Neural Processing Letters (2024) 56:145
https://doi.org/10.1007/s11063-024-11598-w

Learning Reliable Dense Pseudo-Labels for Point-Level Weakly-Supervised Action Localization

Yuanjie Dang[1] · Guozhu Zheng[1] · Peng Chen[1] · Nan Gao[1] · Ruohong Huan[1] · Dongdong Zhao[1] · Ronghua Liang[1]

Accepted: 17 March 2024

Abstract
Point-level weakly-supervised temporal action localization aims to accurately recognize and localize action segments in untrimmed videos, using only point-level annotations during training. Current methods primarily focus on mining sparse pseudo-labels and generating dense pseudo-labels. However, due to the sparsity of point-level labels and the impact of scene information on action representations, the reliability of dense pseudo-label methods still remains an issue. In this paper, we propose a point-level weakly-supervised temporal action localization method based on local representation enhancement and global temporal optimization. This method comprises two modules that enhance the representation capacity of action features and improve the reliability of class activation sequence classification, thereby enhancing the reliability of dense pseudo-labels and strengthening the model's capability for completeness learning. Specifically, we first generate representative features of actions using pseudo-label feature and calculate weights based on the feature similarity between representative features of actions and segments features to adjust class activation sequence. Additionally, we maintain the fixed-length queues for annotated segments and design a action contrastive learning framework between videos. The experimental results demonstrate that our modules indeed enhance the model's capability for comprehensive learning, particularly achieving state-of-the-art results at high IoU thresholds.

Keywords Action localization · Point-level weak supervision · Contrastive learning

1 Introduction

The goal of point-level weakly-supervised temporal action localization(P-WSTAL) is to identify action instances within the temporal dimension of unprocessed videos using point-level annotations. Point-level annotations refer to the practice of annotating only a single timestamp and action class for each action instance during training. In comparison to traditional

✉ Yuanjie Dang
dangyj@zjut.edu.cn

[1] The College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China

Published online: 10 April 2024

weakly supervised temporal action localization methods that solely rely on video-level labels, the additional cost is limited, yet the performance improvement is significant. Moreover, a substantial body of empirical evidence suggests that point-level annotation information can effectively reduce the performance gap between weakly supervised and fully supervised approaches. [17, 21]

While point-level supervised methods integrate more action information into temporal action localization, the sparse nature of point-level labels can still lead to issues such as incomplete action instance detection and inaccurate action boundaries. To address this challenge, some researchers proposed sparse pseudo-labels based method and dense pseudo-labels based method. For example, SF-Net [21] mines sparse pseudo-labels through segment prediction scores and score thresholds, while LACP [17] generates dense pseudo-labels through point annotations and segment prediction scores. Furthermore, for instance, CRRC [9] proposes generating correction weights to enhance the reliability of sparse pseudo-labels. However, these existing methods primarily focus on enhancing action representations within individual videos, aiming to improve representation compactness, without effectively separating the influence of scene information on action representation enhancement. Additionally, the correction weights in these methods is typically targeted at segments with prediction scores exceeding the average. This approach overlooks the potential for global optimization of correction weights across the temporal domain. Considering the sparsity of point annotations, this limitation persists, and the issue of ensuring the reliability of dense pseudo-labels remains inadequately addressed. Due to these aforementioned challenges, the model's ability to learn completeness still falls short.

To address the aforementioned challenges, we propose a point-level weakly-supervised temporal action localization method based on local representation enhancement and global temporal optimization called LRDP-Net. On the one hand, we leverage point-level label based on local feature to enhance action representations, reducing the impact of noisy labels on learning video content. On the other hand, we utilize prior knowledge from point-label to perform global temporal optimization, thereby improving reliability of dense labels. And prior knowledge refers to the action prototype features from point-label annotated video segments, along with the count of actions within the video. These action prototype features are then leveraged to compute weights to optimize classification results. The count of actions is manifested in point-level classification loss, enabling the model to assimilate details regarding the quantity of action instances in the video. Specifically, our framework comprises two modules: a Prior Knowledge-based Classifier Optimization(PKCO) Module and a Scene-agnostic Action Representation Learning(SARL) module. Firstly, our approach introduces scene-agnostic action segment information by leveraging point-level label from different videos and performing contrastive learning between local feature of point-level label across videos. We maintain multiple fixed-length queues to save the features of point-level label from different videos and dynamically update the queues during the comparative learning process. In addition, considering point-level labels tend to be action key frames, we choose to use point-level label features to initialize action representative features. Therefore, this further enhances the compactness of action features and the robustness of action representative features. Afterwards, based on these robust action representation features which are treated as prior knowledge, we calculate the feature similarity between segments and different action classes, obtaining weight values for all action classes, which are integrated to form the weight of segment-action class probabilities. These weights are then used to refine the class activation sequences, aligning them more accurately with the true segment categories. Through this global temporal optimization operation we further enhance the performance of the classifier and thereby improve the reliability of dense pseudo-labels.

We summarize our main contributions as follows:(a)We propose a Scene-agnostic Action Representation Learning module to improve the robustness of action representative features by introducing prior knowledge of the same class of actions across different videos. (b) On the basis of robust action representative features, we propose a Prior Knowledge-based Classifier Optimization Module to generate corrective weights by measuring segment and action representative features to improve the reliability of dense labels. (c)We validate the effectiveness of our approach on THUMOS'14, GTEA, and BEOID datasets, achieving state-of-the-art performance at high thresholds.

2 Related Works

Point-level Weakly-supervised Temporal Action Localization. Different from the traditional WSTAL method [1, 7, 19], P-WSTAL provides single-frame labels for each action instance. Moltisanti et al. [23] first introduced point annotations to the field of action localization. SF-Net [21] then proposed a point-based mining algorithm to obtain sparse pseudo-labels, partially alleviating the issue of limited annotation information. DCM [14] chose to train a keyframe detector using annotated points and divide the entire video into multiple segments using these key points, transforming the task from action localization in a complete video to localization within several segments. LACP [17] introduced a strategy for action completeness learning based on dense labels, with a focus on action integrity. After obtaining dense labels, LACP primarily performed foreground-background separation operations. CRRC [9], on the other hand, argued that previous pseudo-label mining methods overlooked the relationships in the feature space and proposed a novel metric pseudo-label mining algorithm. In some other methods aimed at enhancing the reliability of pseudo-labels, Wang et al. [29]proposed SCAM, which integrates class activation mappings from multiple information sources to enhance the accuracy of pseudo-labels. Chen et al. [3, 4]introduced LSTI and STVS, where the former enhances pseudo-labels reliability by incorporating cross-domain information, $i.e.$, inter-frame information, based on intra-frame information, while the latter achieves significant spatial and temporal interaction through a novel temporal unit, enabling the network to better perceive temporal information.

Contrastive Representation Learning. Contrastive representation learning primarily involves learning an embedding space based on internal patterns in the data, where related features are pulled together while unrelated features are pushed apart through a noise-contrastive mechanism. In addition, there are also graph contrastive learning [31, 36]. CMC [28] introduced a contrastive learning framework that maximizes the mutual information between different views of the same scene. SimCLR [5] selects negative samples by using augmented views of other items within a mini-batch. MoCo [11], on the other hand, introduces a memory bank to dynamically update negative samples, overcoming the limitation of batch size and achieving negative sample consistency.

3 Proposed Method

In this section, we will provide an overview of the P-WSTAL problem definition and detail the baseline setup. We will then delve into the specifics of the PKCO and SARL modules, explaining their implementation in detail. Finally, we will discuss the optimization and inference processes of our model. Figure 1 illustrates the overall architecture of our framework.

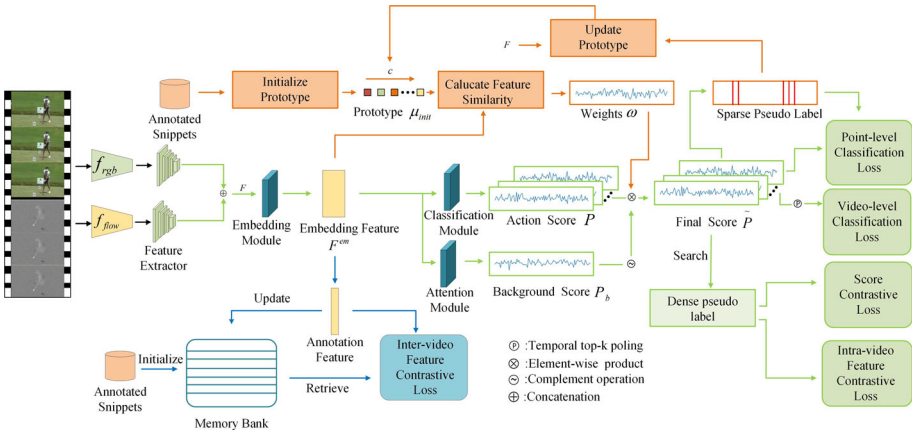

Fig. 1 Overview of the LRDP-Net framework. The green part is our baseline. The orange part is PKCO Module, which computes the similarity between representative features of actions and snippet features to obtain correction weights, thus optimizing the CAS sequence. The blue part is SARL Module, which utilizes a memory bank initialized with annotated snippets to learn feature comparisons across videos, enhancing the model's representational capacity

3.1 Problem Formulation and Baseline Setup

Problem Formulation Following the previous works [9, 17, 21], we set up the problem of point-level weakly-supervised temporal action localization: During the training process, each training video can contain multiple instances of actions, as well as their corresponding categories. Additionally, annotations and classifications are provided for individual frames within each action instance. Concretely, given an arbitrary video V, its point-level annotations is $\mathcal{A}^{point} = \left\{\left(t_i, y_{t_i}\right)\right\}_{i=1}^{M_{act}}$, where i-th action instance is labeled at the t_i-th segment with its action label y_{t_i}, and M^{act} is the number of action instances in the input video. The $y_{t_i} \in \mathbb{R}^C$ is a binary vector and $y_{t_i}[k] = 1$ if the i-th action instance contains the k-th action class, where C is the total number of action classes. But the video level label $y^{vid} \in \mathbb{R}^C$ is multi-hot vector, which could be readily acquired by aggregating the point-level label y_{t_i} among the temporal dimension.

Baseline Setup As illustrated in Fig. 1, our model architecture involves dividing the input video into segments, each consisting of 16 frames. These segments would be fed into a pre-trained feature extractor to generate RGB and flow features. The fused feature $F \in \mathbb{R}^{D \times T}$ is obtained by concatenating the RGB and flow features, where D represents the dimension of the input features and T denotes the total number of segments in the input video. The fused feature F is then fed into a one-dimensional convolutional layer to obtain task-relevant enhanced features $F^{em} \in \mathbb{R}^{D \times T}$. These features are subsequently input into a classification module to generate segment-level class scores $P \in \mathbb{R}^{T \times C}$. In addition, following the approach of other methods [17, 21], we introduce a attention module to discriminate between foreground and background segments, resulting in background scores $P_b \in \mathbb{R}^T$. After obtaining both classification scores, we fuse them to generate the final score $\widetilde{P}$, *i.e.*, $\widetilde{p}[c] = p[c]\,(1 - P_b)$. The video-level prediction score $\hat{p}^{vid}$ can be obtained by temporal top-k pooling, and following the common pipeline of state-of-the-art methods [9, 17], $k = [\frac{T}{8}]$. After obtaining the video-level prediction results $\hat{p}^{vid}$, we calculate the video-level classification loss L_{video} through a standard binary cross-entropy loss.

Following previous methods [17], loss L_{score} and L_{feat} were computed using the obtained dense pseudo-labels. The generation method for dense labels employs the optimal sequence search algorithm and utilizes the outer-inner-contrast concept [27] to determine the completeness score of a sequence. Specifically, a substantial contrast between inner and outer scores is expected for a complete action instance, whereas it is anticipated to be minimal for a fragmented one. Furthermore, sequence generation proceeds through a search algorithm based on greedy principles.

3.2 Prior Knowledge-Based Classifier Optimization Module

To mitigate the problem of dense pseudo-label noise, we propose an effective prior knowledge-based classifier optimization module that leverages the relative distances between representative features of actions and snippet features to determine the weights of action class probabilities and optimize the prediction scores. Taking inspiration from the concept of DCM [14], which suggests that annotated positions often correspond to action keyframes, we utilize annotated snippets at the beginning of training to obtain reliable representative features of actions. Specifically, the representative features for an action category c can be derived by computing the average of the features from all video snippets categorized as c across the entire dataset. as follows:

$$\mu_k^{init} = \frac{\sum_{F^{em}\in\mathcal{D}} \sum_{f_j^{em}\in F^{em}} f_j^{em} * \mathbb{1}\left(y_j[c] == 1\right)}{\sum_{F^{em}\in\mathcal{D}} \sum_{f_j^{em}\in F^{em}} \mathbb{1}\left(y_j[c] == 1\right)}, \tag{1}$$

where $\mathcal{D}$ denotes the all training video in datasets. f_j^{em} denotes the task-related feature F^{em} of j -th labeled segment, $y_j[c]$ denotes the ground truth for the annotated segment of class c on the j-th video segment. $\mathbb{1}$ is the indicator function, specifically representing that it returns 1 if the input result is True, and returns 0 if the input result is False..

Once the representative features of actions are initialized, we calculate the similarity between video snippet features and the representative features of actions using the Euclidean distance. The weight coefficients for different action class predictions are then determined based on the comparison of similarities.

$$\begin{aligned} \mathcal{S}_j &= \frac{\sum_{c=1}^{C} \exp\left(-\left\| F\left(f_j^{em}\right) - F\left(\mu_c\right)\right\|_2 / \tau\right)}{C}, \\ \omega_j[c] &= \frac{\exp\left(-\left\| F\left(f_j^{em}\right) - F(\mu_c)\right\|_2 / \tau\right)}{\mathcal{S}_j}, \end{aligned} \tag{2}$$

where j represents video snippets, c represents action classes, τ is a temperature parameter, $\mathcal{S}_j$ is The average similarity between the j-th segment and all representative features of actions, and $F(\cdot)$ is the L2 normalization function. The obtained weights are multiplied with the previously computed final scores $\widetilde{P}$ to optimize the sequence of class classifications and Acquiring updated prediction scores $\hat{P}$.

$$\hat{p}_j[c] = \begin{cases} 1, & \text{if } \widetilde{p}_j[c]\omega_j[c] >= 1 \\ \widetilde{p}_j[c]\omega_j[c], & otherwise \end{cases} \tag{3}$$

Where $\hat{p}_j[c]$ represents the optimized predicted score for class c on the j-th video segment, $\hat{p}_j[c]$ and denotes the predicted score for class c on the j-th video segment. Following the sequence optimization, we also conduct sparse pseudo label mining. As we expand the action labels, our representative features of actions are continuously refined, resulting in more accurate representative features of actions. Furthermore, the obtained pseudo action labels are utilized for point-level classification loss calculation. This approach not only facilitates dynamic updates of the representative features of actions but also enriches the label information.

3.3 Scene-Agnostic Action Representation Learning Module

In order to optimize prediction scores effectively, it is essential to construct precise representative features of actions and consider the substantial intra-action variation observed among different segments of the same action. This calls for a compact representation of the model's action features. To tackle this challenge, we introduce a scene-agnostic action representation learning module based on annotated segments. By learning representation contrasts among annotated segments across videos, we enhance the compactness of action representation and improve the robustness of representative features of actions. Scene-agnostic representation learning and scene-specific representation learning mutually reinforce each other, with scene-agnostic learning abstracting the characteristics of actions, thereby refining the robustness of representative features of actions.

Specifically, we start by randomly initializing the fixed-length queues with the same number of action classes, denoted as $S \in \mathbb{R}^{L\times C\times D}$, where L denotes the number of features stored per category in each queue, C denotes the total number of action classes, and D represents the dimensions of the features. Prior to model training, we traverse the dataset and perform random sampling of annotated data for each category in each video, replacing the corresponding values in the original queues S. This initialization process, combined with the inherent size limitation of sequences, ensures that the features in the resulting queues S are scene-agnostic during the initial training phase.

Next, we introduce a feature matching task for the video model to enhance its capability in representing instance features. We extract all annotated features $X \in \mathbb{R}^{N\times D}$ from the task-relevant enhanced features F^{em} and filter out features $S^p \in \mathbb{R}^{L\times D}$ corresponding to segments of the same category, as well as features $S^n \in \mathbb{R}^{L\times(C-1)\times D}$ corresponding to segments of different categories, based on the annotation information. Then the InfoNCE loss is applied:

$$\begin{aligned} \hat{\mathcal{S}}^p &= -\frac{1}{N}\sum_{i=1}^{L} S_i^p, \\ \mathcal{L}_{constract} &= -\frac{1}{N}\sum_{i=1}^{N} \log\left(\frac{\exp(\cos(x_i, \hat{\mathcal{S}}^p)/\tau)}{\sum_{l,c}\exp(\cos(x_i, S_{l,c}^n))}\right), \end{aligned} \tag{4}$$

where $l = 1 \sim L$, $c = 1 \sim C-1$, $\hat{\mathcal{S}}^p$ denotes the average of features within the queues S that correspond to the same category as the segment X_i.

In the loss function, only sequences that share the same category as the segment are considered as positive samples, while others are treated as negative samples. The purpose of using this InfoNCE loss is to guide the network in enhancing the representation of action instances. Additionally, the update of sequences follows the rule of first-in-first-out within a

queue. After calculating the loss, the new features X obtained during the process are updated into the sequence S, enabling dynamic updates of the sequence while ensuring that feature comparisons are conducted across videos.

3.4 Model Optimization and Inference

The overall training objective of our model is as follows.

$$\mathcal{L} = \mathcal{L}_{video} + \lambda_1 \mathcal{L}_{point} + \lambda_2 \mathcal{L}_{constract} + \lambda_3 \mathcal{L}_{score} + \lambda_4 \mathcal{L}_{feat} \tag{5}$$

where λ_* are weighting parameters for balancing the losses, which are determined empirically.

In the testing phase, similar to previous works, when an untrimmed video is inputted, we first utilize thresholds and $\hat{P}^{vid}$ to determine which actions are present in the video. Then, we use thresholds to assess the predicted scores $\hat{P}$ for each segment and select those above the threshold as candidate segments. Subsequently, we combine all candidate segments to generate localized action proposals. The confidence of each action proposal is set as its inner-outer contrast score. To improve the accuracy of the action proposals, we utilize the non-maximum suppression mechanism to remove overlapping proposals to obtain the localization results.

4 Experiment

4.1 Datasets

The THUMOS-14 [13] dataset consists of a total of 20 different action categories, covering a wide range of daily life activities and sports. It includes 200 and 213 untrimmed videos for validation and testing, respectively. The GTEA [18] dataset comprises 28 videos that contain 7 fine-grained categories of daily activities. Each action is performed by four different individuals. Following the setup in previous studies, we select 21 videos as the training set and 7 videos as the test set. The BEOID [6] dataset consists of 58 videos and includes 30 categories. On average, each video contains 12.5 action instances. In model training stage, we only utilize point-level annotations for training, and following mainstream methods [9, 17], out annotation information is sourced from the publicly available datasets of SF-Net.

4.2 Implementation Details

Following the current SOTA methods [8, 9], we adopt LACP as the baseline of our model. Specifically, we employ a pre-trained I3D network [2] on the Kinetics-400 [2] for extracting both RGB and flow features, which are represented as high-dimensional vectors of size 1024. The input videos are divided into segments consisting of 16 frames each. During the training phase, we utilize the Adam optimizer [16] with a learning rate set to 10^{-4}. The loss function weights in this model are set to:$\lambda_2 = 1.0, \lambda_1$ and other loss function weights are set in the same way as the baseline method. The length of memory bank S for each class-specific sequence is 45($i.e., L = 45$), and the temperature coefficient is 1.0($i.e., \tau = 1.0$). To ensure optimal convergence, we train the model for 2500 epochs in each training iteration. Following the standard protocol for temporal action localization, we calculate the mean average precisions

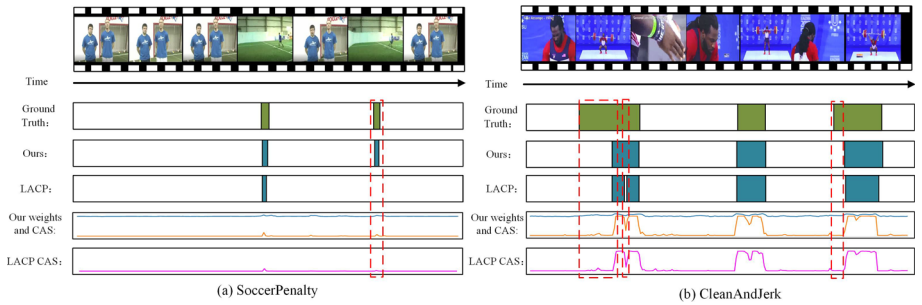

Fig. 2 Visualization of qualitative results on the THUMOS14 dataset. We provide two examples with different action classes: **a** *SoccerPenalty* and **2** *ClenAndJerk*. The green bar charts represent the ground truth, and the blue bar charts represent the localization results of the LACP and LRDP-Net, respectively. The blue curve represents the weights, the orange curve represents the CAS of LRDP-Net, and the magenta curve represents the CAS of LACP

(mAPs) across various intersection over union (IoU) thresholds to evaluate the performance of our action model.

4.3 Comparisons with the State-of-the-art

In Table 1, we compare several point-level weakly-supervised, weakly-supervised, and fully-supervised methods on the THUMOS'14 benchmark, which has been widely used in recent years. It is worth noting that fully-supervised methods require significantly higher annotation costs compared to weakly-supervised approaches. In the comparison of point-level weakly-supervised methods, our model outperforms the state-of-the-art approaches at high IoU thresholds. Specifically, we achieve a 0.3% lead in mAP@IoU(0.3−0.7) and an impressive 0.8% lead in mAP@IoU(0.7). However, our method slightly lags behind CRRC's performance in other metrics, particularly in terms of performance at lower thresholds. Considering the disparities in the problem-solving approaches between CRRC and our proposed method, we posit that enhancing the reliability of sparse pseudo labels contributes significantly to the model's prowess in action recognition, thus resulting in better performance at lower thresholds. On the other hand, improving the reliability of dense pseudo labels considerably enhances the model's ability to learn action completeness, thereby yielding superior performance at higher thresholds. When compared to traditional weakly-supervised methods, our approach consistently outperforms them, both at low and high IoU thresholds, highlighting the effectiveness of incorporating point-level annotations.

Furthermore, when comparing with fully-supervised methods, our approach performs competitively at low IoU thresholds but lags behind at high IoU thresholds. The main reason for this discrepancy is the lack of precise frame-level annotations, which affects the learning of action completeness. It is worth noting that our method achieves state-of-the-art performance at high IoU thresholds under weak supervision. After incorporating Prior Knowledge-based Classifier Optimization Module and Scene-agnostic Action Representation Learning Module, our model demonstrates improvements across various performance metrics, validating the effectiveness of the proposed modules.

We also conducted experiments on the BEOID and GTEA benchmarks, as shown in Table 2 and Table 3. Our method surpasses the current state-of-the-art approaches on both datasets, exhibiting significantly higher performance in terms of mAP@IoU(0.1−0.7).

Table 1 Comparisons of LRDP-Net with other methods on THUMOS14 dataset

Supervision	Method	mAP@IoU(%)							AVG mAP(%)		
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.1:0.5	0.3:0.7	0.1:0.7
Frame-level (TAL)	P-GCN [32](2019)	69.5	67.5	63.6	57.8	49.1	-	-	61.5	-	-
	G-TAD [30](2020)	66.1	64.2	54.5	47.6	40.2	30.8	23.4	54.5	39.3	46.7
	Zhao et al. [37](2020)	-	-	53.9	50.7	45.4	38.0	28.5	-	43.3	-
	GCM [33](2021)	72.5	70.9	66.5	60.8	51.9	-	-	64.5	-	-
	ActionFormer [35](2022)	-	-	82.1	77.8	71.0	59.4	43.9	-	66.8	-
	TriDet [26](2023)	-	-	83.6	80.1	72.9	62.4	47.4	-	69.3	-
Video-level (WSTAL)	DGAM [25](2020)	60.0	54.2	46.8	38.2	28.8	19.8	11.4	45.6	29.0	37.0
	CoLA [34](2021)	66.2	59.5	51.5	41.9	32.2	22.0	13.1	50.3	32.1	40.9
	RSKP [12](2022)	71.3	65.3	55.8	47.5	38.2	25.4	12.5	55.6	35.9	45.1
	ASM-Loc [10](2022)	71.2	65.5	57.1	46.8	36.6	25.2	13.4	55.4	35.8	45.1
	P-MIL [24](2023)	71.8	67.5	58.9	49.0	40.0	27.1	15.1	57.4	38.0	47.0
	PivoTAL [22](2023)	74.1	69.6	61.7	52.1	42.8	30.6	16.7	60.1	40.8	49.6
Point-level (P-WSTAL)	Moltisanti et al. [23](2019)	24.3	19.9	15.9	12.5	9.0	-	-	16.3	-	-
	SF-Net [21](2020)	68.3	62.3	52.8	42.4	30.5	20.6	12.0	51.2	31.6	41.2
	Ju et al. [15](2020)	72.3	64.7	58.2	47.1	35.9	23.0	12.8	55.6	35.4	44.9
	LACP [17](2021)	75.7	71.4	64.6	56.5	45.3	34.5	**21.8**	62.7	44.5	52.8
	DCM [14](2021)	70.2	63.5	55.6	44.7	32.3	22.0	12.3	53.3	33.4	42.9
	CRRC [9](2022)	**77.8**	**73.5**	**67.1**	**57.9**	**46.6**	33.7	19.8	**64.6**	45.1	**53.8**
	LRDP-Net(Ours)	76.7	72.6	66.2	57.8	46.3	**35.0**	21.5	63.9	**45.4**	53.7

Bold values indicate optimal performance in point-level supervision
We report three kinds of average mAP under different IoU thresholds for fair comparison, namely, 0.1:0.5, 0.3:0.7 and 0.1:0.7

Table 2 State-of-the-art comparison on GTEA

Dataset	Method	mAP@IoU(%)				
		0.1	0.3	0.5	0.7	AVG
GETA	SF-Net [21]	58.0	37.9	19.3	11.9	31.0
	Ju et al. [15]	59.7	38.3	21.9	18.1	33.7
	Li et al. [20]	60.2	44.7	28.8	12.2	36.4
	LACP [17]	63.9	55.7	33.9	20.8	43.5
	Ours	**65.2**	**57.3**	**35.2**	**20.9**	**44.7**

Bold values indicate optimal performance
AVG denotes the average mAP at the thresholds 0.1:0.1:0.7

Table 3 State-of-the-art comparison on BEOID

Dataset	Method	mAP@IoU(%)				
		0.1	0.3	0.5	0.7	AVG
BEOID	SF-Net [21]	62.9	40.6	16.7	3.5	30.9
	Ju et al. [15]	63.2	46.8	20.9	5.8	34.9
	Liet al. [20]	71.5	40.3	20.3	5.5	34.4
	LACP [17]	76.9	61.4	42.7	25.1	51.8
	Ours	**78.2**	**62.9**	**43.3**	**25.3**	**52.4**

Bold values indicate optimal performance
AVG denotes the average mAP at the thresholds 0.1:0.1:0.7

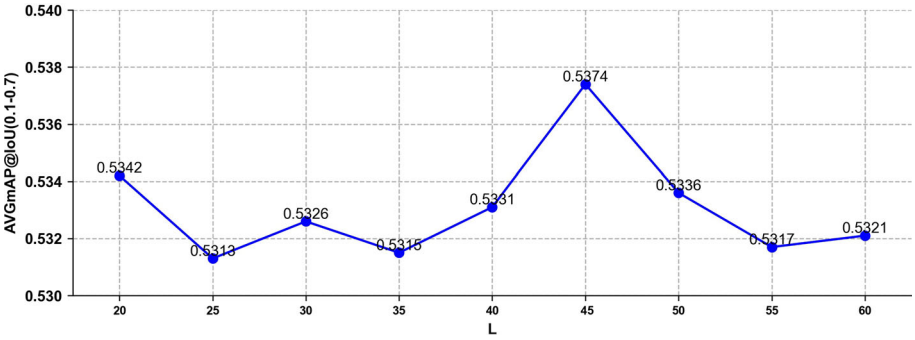

Fig. 3 Hyper-parameter analysis of L in Eq. (4). The X-axis represents the length of queues and the Y-axis denotes the average mAP IoU@(0.1:0.7)

4.4 Ablation Study

4.4.1 Effect of each Component

In Table 4, we perform an ablation study to investigate the contribution of each component. It can be noted that the average mAP is improved by 0.7% and 0.5% with the utilization of the PKCO and SARL modules, respectively. Furthermore, when the two modules are combined, the average mAP shows a significant improvement of 1.4%. This indicates the mutual promotion between modules, while also confirming the enhancement of action representative features accuracy by the SARL module. On another aspect, comparing the performance across different IoU thresholds reveals that the performance improvement is not as significant

Table 4 Ablation studies of LRDP-Net on the THUMOS-14 dataset

Exp	baseline	PKCO	SARL	mAP@0.1	mAP@0.2	mAP@0.3	mAP@0.4	mAP@0.5	mAP@0.6	mAP@0.7	AVG mAP(%)
1	✓			75.7	71.1	64.9	56.5	44.7	33.4	19.9	52.3
2	✓	✓		76.7	72.3	65.6	57.1	44.9	34.4	20.4	53.0
3	✓		✓	76.2	72.1	65.7	57.0	45.3	33.6	19.8	52.8
4	✓	✓	✓	**76.7**	**72.6**	**66.2**	**57.8**	**46.3**	**35.0**	**21.5**	**53.7**

Bold values indicate optimal performance
AVG mAP denotes the average mAP at the thresholds 0.1:0.1:0.7

Table 5 Model parameters and runtime speed analysis

Module	Parameters	Inference Time	FPS
LACP [17]	12.6	2.57ms	389.75
Ours	12.6M	2.64ms	378.47

at lower thresholds as it is at higher thresholds. This is due to the elevated impact of refined prediction scores on the model's capability for integrity learning, surpassing the effect of compact feature representation on the model's ability for action recognition, resulting in a superior performance of the model at higher thresholds.

4.4.2 Effect of Class Activation Sequence Optimization

To verify the corrective effect of the weights proposed by the PKCO module on prediction scores, we conducted visualizations of the weights and final prediction scores, as illustrated in Fig. 2. Under the correction effect of the weights, our model not only avoids missing action segments but also accurately recognizes complete actions. This results confirm the effectiveness of the PKCO module in improving the accuracy of prediction scores.

4.4.3 Hyper-Parameter of Fixed-Length Queues L

The hyper-parameter of the fixed length queue during training is examined to investigate the best choice for our SARL module, and the performance influence are depicted in Fig. 3. From Fig. 3, we observe that the model shows the most promising performance when L take 45. This highlights that with longer sequences, the number of segments per action category increases. Consequently, an excessive number of negative samples during loss function computation hampers the model's learning of positive samples, ultimately diminishing its generalization ability.

4.4.4 Model Parameter and Runtime Speed

In both out proposed method PKCO and SARL modules, we do not add any extra parameters to be learned. Additionally, considering that there is no extra computational burden during inference, our inference time has hardly increased. The parameters as well as the inference speed are reported in Table 5.

4.5 Qualitative Comparison

We present a qualitative comparison with LACP in Fig. 2. It shows that our method achieves more precise localization of action instances. In the action *ClenAndJerk* example, LACP exhibits issues with the continuity of action detection, while our model is capable of detecting complete action instances without fragmentation. In the action *SoccerPenalty* example, LACP directly misses one action instance. This experiment demonstrates that our model effectively enhances the baseline's capability for completeness learning.

5 Conclusion

In this work, we proposes a novel framework for point-supervised temporal action localization to tackle the challenge of reliable dense labeling. In particular, we derive corrective weights by computing the similarity between representative features of actions and segments. These weights are employed to adjust prediction scores based on the feature distance relationship, thereby enhancing the reliability of dense labels. Furthermore, in order to reduce the impact of scene information on action representation and improve the robustness of action representative features, we introduce a novel loss that encourages feature similarity comparison among annotated segments across videos. Finally, we validate the effectiveness of LRDP-Net by conducting experiments that compared its performance against those of existing methods. Ablation experiments demonstrated that each component of the proposed method contributed performance improvements on P-WTAL.

Acknowledgements This work was supported in part by the Natural Science Foundation of China under Grant 62036009, 62276237, 62206250, the Zhejiang Provincial Natural Science Foundation of China under Grant LQ22F020007, and the Ten Thousand Talent Program of Zhejiang Province.

Declarations

Conflicts of interest The authors declare that there is no Conflict of interest.

References

1. Bi M, Li J, Liu X, Zhang Q, Yang Z (2023) Action-aware network with upper and lower limit loss for weakly-supervised temporal action localization. Neural Process Lett 55:4307–4324
2. Carreira J, Zisserman A (2017) Quo vadis, action recognition? a new model and the kinetics dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). 6299–6308
3. Chen C, Wang G, Peng C, Fang Y, Zhang D, Qin H (2021) Exploring rich and efficient spatial temporal interactions for real-time video salient object detection. IEEE Trans Image Process 30:3995–4007
4. Chen C, Wang G, Peng C, Zhang X, Qin H (2019) Improved robust video saliency detection based on long-term spatial-temporal information. IEEE Trans Image Process 29:1090–1010
5. Chen T, Kornblith S, Norouzi M, Hinton G (2020) A simple framework for contrastive learning of visual representations. In: Proceedings of the International Conference on Machine Learning (ICML). 1597–1607
6. Damen D, Leelasawassuk T, Mayol-Cuevas W (2016) You-do, i-learn: egocentric unsupervised discovery of objects and their modes of interaction towards video-based guidance. Comput Vis Image Underst 149:98–112
7. Dou P, Hu H (2023) Complementary attention network for weakly supervised temporal action localization. Neural Proc Lett 55:6713–6732
8. Fang Z, Fan J, Yu J (2023) Lpr: learning point-level temporal action localization through re-training. Multimedia Syst 29:2545–2562
9. Fu J, Gao J, Xu C (2022) Compact representation and reliable classification learning for point-level weakly-supervised action localization. IEEE Trans Image Process 31:7363–7377

10. He B, Yang X, Kang L, Cheng Z, Zhou X, Shrivastava A (2022) Asm-loc: action-aware segment modeling for weakly-supervised temporal action localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 13925–13935
11. He K, Fan H, Wu Y, Xie S, Girshick R (2020) Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR). 9729–9738
12. Huang L, Wang L, Li H (2022) Weakly supervised temporal action localization via representative snippet knowledge propagation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 3272–3281
13. Idrees H, Zamir AR, Jiang YG (2017) The thumos challenge on action recognition for videos "in the wild." Comput Vis Image Und 155:1–23
14. Ju C, Zhao P, Chen S, Zhang Y, Wang Y, Tian Q (2021) Divide and conquer for single-frame temporal action localization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). 13455–13464
15. Ju C, Zhao P, Zhang Y, Wang Y, Tian Q (2020) Point-level temporal action localization: bridging fully-supervised proposals to weakly-supervised losses , arXiv preprint arXiv:2012.08236
16. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980
17. Lee P, Byun H (2021) Learning action completeness from points for weakly-supervised temporal action localization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). 13648–13657
18. Lei P, Todorovic S (2018) Temporal deformable residual networks for action segmentation in videos. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). 6742–6751
19. Li B, Pan Y, Liu R, Zhu Y (2023) Separately guided context-aware network for weakly supervised temporal action detection. Neural Process Lett 55:6269–6288
20. Li Z, Abu Farha Y, Gall J (2021) Temporal action segmentation from timestamp supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 8365–8374
21. Ma F, Zhu L, Yang Y, Zha S, Kundu G, Feiszli M, Shou Z (2020) Sf-net: single-frame supervision for temporal action localization. In: Proceedings of the European Conference on Computer Vision (ECCV). 420–437
22. Mamshad Nayeem R, Mittal G, Yu Y, Hall M, Sajeev S, Shah M, Chen M (2023) Pivotal: Prior-driven supervision for weakly-supervised temporal action localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 22992–23002
23. Moltisanti D, Fidler S, Damen D (2019) Action recognition from single timestamp supervision in untrimmed videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 9915–9924
24. Ren H, Yang W, Zhang T, Zhang Y (2023) Proposal-based multiple instance learning for weakly-supervised temporal action localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2394–2404
25. Shi B, Dai Q, Mu Y, Wang J (2020) Weakly-supervised action localization by generative attention modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 1009–1019
26. Shi D, Zhong Y, Cao Q, Ma L, Li J, Tao D (2023) Tridet: Temporal action detection with relative boundary modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 18857–18866
27. Shou Z, Gao H, Zhang L, Miyazawa K, Chang S.F (2018) Autoloc: weakly-supervisedtemporalaction localizationinuntrimmedvideos. In: Proceedings of the European Conference on Computer Vision (ECCV). 154–171
28. Tian Y, Krishnan D, Isola P (2020) Contrastive multiview coding. In: Proceedings of the European Conference on Computer Vision (ECCV). 776–794
29. Wang G, Chen C, Fan D, Hao A, Qin H (2021) From semantic categories to fixations: a novel weakly-supervised visual-auditory saliency detection approach. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). 15119–15128
30. Xu M, Zhao C, Rojas D.S, Thabet A, Ghanem B (2020) G-tad: sub-graph localization for temporal action detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 10156–10165
31. Xu S, Luo W, Jia X (2023) Graph contrastive learning with constrained graph data augmentation. Neural Process Lett 55:10705–10726
32. Zeng R, Huang W, Tan M, Rong Y, Zhao P, Huang J, Gan C (2019) Graph convolutional networks for temporal action localization. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV). 7094–7103

33. Zeng R, Huang W, Tan M, Rong Y, Zhao P, Huang J, Gan C (2021) Graph convolutional module for temporal action localization in videos. IEEE Trans Pattern Anal Mach Intell (TPAMI) 44(10):6209–6223
34. Zhang C, Cao M, Yang D, Chen J, Zou Y(2021) Cola: weakly-supervised temporal action localization with snippet contrastive learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (ICCV). 16010–16019
35. Zhang C.L, Wu J, Li Y (2022) Actionformer: localizing moments of actions with transformers. In: Proceedings of the European Conference on Computer Vision (ECCV). 492–510
36. Zhang S, Chen F, Zhang J, Liu A, Wang F (2022) Multi-level self-supervised representation learning via triple-way attention fusion and local similarity optimization. Neural Process Lett 55:5763–5781
37. Zhao P, Xie L, Ju C, Zhang Y, Wang Y, Tian Q (2020) Bottom-up temporal action localization with mutual regularization. In: Proceedings of the European Conference on Computer Vision (ECCV). 539–555