was Prof. Yuxin Peng. *(Corresponding author: Zhaozheng Yin.)*

Md Moniruzzaman is with the Department of Computer Science, Stony Brook University, Stony Brook, NY 11794 USA (e-mail: mmoniruzzama@cs.stonybrook.edu).

Zhaozheng Yin is with the Department of Computer Science and Department of Biomedical Informatics, Stony Brook University, Stony Brook, NY 11794 USA (e-mail: zyin@cs.stonybrook.edu).

Digital Object Identifier 10.1109/TMM.2024.3521772

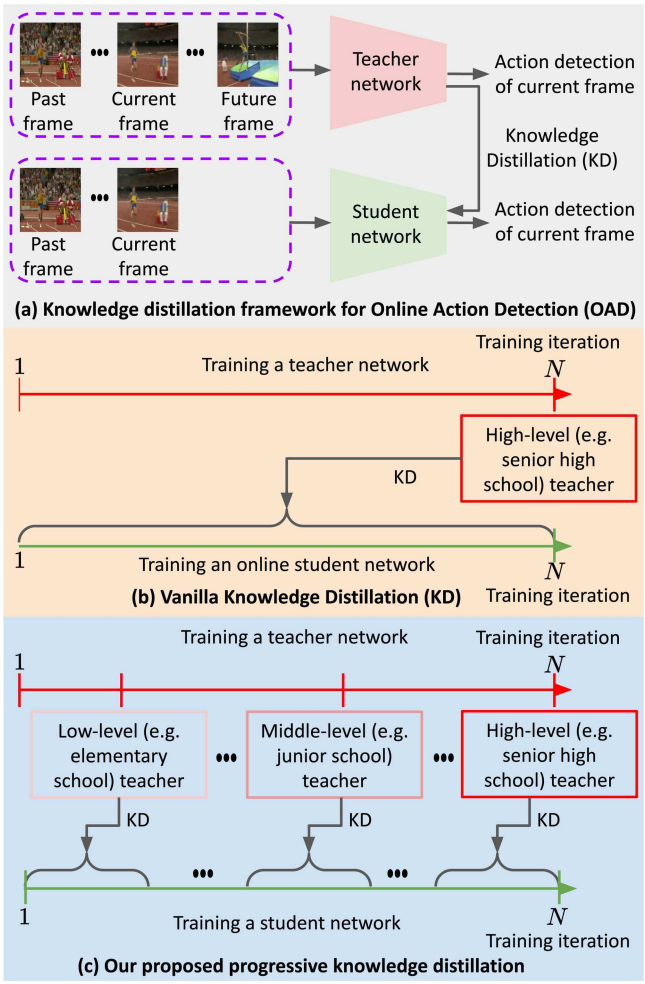


Fig. 1. (a) Knowledge Distillation (KD) for Online Action Detection (OAD). The KD framework for OAD involves: a teacher network that examines the past, current, and future frames for detecting the action of the current frame; a student network for detecting the action of the current frame only from the past and current frames; and a teacher-student learning mechanism for knowledge distillation from the teacher network to the student network; **(b) Vanilla Knowledge Distillation (KD).** A teacher network after the last training iteration (defined as *high-level teacher*) is considered to guide the student network throughout all training iterations; **(c) Our progressive knowledge distillation.** The student network progressively distills knowledge from different teachers in the order of low- to high-level teachers.

single high-level teacher to guide the student network throughout all training iterations may result in poor distillation. To remedy this, different levels of teachers are expected, which will progressively provide more transferrable knowledge to the student network for effective distillation.

In this paper, we propose a novel progressive knowledge distillation from different levels of teachers (PKD-DLT) for online action detection. Rather than using a single high-level teacher network to guide a student network throughout all iterations, we generate several low- and middle-level teachers in addition to the high-level teacher, and progressively transfer the knowledge from different teachers to the student network throughout the training iterations, as shown in Fig. 1(c). More specifically, we train a student network that takes the past and current frames as input, and progressively distills the knowledge from different teachers in the order of low- to high-level teachers, to accurately detect ongoing actions from streaming videos. The intuition of our progressive teacher-student learning approach can be explained as analogous to human education. For the first few grades, the student learns from teachers well-trained for elementary school, while the grades continue to increase, the student becomes more knowledgeable and gradually learns from teachers in middle school, junior high school, and so on. Similarly, our student network initially distills the knowledge from the low-level teacher. As the training iterations continue, the student becomes more knowledgeable and gradually distills knowledge from higher-level teachers. Please note that at inference time, we only use the student network, which detects the action of the current frame from the observations of the past and current frames, for online action detection.

Our main contributions are summarized as follows:

- We propose a novel progressive knowledge distillation from different levels of teachers (PKD-DLT) for online action detection, where we progressively transfer the knowledge (in the order of low- to high-level) from different levels of teachers to the student network throughout training iterations, for effective distillation.
- We validate the effectiveness of the proposed PKD-DLT on two popular benchmark datasets THUMOS14 and TVSeries. The experimental results demonstrate that our PKD-DLT is capable of learning a well-performed student network, which can be an effective plug-in to improve the performance of the previous online action detection models and achieve state-of-the-art.

II. Related Work

In this section, we review related works, including offline action detection, online action detection, and knowledge distillation.

Offline action detection: The goal of offline action detection is to localize the start time and end time of each action instance in untrimmed videos, where the entire video is available at any given moment. Most of the existing methods [1], [2], [3], [4], [5], [6], [29], [30], [31], [32], [33], [34], [35], [36] are trained in a fully-supervised manner, where the video-level action class labels along with the frame-wise detailed temporal annotations of each action instance are provided within the training videos. Since the fully-supervised approach requires a lot of annotation efforts, in contrast to the full-supervision-based methods, the research community pays a significant amount of attention to the weakly-supervised action detection [7], [8], [9], [10], [11], [12], [13], [37], [38], [39], [40], [41], [42], which attempts to localize action instances, leveraging only video-level supervision. However, both these fully and weakly supervised methods need to observe the entire video, which is not available in the online action detection task.

Online action detection: Different from offline action detection, the goal of online action detection is to detect ongoing actions from the observation of the current and past video frames [16], [19], [20], [43], [44], [45], [46], [47]. Geest et al. [44] defined the online action detection task for the first

time and introduced the TVSeries dataset. Later, they also proposed two-stream LSTM networks [16] to model the temporal structure for online action detection. IDN [17] manipulated the GRU cell to model the relationship between the current frame and the past frames. Recently, Temporal Recurrent Networks (TRN) [18] utilized LSTM blocks to anticipate future information and proposed TRN cells to combine the predicted future features with the current and past features to identify ongoing actions. More recently, OadTR [20] replaced the LSTM-based networks with a transformer-like architecture to encode the observed frames in parallel with the multi-head self-attention mechanism. Colar [47] considered historical frames as exemplars and utilized the exemplar consultation mechanism to model long-term dependencies for the online action detection task. GateHUB [48] introduced a model to more informatively leverage history and suppress background frames for online action detection. MAT [49] developed a memory anticipation-based approach, to address the online action detection and anticipation tasks. JOADAA [50] introduced a transformer encoder and transformer decoder-based framework to perform online action detection and action anticipation jointly.

However, since online action detection detects what is happening at each frame based on the available observations up to the current time, a major challenge of this task is to detect actions at a specific time that arrive with insufficient observations. To tackle this problem, in this work, we utilize the additional future frames available at the training phase and propose a novel knowledge distillation framework for online action detection, where a teacher network looks at more frames from the future and the online student network distills the knowledge from the teacher network for detecting ongoing actions from the observations up to the current frames. During the inference time, we only use the student network for online action detection.

Knowledge distillation: Knowledge distillation is a popular research topic in computer vision, which transfers knowledge from a cumbersome teacher to a small student network. Hinton et al. [51] first introduced the knowledge distillation concept to transfer knowledge of a large teacher network as additional supervision for training a smaller student network. Later, several works [25], [52], [53], [54], [55] introduced the transfer of soft-label distribution as knowledge, while some works [56], [57], [58], [59] transferred intermediate features. Recently, a variety of knowledge distillations such as graph-based knowledge distillation [22], [60], contrastive knowledge distillation [61], relational knowledge distillation [21], [62], and multi-modal knowledge distillation [23], [63], [64], [65] were adopted in different tasks.

More recently, some works [24], [66], [67], [68], [69] utilized multiple powerful teacher networks (ranked by parameter size) to guide a less-parameterized student network progressively. You et al. [67] trained a thin student network by incorporating distillation from cumbersome multiple-teacher networks for image classification. Park et al. [68] introduced a knowledge distillation framework to transfer feature-level ensemble knowledge from multiple teachers to a student network. Hao et al. [69] incorporated a student model with a multi-level feature-sharing structure that learns from multiple teacher models. [70] introduced a progressive knowledge distillation mechanism for fast sampling of diffusion models, which reduces the sampling time of diffusion models by distilling a trained deterministic diffusion sampler, using many steps, into a new diffusion model that takes half as many sampling steps. They progressively applied this distillation procedure to halve the number of required sampling steps each time. [71] proposed a progressive knowledge distillation mechanism that transfers intermediate supervision signals of a cumbersome teacher model into a lightweight student network. [72] introduced a progressive self-knowledge distillation, which progressively distills a model's own knowledge by combining the ground truth and the predictions from the preceding iteration. This approach essentially involves the student model transitioning into the role of the teacher. Xie et al. [73] proposed a capacity dynamic distillation framework, the student model is initially a heavy model to learn distilled knowledge fruitfully, and then the student model is gradually compressed. However, these progressive knowledge distillation mechanisms were mainly adopted in image classification, retrieval, and generation.

However, there are not many literature studies on knowledge distillation for online action detection. The privileged knowledge distillation (PKD) [43] is the only method on knowledge distillation for online action detection. PKD utilized multiple teacher networks with different observations to guide a student network, where all teacher networks are fully trained, i.e., high-level teachers. But, during the training of the student network, there is usually a large knowledge gap between the student network and the high-level teachers in the early training iterations. Thus, we propose a novel progressive knowledge distillation from different levels of teachers (PKD-DLT), where in addition to a high-level teacher, we also generate several low- and middle-level teachers, and progressively transfer the knowledge (in the order of low- to high-level) to the student network throughout training iterations, for effective distillation. Table I summarizes the innovations of our PKD-DLT compared to the existing PKD.

III. Method

In this section, first, we introduce the problem statement (Section III-A). Then, we present our vanilla knowledge distillation for online action detection (Section III-B). Finally, we introduce our progressive knowledge distillation from different levels of teachers for online action detection (Section III-C).

A. Problem Statement

Given a video stream that contains sequential various types of actions, online action detection aims to detect ongoing actions in real time without access to the video frames in the future. Formally, given a streaming video sequence $\mathbf{V} = \{v\}_{-L}^{0}$, our task is to classify the action in the current frame v_0 based on the observation of the past L frames and the current frame. We use $\mathbf{y}_0 \in \mathbb{R}^{C+1}$ to represent the action and background classes of the current frame v_0, where there are a total of C action classes and the $(C+1)$th class represents the background class. We tackle

TABLE I
EXISTING PRIVILEGED KNOWLEDGE DISTILLATION (PKD) [43] VS. OUR PROGRESSIVE KNOWLEDGE DISTILLATION FROM DIFFERENT LEVELS OF TEACHERS (PKD-DLT) FOR ONLINE ACTION DETECTION

Existing PKD	Our PKD-DLT
Existing PKD utilized multiple teachers to guide a student network, where all teachers are thoroughly or fully trained, i.e., high-level teachers. **Limitations:** 1) In the early training iterations, usually, there is a significant knowledge gap between the student network and the fully-trained (high-level) teachers. Therefore, using fully-trained (high-level) teachers to guide the student network throughout all training iterations may not always guide the student properly; and 2) It is a time-consuming process. It requires multiple fully trained teachers, eventually leading to training the student network multiple times.	We propose a novel progressive knowledge distillation from different levels of teachers (PKD-DLT), where in addition to a high-level teacher (network after the last training iteration), we also generate several low- and middle-level teachers from different respective training iterations, and progressively transfer the knowledge (in the order of low- to high-level) to the student network throughout training iterations, for effective distillation. Since we generate different levels of teachers from different training iterations and the student network also progressively distills the knowledge throughout training iterations, we need to train both teacher and student networks only once.

this problem by developing a novel progressive knowledge distillation framework that contains different levels of teachers to guide a student network for detecting ongoing actions in an online mode.

B. Vanilla Knowledge Distillation for Online Action Detection

The key concept of Knowledge Distillation (KD) for online action detection is training a student network to distill the *soft targets* (i.e., the labels that reduce the most confident value of the one-hot-vector and assign a small amount of probability mass to semantically similar actions) of a teacher network, where the difference between the teacher network and the student network lies in their corresponding input data rather than the network architecture. Both the teacher and student networks can be any state-of-the-art online action detection model, with KD as a plug-in for the training mechanism. More specifically, our vanilla KD is a two-stage approach:

- Firstly, we train a teacher network that detects ongoing action from the observations of the past, current, and future frames, $\mathbf{T} : \{v\}_{-L}^{L} \rightarrow \tilde{\mathbf{y}}_0^{\mathbf{T}}$, where $\tilde{\mathbf{y}}_0^{\mathbf{T}} \in \mathbb{R}^{C+1}$ represents the classification scores of the current frame v_0 for the teacher network.
- Secondly, we freeze the parameters of the teacher network and transfer its predictive capability to the student network through knowledge distillation. In other words, we train a student network that distills the knowledge from the teacher network and detects ongoing action from the observations of the past and current frames, $\mathbf{S} : \{v\}_{-L}^{0} \rightarrow \tilde{\mathbf{y}}_0^{\mathbf{S}}$, where $\tilde{\mathbf{y}}_0^{\mathbf{S}} \in \mathbb{R}^{C+1}$ represents the classification scores of the current frame v_0 for the student network.

Since the teacher model observes more frames in the future to make a decision, the soft target from the teacher model can transfer "dark knowledge" containing privileged information on similarity among different action categories to enhance the learning of the student network. To achieve this goal, the vanilla KD is formulated by minimizing the Kullback-Leibler (KL) divergence between the prediction of the student and the teacher, as follows:

$$\mathcal{L}_{\mathbf{T}\rightarrow\mathbf{S}}^{KD} = \mathrm{KL}(\mathrm{softmax}(\tilde{\mathbf{y}}_0^{\mathbf{S}}/\tau), \mathrm{softmax}(\tilde{\mathbf{y}}_0^{\mathbf{T}}/\tau)) \quad (1)$$

where the right arrow in the subscript indicates the teaching direction. τ is the temperature parameter to control the softening of logits. In addition to the distillation loss, the student network has its own online action detection loss $\mathcal{L}_{\mathbf{S}}^{OAD}$, which is usually a cross-entropy loss between the predicted and ground-truth labels of the current frame:

$$\mathcal{L}_{S}^{OAD} = \mathrm{CE}(\mathrm{softmax}(\tilde{\mathbf{y}}_0^{\mathbf{S}}), \mathbf{y}_0) \quad (2)$$

As a result, the student's total loss is derived as follows:

$$\mathcal{L}_{\mathbf{S}} = \mathcal{L}_{\mathbf{S}}^{OAD} + \mathcal{L}_{\mathbf{T}\rightarrow\mathbf{S}}^{KD} \quad (3)$$

Please note that we only use the online action detection loss to train the teacher network:

$$\mathcal{L}_{\mathbf{T}} = \mathcal{L}_{\mathbf{T}}^{OAD} = \mathrm{CE}(\mathrm{softmax}(\tilde{\mathbf{y}}_0^{\mathbf{T}}), \mathbf{y}_0) \quad (4)$$

C. Progressive Knowledge Distillation From Different Levels of Teachers for Online Action Detection

Usually, in the KD framework, a high-level teacher network is considered to guide the student network throughout all training iterations. But, at the early training stage, there is a large knowledge gap between the student network and the high-level teacher network. Therefore, using a single high-level teacher to guide the student network throughout all training iterations may not always guide the student properly. To remedy this, in this paper, rather than only using a single high-level teacher, we also generate several low- and middle-level teachers and progressively transfer knowledge from different teachers to the student network throughout training iterations for effective knowledge distillation. More specifically, we design our progressive knowledge distillation from different levels of teachers (PKD-DLT), as follows:

- Similar to vanilla KD, we first train a teacher network that detects ongoing action from the observations of the past, current, and future frames. However, different from vanilla KD, rather than only storing the parameters of the high-level teacher, i.e., the parameters of the last training iteration, we also uniformly store the parameters of several low- and middle-level teachers. Formally, from N training iterations, we uniformly select D number of teachers with $n = N/D$ training iteration gap between two consecutive

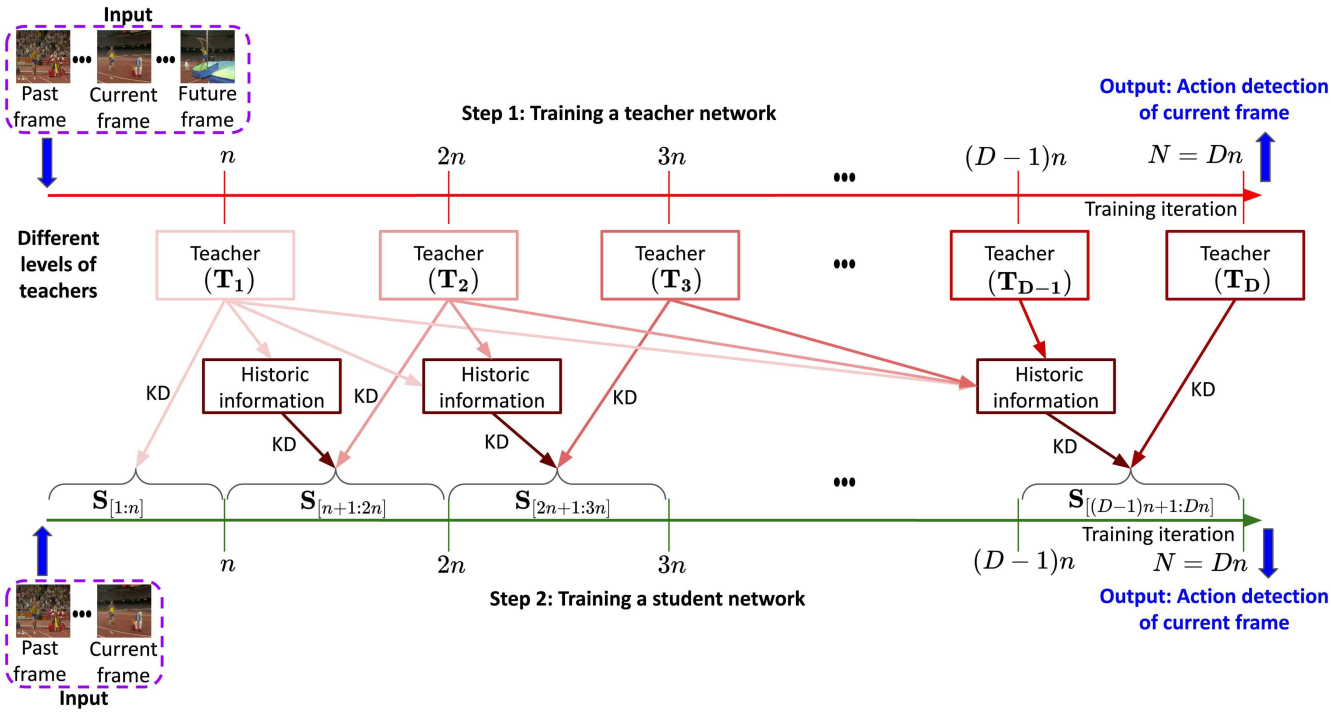


Fig. 2. Illustration of our progressive knowledge distillation mechanism. In the first step, a teacher network is trained to detect the action of the current frame from the observation of the past, current, and future frames. Rather than only storing the parameters of the high-level teacher, the parameters of several low- and middle-level teachers are also stored uniformly with n training iteration gap to different levels of teachers. In the second step, a student network is trained to detect the action of the current frame from the observation of the past and current frames. During the training, the student network progressively distills the knowledge from different teachers in the order of low-level $\mathbf{T}_1$ to high-level $\mathbf{T}_D$ teachers. For the first few iterations, the student network distills the knowledge from the low-level teacher $\mathbf{T}_1$. As the training iterations continue, the student becomes more knowledgeable and gradually distills the knowledge from more higher-level teachers and also distills the historic information from old teachers. During the inference, we only use the student network for our online action detection.

teachers. In other words, we store the parameters of the first teacher $\mathbf{T}_1$ at the n-th iteration, the second teacher $\mathbf{T}_2$ at the $2n$-th iteration, and so on.

- Secondly, we train a student network that takes only the observed past and current frames as input and progressively distills the knowledge throughout training iterations from different teachers in the order of the low-level teacher $\mathbf{T}_1$ to the high-level teacher $\mathbf{T}_D$, to accurately detect ongoing actions in real-time.

As shown in Fig. 2, to progressively distill the knowledge from different teachers in the order of lower-level to higher-level teachers, we first divide the N training iterations of the student network into D chunks, where each chunk contains $n = N/D$ iterations. During the first chunk, i.e., training the student network $\mathbf{S}_{[1:n]}$ from iteration 1 to n, the student network distills the knowledge from the low-level teacher $\mathbf{T}_1$. As the training iterations continue, the student becomes more knowledgeable and gradually distills knowledge from more higher-level teachers. Although the student becomes more knowledgeable and gradually interacts with new teachers, we let the student not forget the knowledge from old teachers. Therefore, in addition to distilling the knowledge from a new teacher, the student network also distills the historic knowledge from old teachers. Formally, during the training of the d-th chunk, the student network $\mathbf{S}_{[(d-1)n+1:dn]}$ distills the knowledge from the d-th teacher $\mathbf{T}_d$ and the historic information from old teachers $\mathbf{T}_1$ to $\mathbf{T}_{d-1}$. This progressive knowledge distillation process for D training chunks can be formulated as:

$$\begin{aligned}\mathcal{L}^{KD}_{\mathbf{S}_{[1:n]}} &= \mathcal{L}^{KD}_{\mathbf{T}_1 \to \mathbf{S}_{[1:n]}} \\ \mathcal{L}^{KD}_{\mathbf{S}_{[(d-1)n+1:dn]}} &= \lambda_1 \mathcal{L}^{KD}_{\mathcal{H}(\mathbf{T}_1,\ldots,\mathbf{T}_{d-1}) \to \mathbf{S}_{[(d-1)n+1:dn]}} \\ &\quad + \lambda_2 \mathcal{L}^{KD}_{\mathbf{T}_d \to \mathbf{S}_{[(d-1)n+1:dn]}}, \quad d = 2, \ldots, D\end{aligned} \tag{5}$$

where λ_1 and λ_2 are the balancing hyper-parameters. $\mathcal{H}$ represents the historic information from old teachers. More specifically, we obtain historic information from the fusion of the old teachers' logits, which generates a more smoothed label for regularization, as follows:

$$\begin{aligned}\mathcal{H}(\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_k) &= \alpha \mathbf{T}_k(\{v\}^L_{-L}) \\ &\quad + \sum_{i=1}^{k-1} \left(\frac{1-\alpha}{k-1}\right) \mathbf{T}_i(\{v\}^L_{-L})\end{aligned} \tag{6}$$

where α is a combination factor $\alpha \in [0, 1]$. Finally, the student's total loss is derived as follows:

$$\mathcal{L}_{\mathbf{S}} = \mathcal{L}^{OAD}_{\mathbf{S}} + \sum_{d=1}^{D} \mathcal{L}^{KD}_{\mathbf{S}_{[(d-1)n+1:dn]}}, \tag{7}$$

where $\mathcal{L}_{\mathrm{S}}^{OAD}$ is from (2) and we use the OAD loss (4) to train the teacher network. During the inference, we only use the student network to detect the current ongoing actions from the observations of the past and current frames.

Note, our PKD-DLT may seem to be conceptually similar to the moving-averaged KD. In moving-averaged KD, the teacher is formulated from the student network. More specifically, the teacher network is an average of consecutive student networks throughout training iterations. However, the moving-averaged KD is not an effective solution for online action detection. The main reason is that when the student network detects an action from insufficient observation, the teacher network will also need to detect that action from insufficient observation, which may result in a wrong detection for both networks. Thus, we propose a novel PKD-DLT, where a teacher network looks at more frames from the future for sufficient observations and the student network progressively distills the knowledge from different levels of teachers for detecting actions from the observation up to the current frames. During the inference, we only use the student network to detect the current ongoing actions from the observations of the past and current frames.

IV. Experiments

A. Datasets

We conduct our experiments on two benchmark datasets that are widely used in the community of online action detection: THUMOS14 [74] and TVSeries [44].

THUMOS14 [74]: THUMOS14 has annotations for 200 validation videos and 213 testing videos for the online action detection task, which belong to 20 classes from sports videos. This dataset includes drastic intra-category varieties, motion blur, significant changes in the length of the action instances (from less than a second to minutes), and many background frames, which have diverse contexts and variations in motion patterns. All of these properties make this dataset challenging for online action detection. Following the literature [17], [19], [20], [75], we train our model on the validation set and evaluate on the test set.

TVSeries [44]: TVSeries includes 27 episodes of 6 popular TV series with a total duration of about 16 hours, which is annotated with 30 daily actions (e.g., run, drink, etc.). The online action detection on this dataset is challenging since this dataset contains many unconstrained perspectives, a large proportion of background frames, and temporal overlapping action instances. We follow the train-test splits provided by the dataset to evaluate our method.

Evaluation metric: Following the literature [17], [18], [20], [47], [76], we use per-frame mean average precision (mAP) and mean calibrated average precision (mcAP) to evaluate the performance of online action detection on THUMOS14 and TVSeries datasets, respectively.

B. Implementation Details

Our proposed PKD-DLT can be used as a plug-in to any state-of-the-art online action detection model. To show the effectiveness of our PKD-DLT, we plug it into three latest online action detection methods: OadTR [20], Colar [47], and MAT [49]. For the feature extractor, following the literature [17], [18], [20], [47], [76], we use the same two-stream network [77], whose spatial stream adopts ResNet-200 [78] and temporal stream adopts BN-Inception [79]. Similar to existing methods, we report the performances of two experimental settings, where the two-stream is pre-trained on either ActivityNet v1.3 [80] (TSN-Anet) or Kinetics-400 [81] (TSN-Kinetics), for a fair comparison. By validation, we set $\tau = 3$, $\alpha = 0.8$, $n = 200$, $\lambda_1 = 0.2$, $\lambda_2 = 0.7$, and $L = 63$.

TABLE II
Results on THUMOS14

Method	Feature	mAP
RED [76], BMVC2017	TSN-Anet	45.3
TRN [18], ICCV2019	TSN-Anet	47.2
IDN [17], CVPR2020	TSN-Anet	50.0
OadTR [20], ICCV2021	TSN-Anet	58.3
Colar [47], CVPR2022	TSN-Anet	59.4
GateHub [48], CVPR2022	TSN-Anet	69.1
MAT [49], ICCV2023	TSN-Anet	70.4
OadTR [20] + **PKD-DLT**	TSN-Anet	**61.2** (2.9 ↑)
Colar [47] + **PKD-DLT**	TSN-Anet	**62.1** (2.7 ↑)
MAT [49] + **PKD-DLT**	TSN-Anet	**72.8** (2.4 ↑)
IDN [17], CVPR2020	TSN-Kinetics	60.3
PKD [43], PR2020	TSN-Kinetics	64.5
OadTR [20], ICCV2021	TSN-Kinetics	65.2
Colar [47], CVPR2022	TSN-Kinetics	66.9
Colar [47] + moving-averaged KD	TSN-Kinetics	67.5
GateHub [48], CVPR2022	TSN-Kinetics	70.7
MAT [49], ICCV2023	TSN-Kinetics	71.6
JOADAA [50], WACV2024	TSN-Kinetics	72.6
OadTR [20] + **PKD-DLT**	TSN-Kinetics	**67.8** (2.6 ↑)
Colar [47] + **PKD-DLT**	TSN-Kinetics	**69.1** (2.2 ↑)
MAT [47] + **PKD-DLT**	TSN-Kinetics	**73.6** (2.0 ↑)

We plug our proposed PKD-DLT into three latest online action detection methods OadTR [20], colar [47], and MAT [49] to show its effectiveness to boost the performance of existing methods and achieve state-of-the-art performance.

C. Comparison With State-of-The-Arts

In this subsection, we compare the performance of our proposed PKD-DLT with other state-of-the-art methods on THUMOS14 and TVSeries datasets.

THUMOS14: Table II summarizes the results of existing state-of-the-art online action detection methods on the THUMOS14 dataset. We plug our PKD-DLT into three latest online action detection methods OadTR [20], Colar [47], and MAT [49]. As shown in Table II, based on TSN-Anet features, our PKD-DLT brings an mAP gain of $+2.9\%$ over OadTR [20], an mAP gain of $+2.7\%$ over Colar [47], and an mAP gain of $+2.4\%$ over MAT [49]. On the other hand, we also find that our PKD-DLT can boost the performance of OadTR [20], Colar [47], and MAT [49] by $+2.6\%$, $+2.2\%$, and $+2.0\%$, respectively, when the comparison is based on TSN-Kinetics features.

TABLE III
RESULTS ON TVSERIES

Method	Feature	mcAP
LRCN [82], CVPR2015	RGB	64.1
RED [76], BMVC2017		71.2
2S-FN [16], WACV2018		72.4
TRN [18], ICCV2019		75.4
IDN [17], CVPR2020		76.6
FV-SVM [16], WACV2018	Flow	74.3
IDN [17], CVPR2020		80.3
RED [76], BMVC2017	TSN-Anet	79.2
TRN [18], ICCV2019		83.7
IDN [17], CVPR2020		84.7
PKD [43], PR2020		86.4
OadTR [20], ICCV2021		85.4
Colar [47], CVPR2022		86.0
GateHub [48], CVPR2022		88.4
MAT [49], ICCV2023		88.6
OadTR [20] + **PKD-DLT**	TSN-Anet	**86.8** (1.4 ↑)
Colar [47] + **PKD-DLT**	TSN-Anet	**87.1** (1.1 ↑)
MAT [49] + **PKD-DLT**	TSN-Anet	**89.6** (1.0 ↑)
IDN [17], CVPR2020	TSN-Kinetics	86.1
OadTR [20], ICCV2021		87.2
Colar [47], CVPR2022		88.1
Colar [47] + moving-averaged KD		88.3
GateHub [48], CVPR2022		89.6
MAT [49], ICCV2023		89.7
OadTR [20] + **PKD-DLT**	TSN-Kinetics	**88.4** (1.2 ↑)
Colar [47] + **PKD-DLT**	TSN-Kinetics	**89.1** (1.0 ↑)
MAT [47] + **PKD-DLT**	TSN-Kinetics	**90.5** (0.8 ↑)

We plug our proposed PKD-DLT into three latest online action detection methods OadTR [20], colar [47], and MAT [49] to show its effectiveness to boost the performance of existing methods and achieve state-of-the-art performance.

TVSeries: We also conduct the same experiment on the TVSeries [44] dataset to confirm the generality of the proposed plug-in. As shown in Table III, for the TSN-Anet feature input, we obtain performance gains of $+1.4\%$, $+1.1\%$, and $+1.0\%$ by incorporating the proposed PKD-DLT into OadTR [20], Colar [47], and MAT [49] respectively, while we also achieve improvements of $+1.2\%$, $+1.0\%$, and $+0.8\%$ over these three methods, respectively, with the TSN-Kinetics feature input.

Performance under different action portions: Since one of the most important characteristics of online action detection is to detect an action at an early stage, we further verify the effectiveness of our plug-in on existing methods at different action stages. Table IV shows the online action detection performance on the TVSeries dataset when different action stages are observed. For example, the mcAP value within the action stage 40%–50% represents how accurately the model can detect an action when observing 40%–50% of the action. Since there are significant changes in the length of the action instances (from less than a second to minutes) in the THUMOS14 dataset, evaluating performance across different action portions for instances that last barely milliseconds becomes impractical. Therefore, following the literature [17], [18], [20], [43], [47], we report the online action detection performance under different action portions on the TVSeries dataset. The results show that our PKD-DLT can boost the performance of OadTR [20] and Colar [47] at all action stages, for both the ActivityNet and Kinetics features. This demonstrates the superiority of our PKD-DLT in improving the online action detection performances at early stages as well as all stages.

Existing knowledge distillation for online action detection vs. our knowledge distillation: We compare our proposed knowledge distillation method (PKD-DLT) with the latest knowledge distillation-based method (PKD [43]) for online action detection. Note, the privileged knowledge distillation (PKD) [43] is the only method for knowledge distillation in online action detection so far. As shown in Table II to Table IV, our PKD-DLT achieves superior performances compared to existing PKD [43] on both THUMOS14 and TVSeries datasets, on all metrics. For example, on the THUMOS14 dataset (Table II), our PKD-DLT on Colar achieves 69.1% mAP, compared to PKD [43] with 64.5% mAP.

In terms of efficiency, the existing PKD [43] is also a time-consuming process. It requires multiple fully trained teachers, eventually leading to training the student network multiple times. On the other hand, we generate different levels of teachers from different training iterations, and the student network also progressively distills the knowledge throughout training iterations. Therefore, we need to train both teacher and student networks only once, which is a more efficient training process compared to the existing PKD [43].

Moving-averaged KD vs. our PKD-DLT: Since our PKD-DLT may seem to be conceptually similar to the moving-averaged KD, we also compare our PKD-DLT with the moving-averaged KD to show the effectiveness of our PKD-DLT over the moving-averaged KD. For this comparison, we first plug the moving-averaged KD into Colar [47], and then replace it with our PKD-DLT. As shown in Table II and Table III, our PKD-DLT achieves better performances compared to the moving-averaged KD on both THUMOS14 and TVSeries datasets, respectively. The key factor contributing to this performance improvement lies in the difference in how the teacher networks operate. In the moving-averaged KD, the teacher network is an average of consecutive student networks throughout training iterations. Therefore, if the student network detects an action based on insufficient observation, the teacher network also identifies that action with insufficient observation. In contrast, the teacher network in our PKD-DLT examines more frames from the future, ensuring sufficient observations. The student network then progressively distills knowledge from various levels of teachers, which results in more effective distillation and, consequently, a notable improvement in detection performances. For example, on the THUMOS14 dataset (Table II), our PKD-DLT on Colar achieves 69.1% mAP, outperforming the moving-averaged KD on Colar, which attains 67.5% mAP.

D. Ablation Study

Ablation study on different numbers of iteration gaps: We conduct experiments to examine how different iteration gaps between two consecutive teachers (i.e., parameter n in Fig. 2)

TABLE IV
RESULTS ON TVSERIES WHEN ONLY PORTIONS OF ACTIONS ARE CONSIDERED (E.G., 40%−50% MEANS ONLY FRAMES OF THIS RANGE OF ACTION INSTANCES ARE USED TO COMPUTE MCAP AFTER DETECTING CURRENT ACTIONS ON ALL FRAMES IN AN ONLINE MANNER)

Method	Portion of actions									
	0%-10%	10%-20%	20%-30%	30%-40%	40%-50%	50%-60%	60%-70%	70%-80%	80%-90%	90%-100%
CNN [44]	61.0	61.0	61.2	61.1	61.2	61.2	61.3	61.5	61.4	61.5
LSTM [44]	63.3	64.5	64.5	64.3	65.0	64.7	64.4	64.4	64.4	64.3
FV-SVM [44]	67.0	68.4	69.9	71.3	73.0	74.0	75.0	75.4	76.5	76.8
TRN [18]	78.8	79.6	80.4	81.0	81.6	81.9	82.3	82.7	82.9	83.3
IDN [17]	80.6	81.1	81.9	82.3	82.6	82.8	82.6	82.9	83.0	83.9
OadTR [20]	79.5	83.9	86.4	85.4	86.4	87.9	87.3	87.3	85.9	84.6
Colar [47]	80.2	84.4	87.1	85.8	86.9	88.5	88.1	87.7	86.6	85.1
OadTR [20] + **PKD-DLT**	**81.2** (1.7 ↑)	**85.4** (1.5 ↑)	**87.7** (1.3 ↑)	**86.7** (1.3 ↑)	**87.6** (1.2 ↑)	**89.0** (1.1 ↑)	**88.5** (1.2 ↑)	**88.6** (1.3 ↑)	**87.5** (1.6 ↑)	**86.2** (1.6 ↑)
colar [47] + **PKD-DLT**	**81.6** (1.4 ↑)	**85.7** (1.3 ↑)	**88.2** (1.1 ↑)	**86.8** (1.0 ↑)	**87.8** (0.9 ↑)	**89.3** (0.8 ↑)	**89.0** (0.9 ↑)	**88.7** (1.0 ↑)	**87.7** (1.1 ↑)	**86.3** (1.2 ↑)
IDN * [17]	81.7	81.9	83.1	82.9	83.2	83.2	83.2	83.0	83.3	86.6
PKD * [43]	82.1	83.5	86.1	87.2	88.3	88.4	89.0	88.7	88.9	87.7
OadTR * [20]	81.2	84.9	87.4	87.7	88.2	89.9	88.9	88.8	87.6	86.7
Colar * [47]	82.3	85.7	88.6	88.7	88.8	91.2	89.6	89.9	88.6	87.3
OadTR [20] + **PKD-DLT** *	**82.7** (1.5 ↑)	**86.3** (1.4 ↑)	**88.7** (1.3 ↑)	**88.8** (1.1 ↑)	**89.2** (1.0 ↑)	**90.9** (1.0 ↑)	**90.0** (1.1 ↑)	**89.9** (1.1 ↑)	**88.9** (1.3 ↑)	**88.1** (1.4 ↑)
Colar [47] + **PKD-DLT** *	**83.6** (1.3 ↑)	**86.8** (1.1 ↑)	**89.6** (1.0 ↑)	**89.7** (1.0 ↑)	**89.9** (1.1 ↑)	**92.0** (0.8 ↑)	**90.5** (0.9 ↑)	**90.9** (1.0 ↑)	**89.6** (1.0 ↑)	**88.5** (1.2 ↑)

We plug our PKD-DLT into OadTR [20] and colar [47], and our approach improves the action detection performances at early stages as well as all stages. * Means the feature extractor is pre-trained on kinetics.

TABLE V
ABLATION STUDY ON DIFFERENT NUMBERS OF ITERATION GAPS

# Iteration gap	OadTR [20] + PKD-DLT		Colar [47] + PKD-DLT		Memory consumption
	THU-MOS14	TV-Series	THU-MOS14	TV-Series	
$n = 50$	60.1	86.1	61.2	86.6	$\sim$ 8.4 GB
$n = 100$	60.7	86.4	61.6	86.8	$\sim$ 4.2 GB
$n = 150$	60.9	86.6	61.7	86.9	$\sim$ 3.0 GB
$n = 200$	**61.2**	**86.8**	**62.1**	**87.1**	$\sim$ 2.1 GB
$n = 250$	61.0	86.7	61.9	87.0	$\sim$ 1.8 GB
$n = 300$	60.5	86.4	61.5	86.7	$\sim$ 1.5 GB

affect online action detection performance. As shown in Table V, when the iterations gap is relatively small (e.g., 50), the knowledge gap between two consecutive teachers is not significant and the performance gain from multi-level teachers is relatively low. As the iteration gap gradually increases, the teachers become more knowledgeable compared to their predecessors, and performance improves. However, when it exceeds a specific value (e.g., $n = 200$), there is a large knowledge gap between two consecutive teachers, eventually, a large knowledge gap between the student and teacher, which results in a distillation performance drop.

Memory efficiency and computational cost: Since we save model parameters for different levels of teachers, memory consumption can be a concern. However, we utilize a uniform sampling strategy to save the parameters of several teachers, which does not take up a lot of memory. For instance, the teacher network consists of 75.8 million parameters, which takes approximately 300 MB of memory. From N (1400 in our experiment) training iterations, we only save the parameters of D (e.g., 7) teachers with $n = N/D$ (e.g., 200) iteration gap between two consecutive teachers. This approach necessitates approximately $(N/n \times 300)$ (e.g., 2.1GB) of memory. We perform the analysis of memory consumption across various numbers of iteration gaps in Table V.

On the other hand, in our PKD-DLT, both the teacher and student networks should be trained, which brings extra computational costs compared to directly training the student network. On average, the training time for PKD-DLT is about $2\times$ to train a student network without KD. It takes approximately 60 minutes to train our network on THUMOS14 with a single Tesla V100 GPU. We only use the student network during the inference for OAD. Thus, our approach has the same inference complexity as the student network without KD.

Effects of different levels of teachers: We perform experiments to examine how different levels of teachers affect the student within training chunks, as shown in Table VI. To teach a student network at each training chunk d, we consider five different groups of teachers:

- $\mathcal{H}(\mathbf{T}_1, \ldots, \mathbf{T}_{d-1})$: the fusion of old teachers
- $\mathbf{T}_{d-1}$: the old teacher only from the previous chunk
- $\mathbf{T}_d$: the current teacher
- $\mathbf{T}_{d+1}$: the future teacher only from the next chunk, and
- $\mathcal{H}(\mathbf{T}_{d+1}, \ldots, \mathbf{T}_D)$: the fusion of future teachers

The top set in Table VI summarizes the results where the student network distills knowledge from a single category of different teacher levels. We find that the student distills better knowledge from the corresponding current teacher due to the relatively smaller knowledge gap between them. For example, as analogous to K-12 education, normally, a student in a middle school learns the best from a teacher well-trained for middle school education, compared to other levels of teachers. On the other hand, the bottom set in Table VI summarizes the results in which the student network distills the knowledge from the combination of different levels of teachers. In this scenario, we find that the knowledge from the current teacher and the

TABLE VI
EFFECTS OF DIFFERENT LEVELS OF TEACHERS

KD from					OadTR [20] + PKD-DLT		Colar [47] + PKD-DLT	
$\mathcal{H}(\mathbf{T}_1, ..., \mathbf{T}_{d-1})$	$\mathbf{T}_{d-1}$	$\mathbf{T}_d$	$\mathbf{T}_{d+1}$	$\mathcal{H}(\mathbf{T}_{d+1}, ..., \mathbf{T}_D)$	THUMOS14	TVSeries	THUMOS14	TVSeries
✓					60.5	86.3	61.4	86.7
	✓				60.4	86.2	61.2	86.6
		✓			**60.7**	**86.4**	**61.6**	**86.8**
			✓		60.1	86.0	61.0	86.4
				✓	60.0	85.9	60.9	86.3
✓		✓			**61.2**	**86.8**	**62.1**	**87.1**
	✓	✓			60.9	86.6	61.8	86.9
		✓	✓		60.5	86.3	61.3	86.7
		✓		✓	60.2	86.1	61.1	86.5

TABLE VII
ABLATION STUDY ON TEACHERS WITH DIFFERENT FUTURE OBSERVATIONS

Approach	OadTR [20] + PKD-DLT		Colar [47] + PKD-DLT	
	THUMOS14	TVSeries	THUMOS14	TVSeries
$\mathbf{T}$(+32 frames) $\rightarrow$ $\mathbf{S}$	60.1	86.2	61.1	86.7
$\mathbf{T}$(+64 frames) $\rightarrow$ $\mathbf{S}$	**61.2**	**86.8**	**62.1**	**87.1**
$\mathbf{T}$(+96 frames) $\rightarrow$ $\mathbf{S}$	60.9	86.6	61.7	87.0
$\mathbf{T}$(+128 frames) $\rightarrow$ $\mathbf{S}$	60.4	86.3	61.3	86.8

T: teacher network; and S: student network.

TABLE VIII
EFFECTIVENESS OF OUR PKD-DLT OVER VANILLA KD

Approach	Feature	OadTR [20]		Colar [47]	
		THUMOS14	TVSeries	THUMOS14	TVSeries
$\mathbf{S}$	TSN-Anet	58.3	85.4	59.4	86.0
$\mathbf{T} \rightarrow \mathbf{S}$ (vanilla KD)		59.9	86.1	60.9	86.6
$\mathbf{T} \rightarrow \mathbf{S}$ (PKD-DLT)		61.2	86.8	62.1	87.1
$\mathbf{S}$	TSN-Kinetics	65.2	87.2	66.9	88.1
$\mathbf{T} \rightarrow \mathbf{S}$ (vanilla KD)		66.4	87.7	68.0	88.6
$\mathbf{T} \rightarrow \mathbf{S}$ (PKD-DLT)		67.8	88.4	69.1	89.1

T: teacher network; and S: student network.

historical knowledge from old teachers lead the student to achieve better performance (e.g., a student in a middle school will learn from the teacher in middle school, in the meanwhile, the student should not forget the knowledge learned from teachers in elementary schools). By obtaining historical knowledge through the fusion of old teachers' logits, we produce a more smoothed label for regularization, which improves performances.

Ablation study on teachers with different observations: We further perform experiments to examine how teacher networks with different future observations affect the student network, as shown in Table VII. As the teacher network looks at more frames from the future, it can successfully detect action by capturing long-range temporal structures with sufficient observations of that action. However, when the observation of more frames from the future exceeds a certain number, distractive observations may occur. We achieve the best performance from the teacher that observes 64 more frames in the future.

Effectiveness of PKD-DLT over vanilla KD: As shown in Table VIII, we conduct experiments to justify that using a single high-level teacher to guide the student network throughout all training iterations (i.e., vanilla KD) may result in poor distillation compared to our PKD-DLT. Please note that the vanilla KD in Table VIII is not from previous works. It is also formulated from our knowledge distillation framework. For our vanilla KD, we only distill the knowledge from the high-level teacher network, while we progressively distill the knowledge from low- to high-level teacher networks for PKD-DLT. The results in Table VIII support this claim across various scenarios:

- *Different Datasets:* We evaluate performances on the THUMOS14 and TVSeries datasets to show the superior performances of our PKD-DLT over vanilla KD across diverse datasets.
- *Different Models:* To compare the effectiveness of our PKD-DLT over vanilla KD, we plug vanilla KD and PKD-DLT into different online action detection methods, OadTR [20] and Colar [47].
- *Different Features:* We use TSN-Anet and TSN-Kinetics features to show the effectiveness of our PKD-DLT over vanilla KD across diverse features.

These comprehensive experiments demonstrate that the vanilla KD, i.e., using a single high-level teacher to guide the student network throughout all training iterations, results in poor distillation compared to our PKD-DLT.

E. Qualitative Results

As shown in Fig. 3, we visualize some online action detection results, where we visualize the detection results for three settings: (i) detection results for the student network (i.e., OadTR [20]) without knowledge distillation (KD); (ii) detection results for the student network with our vanilla KD; and (iii) detection results for the student network with our PKD-DLT. As shown in Fig. 3, the student network without KD roughly detects actions with some false positives and false negatives. Since the student network without KD learns to detect ongoing actions from insufficient observations (i.e., observations of the past and current frames), it incorrectly treats some background frames as action and some action frames as background, which results in false positive and false negative errors. In contrast, the student network with our vanilla KD and the student network with our PKD-DLT learn to detect ongoing actions by distilling knowledge from the teacher network that looks at more frames (i.e., the past, current, and future frames) for sufficient observations. Therefore, as shown in Fig. 3, the student networks with our vanilla KD and our PKD-DLT improve the detection performance compared to the student network without KD by detecting more action instances and suppressing more background-related frames. On the other hand, our PKD-DLT outputs more precise detection than the student network with our vanilla KD, which

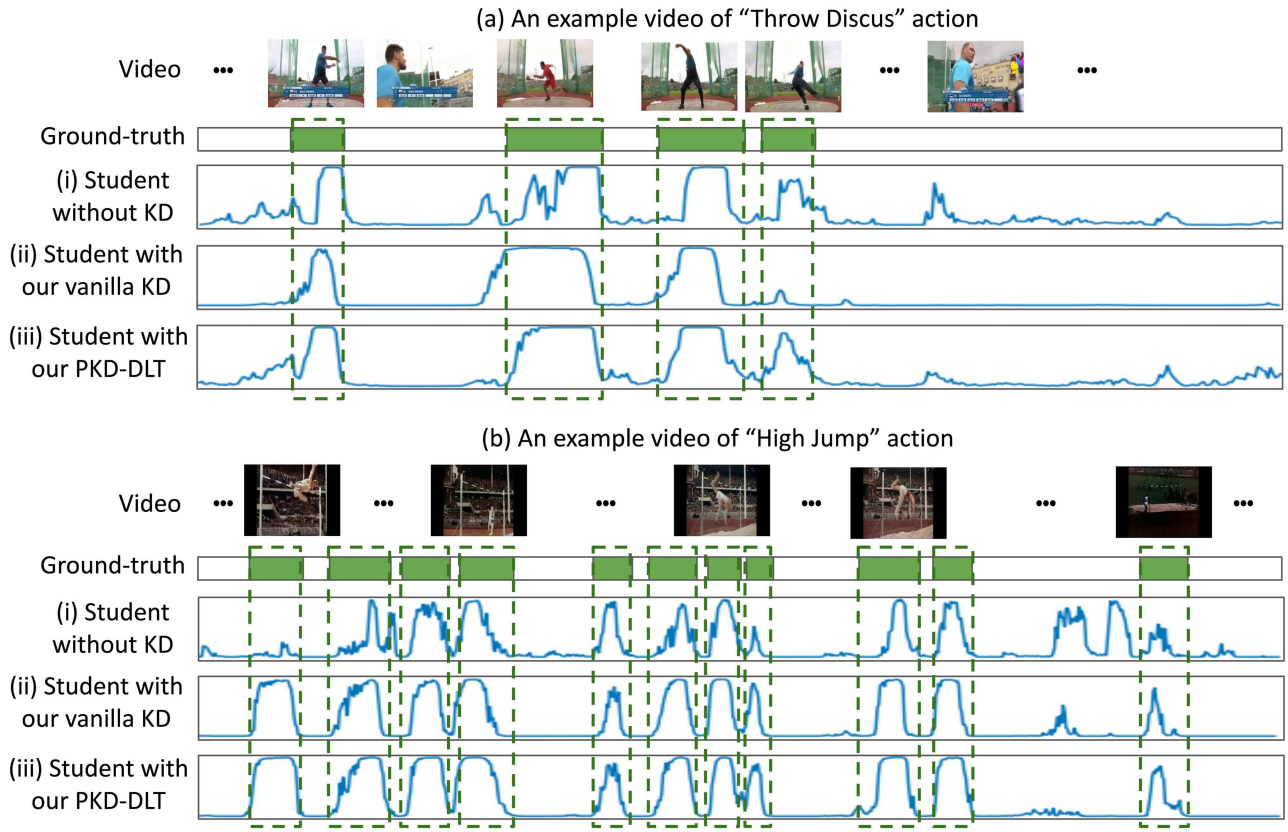


Fig. 3. Qualitative results. With many background frames, the video (a) and (b) contain multiple instances of "Throw Discus" and "High Jump" actions, respectively. We visualize the detection results for three settings: (i) detection results for the student network without knowledge distillation (KD); (ii) detection results for the student network with our vanilla KD; and (iii) detection results for the student network with our PKD-DLT.

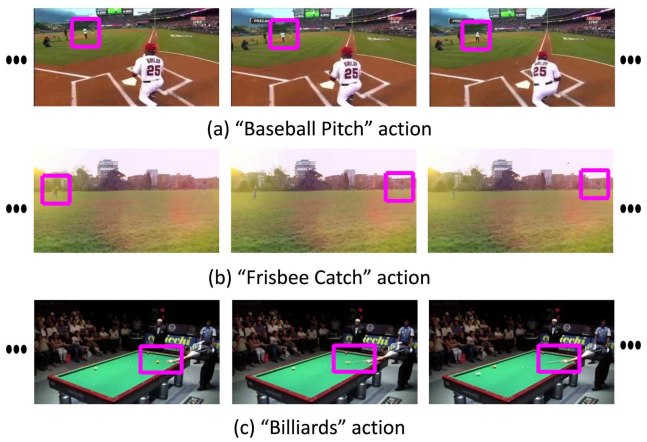


Fig. 4. Failure cases. (a), (b), and (c) contain sample frames of "Baseball Pitch", "Frisbee Catch" and "Billiards" actions, respectively. Magenta boxes indicate where the action of interest is being performed.

validates the effectiveness of knowledge distillation from different levels of teachers compared to the distillation only from a high-level teacher.

F. Failure Analysis and Future Work

Although we achieve state-of-the-art performances, the online action detection performance is not 100% correct yet. In Fig. 4, we visualize some sample frames for which we get incorrect detection. Most of the failure cases are from: (1) **Barely visible action.** The action in which the subject (i.e., the person who is performing the corresponding action) is very far away from the camera (e.g., Fig. 4(a) and (b)); and (2) **Tiny motion.** The action incurs only tiny movements, as shown in Fig. 4(c).

In this work, we plug our PKD-DLT into the recent online action detection models: OadTR [20] and Colar [47], which are capable of capturing long-range temporal structures. Motivated by the failure cases, in the future, we will first develop an online action detection model to capture long-range structures in both spatial and temporal domains, and then plug our PKD-DLT into that model.

On the other hand, recent advancements with large foundation models like GPT-4 V have demonstrated impressive capabilities in visual tasks. The potential applications of employing these models in online action detection tasks include: (i) Feature Extraction: Pretrained large foundation models can provide rich, high-dimensional feature representations essential for online action detection; (ii) Multimodal Integration: Large foundation models can integrate visual information with other data sources, such as audio, to enhance the performance of online action detection; and (iii) Pretrained large foundation models can be fine-tuned on task-specific datasets to improve performance of online action detection. However, achieving real-time processing of video frames is crucial for online action

detection. By progressively distilling knowledge from large models into smaller ones, we can develop efficient student models that maintain high performance while demanding fewer computational resources.

V. Conclusion

We propose a novel progressive knowledge distillation from different levels of teachers (PKD-DLT) for online action detection. Rather than using a single high-level teacher network to guide a student network throughout all training iterations, we also generate several low- and middle-level teachers, and progressively transfer the knowledge from different levels of teachers to the student network throughout training iterations. Experimental results on two challenging datasets demonstrate that our PKD-DLT is an effective plug-in to improve the performances of previous online action detection methods and achieve state-of-the-art.

References

[1] 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.

[2] 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. 5734–5743.

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

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

[5] F. Long et al., "Gaussian temporal awareness networks for action localization," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 344–353.

[6] Z. Zhu, W. Tang, L. Wang, N. Zheng, and G. Hua, "Enriching local and global contexts for temporal action localization," in *Proc. Comput. Vis. Pattern Recognit.*, 2021, pp. 13516–13525.

[7] 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.

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

[9] D. Liu, T. Jiang, and Y. Wang, "Completeness modeling and context separation for weakly supervised temporal action localization," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 1298–1307.

[10] M. Moniruzzaman, Z. Yin, Z. He, R. Qin, and M. C. Leu, "Action completeness modeling with background aware networks for weakly-supervised temporal action localization," in *Proc. ACM Int. Conf. Multimedia*, 2020, pp. 2166–2174.

[11] L. Huang, L. Wang, and H. Li, "Foreground-action consistency network for weakly supervised temporal action localization," in *Proc. Int. Conf. Comput. Vis.*, 2021, pp. 8002–8011.

[12] L. Huang, L. Wang, and H. Li, "Weakly supervised temporal action localization via representative snippet knowledge propagation," in *Proc. Comput. Vis. Pattern Recognit.*, 2022, pp. 3272–3281.

[13] W. Yang et al., "Uncertainty guided collaborative training for weakly supervised temporal action detection," in *Proc. Comput. Vis. Pattern Recognit.*, 2021, pp. 53–63.

[14] M. Moniruzzaman and Z. Yin, "Feature weakening, contextualization, and discrimination for weakly supervised temporal action localization," *IEEE Trans. Multimedia*, vol. 26, pp. 270–283, 2024.

[15] H. Song et al., "Temporal action localization in untrimmed videos using action pattern trees," *IEEE Trans. Multimedia*, vol. 21, pp. 717–730, 2019.

[16] R. De Geest and T. Tuytelaars, "Modeling temporal structure with LSTM for online action detection," in *Proc. Winter Conf. Appl. Comput. Vis.*, 2018, pp. 1549–1557.

[17] H. Eun, J. Moon, J. Park, C. Jung, and C. Kim, "Learning to discriminate information for online action detection," in *Proc. Comput. Vis. Pattern Recognit.*, 2020, pp. 806–815.

[18] M. Xu, M. Gao, Y.-T. Chen, L. S. Davis, and D. J. Crandall, "Temporal recurrent networks for online action detection," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 5531–5540.

[19] M. Xu et al., "Long short-term transformer for online action detection," in *Proc. Conf. Neural Inf. Process. Syst.*, 2021, pp. 1086–1099.

[20] X. Wang et al., "OADTR: Online action detection with transformers," in *Proc. Int. Conf. Comput. Vis.*, 2021, pp. 7565–7575.

[21] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 3967–3976.

[22] Y. Yang, J. Qiu, M. Song, D. Tao, and X. Wang, "Distilling knowledge from graph convolutional networks," in *Proc. Comput. Vis. Pattern Recognit.*, 2020, pp. 7074–7083.

[23] Y. Aytar, C. Vondrick, and A. Torralba, "SoundNet: Learning sound representations from unlabeled video," in *Proc. Conf. Neural Inf. Process. Syst.*, 2016, pp. 892–900.

[24] W. Son, J. Na, J. Choi, and W. Hwang, "Densely guided knowledge distillation using multiple teacher assistants," in *Proc. Int. Conf. Comput. Vis.*, 2021, pp. 9395–9404.

[25] J. H. Cho and B. Hariharan, "On the efficacy of knowledge distillation," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 4794–4802.

[26] Y. Cui et al., "Uncertainty-guided semi-supervised few-shot class-incremental learning with knowledge distillation," *IEEE Trans. Multimedia*, vol. 25, pp. 6422–6435, 2022.

[27] W. Tan, L. Zhu, J. Li, H. Zhang, and J. Han, "Teacher-student learning: Efficient hierarchical message aggregation hashing for cross-modal retrieval," *IEEE Trans. Multimedia*, vol. 25, pp. 4520–4532, 2023.

[28] C. Chen et al., "Temporal self-ensembling teacher for semi-supervised object detection," *IEEE Trans. Multimedia*, vol. 24, pp. 3679–3692, 2022.

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

[30] L. Xu, X. Wang, W. Liu, and B. Feng, "Cascaded boundary network for high-quality temporal action proposal generation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3702–3713, Oct. 2019.

[31] P. Chen et al., "Relation attention for temporal action localization," *IEEE Trans. Multimedia*, vol. 22, pp. 2723–2733, 2020.

[32] Y. Shi, Y. Tian, Y. Wang, and T. Huang, "Sequential deep trajectory descriptor for action recognition with three-stream CNN," *IEEE Trans. Multimedia*, vol. 19, pp. 1510–1520, 2017.

[33] R. R. A. Pramono, Y.-T. Chen, and W.-H. Fang, "Spatial-temporal action localization with hierarchical self-attention," *IEEE Trans. Multimedia*, vol. 24, pp. 625–639, 2022.

[34] D. Guo, W. Li, and X. Fang, "Fully convolutional network for multiscale temporal action proposals," *IEEE Trans. Multimedia*, vol. 24, pp. 625–639, 2018.

[35] F. Long et al., "Coarse-to-fine localization of temporal action proposals," *IEEE Trans. Multimedia*, vol. 22, pp. 1577–1590, 2020.

[36] K. Xia et al., "Exploring action centers for temporal action localization," *IEEE Trans. Multimedia*, vol. 25, pp. 9425–9436, 2023.

[37] 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. 6402–6411.

[38] M. Moniruzzaman, Z. Yin, Z. He, R. Qin, and M. C. Leu, "Human action recognition by discriminative feature pooling and video segment attention model," *IEEE Trans. Multimedia*, vol. 24, pp. 689–701, 2022.

[39] P. Lee, J. Wang, Y. Lu, and H. Byun, "Weakly-supervised temporal action localization by uncertainty modeling," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 1854–1862.

[40] P. Lee, Y. Uh, and H. Byun, "Background suppression network for weakly-supervised temporal action localization," in *Proc. Assoc. Advance. Artif. Intell.*, 2020, pp. 11320–11327.

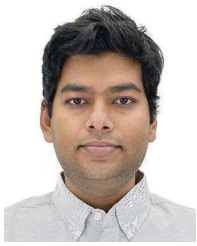
[41] Y. Zhai et al., "Action coherence network for weakly-supervised temporal action localization," *IEEE Trans. Multimedia*, vol. 24, pp. 1857–1870, 2022.

[42] S. Zhang, L. Song, C. Gao, and N. Sang, "GLNet: Global local network for weakly supervised action localization," *IEEE Trans. Multimedia*, vol. 22, pp. 2610–2622, 2020.

[43] P. Zhao, L. Xie, J. Wang, Y. Zhang, and Q. Tian, "Progressive privileged knowledge distillation for online action detection," *Pattern Recognit.*, vol. 129, 2022, Art. no. 108741.

[44] R. De Geest et al., "Online action detection," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 269–284.

[45] S. Qu et al., "LAP-Net: Adaptive features sampling via learning action progression for online action detection," 2020, *arXiv:2011.07915*.
[46] M. Gao, Y. Zhou, R. Xu, R. Socher, and C. Xiong, "WOAD: Weakly supervised online action detection in untrimmed videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 1915–1923.
[47] L. Yang, J. Han, and D. Zhang, "Colar: Effective and efficient online action detection by consulting exemplars," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 3160–3169.
[48] J. Chen, G. Mittal, Y. Yu, Y. Kong, and M. Chen, "Gatehub: Gated history unit with background suppression for online action detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 19925–19934.
[49] J. Wang, G. Chen, Y. Huang, L. Wang, and T. Lu, "Memory-and-anticipation transformer for online action understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 13824–13835.
[50] M. Guermal, A. Ali, R. Dai, and F. Brémond, "JOADAA: Joint online action detection and action anticipation," in *Proc. Winter Conf. Appl. Comput. Vis.*, 2024, pp. 6889–6898.
[51] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.
[52] Y. Li et al., "Learning from noisy labels with distillation," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 1910–1918.
[53] L. Yuan, F. E. Tay, G. Li, T. Wang, and J. Feng, "Revisiting knowledge distillation via label smoothing regularization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3903–3911.
[54] 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. 10687–10698.
[55] G. Xu, Z. Liu, X. Li, and C. C. Loy, "Knowledge distillation meets self-supervision," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 588–604.
[56] N. Passalis and A. Tefas, "Learning deep representations with probabilistic knowledge transfer," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 588–604.
[57] G. Aguilar et al., "Knowledge distillation from internal representations," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 7350–7357.
[58] I. Chung, S. Park, J. Kim, and N. Kwak, "Feature-map-level online adversarial knowledge distillation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 2006–2015.
[59] J. Yuan, M. H. Phan, L. Liu, and Y. Liu, "FAKD: Feature augmented knowledge distillation for semantic segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2024, pp. 595–605.
[60] Z. Luo, J.-T. Hsieh, L. Jiang, J. C. Niebles, and L. Fei-Fei, "Graph distillation for action detection with privileged modalities," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 166–183.
[61] Y. Tian, D. Krishnan, and P. Isola, "Contrastive representation distillation," 2019, *arXiv:1910.10699*.
[62] W. Chen et al., "Feature estimations based correlation distillation for incremental image retrieval," *IEEE Trans. Multimedia*, vol. 24, pp. 1844–1856, 2022.
[63] H.-J. Ye, S. Lu, and D.-C. Zhan, "Distilling cross-task knowledge via relationship matching," in *Proc. Comput. Vis. Pattern Recognit.*, 2020, pp. 12396–12405.
[64] N. C. Garcia, P. Morerio, and V. Murino, "Modality distillation with multiple stream networks for action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 103–118.
[65] G. Radevski, D. Grujicic, M. Blaschko, M.-F. Moens, and T. Tuytelaars, "Multimodal distillation for egocentric action recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 5213–5224.
[66] S. I. Mirzadeh et al., "Improved knowledge distillation via teacher assistant," in *Proc. Assoc. Advance. Artif. Intell.*, 2020, pp. 5191–5198.
[67] S. You, C. Xu, C. Xu, and D. Tao, "Learning from multiple teacher networks," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2017, pp. 1285–1294.
[68] S. Park and N. Kwak, "Feature-level ensemble knowledge distillation for aggregating knowledge from multiple networks," in *Proc. Eur. Conf. Artif. Intell.*, 2020, pp. 1411–1418.
[69] Z. Hao, Y. Luo, Z. Wang, H. Hu, and J. An, "CDFKD-MFS: Collaborative data-free knowledge distillation via multi-level feature sharing," *IEEE Trans. Multimedia*, vol. 24, pp. 4262–4274, 2022.
[70] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," 2022, *arXiv:2202.00512*.
[71] W. Shi, Y. Song, H. Zhou, B. Li, and L. Li, "Follow your path: A progressive method for knowledge distillation," in *Proc. Eur. Conf. Mach. Learn. Princ. Pract. Knowl. Discov. Databases*, 2021, pp. 596–611.
[72] K. Kim, B. Ji, D. Yoon, and S. Hwang, "Self-knowledge distillation with progressive refinement of targets," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 6567–6576.
[73] Y. Xie, H. Zhang, X. Xu, J. Zhu, and S. He, "Towards a smaller student: Capacity dynamic distillation for efficient image retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 16006–16015.
[74] Y.-G. Jiang et al., "THUMOS challenge: Action recognition with a large number of classes," 2014.
[75] C. Zhang, M. Cao, D. Yang, J. Chen, and Y. Zou, "COLA: Weakly-supervised temporal action localization with snippet contrastive learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 16010–16019.
[76] J. Gao, Z. Yang, and R. Nevatia, "RED: Reinforced encoder-decoder networks for action anticipation," in *Proc. Brit. Mach. Vis. Conf.*, 2017.
[77] L. Wang et al., "Temporal segment networks: Towards good practices for deep action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 20–36.
[78] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[79] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[80] F. Caba 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.
[81] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE Comput. Vis. Pattern Recognit.*, 2017, pp. 6299–6308.
[82] J. Donahue et al., "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Comput. Vis. Pattern Recognit.*, 2015, pp. 2625–2634.



Md Moniruzzaman (Graduate Student Member, IEEE) received the B.S. degree from the Department of Electronics and Communication Engineering, Khulna University of Engineering and Technology, Khulna, Bangladesh. He is currently working toward the Ph.D. degree with the Department of Computer Science, Stony Brook UniversityStony Brook, NY, USA. His research interests include human action anticipation, human action recognition, temporal action localization, and human pose estimation.

Zhaozheng Yin (Senior Member, IEEE) is currently a SUNY Empire Innovation Associate Professor with Stony Brook University, Stony Brook, NY, USA. He is also affiliated with the AI Institute, Department of Biomedical Informatics, and Department of Computer Science. His group has been working on biomedical image analysis, computer vision, and machine learning. He was Area Chair for CVPR, ECCV, MICCAI and WACV, and an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and *Journal of Visual Communication and Image Representation*.