SF-Net: Single-Frame Supervision for Temporal Action Localization

Fan Ma[1], Linchao Zhu[1], Yi Yang[1], Shengxin Zha[2], Gourab Kundu[2], Matt Feiszli[2], and Zheng Shou[2]

[1] University of Technology Sydney, Australia
[2] Facebook

Abstract. In this paper, we study an intermediate form of supervision, i.e., **single-frame supervision**, for temporal action localization (TAL). To obtain the single-frame supervision, the annotators are asked to identify only a single frame *within* the temporal window of an action. This can significantly reduce the labor cost of obtaining full supervision which requires annotating the action *boundary*. Compared to the weak supervision that only annotates the video-level label, the single-frame supervision introduces extra temporal action signals while maintaining low annotation overhead. To make full use of such single-frame supervision, we propose a unified system called **SF-Net**. First, we propose to predict an actionness score for each video frame. Along with a typical category score, the actionness score can provide comprehensive information about the occurrence of a potential action and aid the temporal boundary refinement during inference. Second, we mine pseudo action and background frames based on the single-frame annotations. We identify pseudo action frames by adaptively expanding each annotated single frame to its nearby, contextual frames and we mine pseudo background frames from all the unannotated frames across multiple videos. Together with the ground-truth labeled frames, these pseudo-labeled frames are further used for training the classifier. In extensive experiments on THUMOS14, GTEA, and BEOID, SF-Net significantly improves upon state-of-the-art weakly-supervised methods in terms of both segment localization and single-frame localization. Notably, SF-Net achieves comparable results to its fully-supervised counterpart which requires much more resource intensive annotations. The code is available at `https://github.com/Flowerfan/SF-Net`.

Keywords: single-frame annotation action localization

1 Introduction

Recently, weakly-supervised Temporal Action Localization (TAL) has attracted substantial interest. Given a training set containing only video-level labels, we aim to detect and classify each action instance in long, untrimmed testing videos. In the fully-supervised annotation, the annotators usually need to rollback the

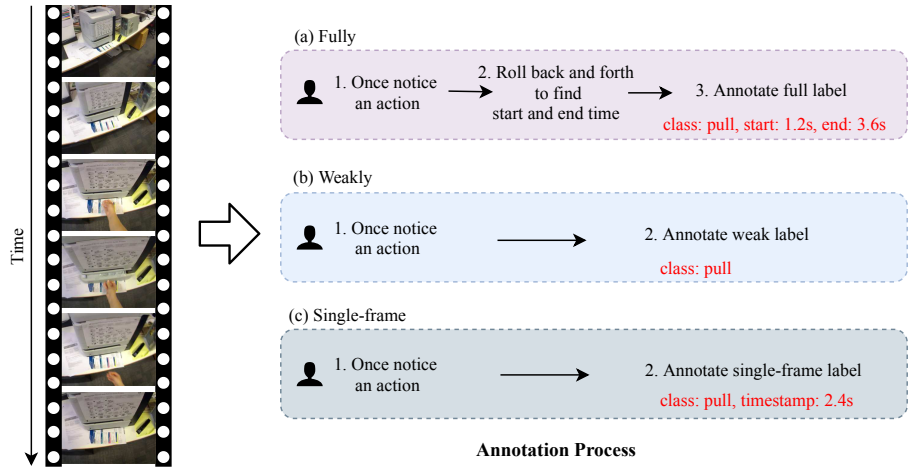

Fig. 1. Different ways of annotating actions while watching a video. (a) Annotating actions in the fully-supervised way. The start and end time of each action instance are required to be annotated. (b) Annotating actions in the weakly-supervised setting. Only action classes are required to be given. (c) Annotating actions in our single-frame supervision. Each action instance should have one timestamp. Note that the time is automatically generated by the annotation tool. Compared to the weakly-supervised annotation, the single-frame annotation requires only a few extra pauses to annotate repeated seen actions in one video.

video for repeated watching to give the precise temporal boundary of an action instance when they notice an action while watching the video [41]. For the weakly-supervised annotation, annotators just need to watch the video once to give labels. They can record the action class once they notice an unseen action. This significantly reduces annotation resources: video-level labels use fewer resources than annotating the start and end times in the fully-supervised setting.

Despite the promising results achieved by state-of-the-art weakly-supervised TAL work [25,27,29], their localization performance is still inferior to fully-supervised TAL work [6,18,28]. In order to bridge this gap, we are motivated to utilize single-frame supervision [23]: for each action instance, only one single positive frame is pointed out. The annotation process for single-frame supervision is almost the same as it in the weakly-supervised annotation. The annotators only watch the video once to record the action class and timestamp when they notice each action. It significantly reduces annotation resources compared to full supervision.

In the image domain, Bearman et al. [2] were the first to propose point supervision for image semantic segmentation. Annotating at point-level was extended by Mettes et al. [22] to video domain for spatio-temporal localization, where each action frame requires one spatial point annotation during training. Moltisanti et al. [23] further reduced the required resources by proposing single-frame super-

vision and developed a method that selects frames with a very high confidence score as pseudo action frames.

However, [23] was designed for whole video classification. In order to make full use of single-frame supervision for our TAL task, the unique challenge of **localizing temporal boundaries of actions** remains unresolved. To address this challenge in TAL, we make three innovations to improve the localization model's capability in distinguishing background frames and action frames. First, we predict "actionness" at each frame which indicates the probability of being any actions. Second, based on the actionness, we investigate a novel background mining algorithm to determine frames that are likely to be the background and leverage these pseudo background frames as additional supervision. Third, when labeling pseudo action frames, besides the frames with high confidence scores, we aim to determine more pseudo action frames and thus propose an action frame expansion algorithm.

In addition, for many real-world applications, detecting precise start time and end time is overkill. Consider a reporter who wants to find some car accident shots in an archive of street camera videos: it is sufficient to retrieve a single frame for each accident, and the reporter can easily truncate clips of desired lengths. Thus, in addition to evaluating traditional segment localization in TAL, we also propose a new task called single-frame localization, which requires only localizing one frame per action instance.

In summary, our contributions are three-fold:

(1) To our best knowledge, this is the first work to use single-frame supervision for the challenging problem of localizing temporal boundaries of actions. We show that the single-frame annotation significantly saves annotation time compared to fully-supervised annotation.

(2) We find that single-frame supervision can provide strong cue about the background. Thus, from frames that are not annotated, we propose two novel methods to mine likely background frames and action frames, respectively. These likely background and action timestamps are further used as pseudo ground truth for training.

(3) We conduct extensive experiments on three benchmarks, and the performances on both segment localization and single-frame localization tasks are largely boosted.

2 Related Work

Action recognition. Action recognition has recently witnessed an increased focus on trimmed videos. Both temporal and spatial information is significant for classifying the video. Early works mainly employed hand-crafted features to solve this task. IDT [34] had been widely used across many video-related tasks. Recently, various deep neural networks were proposed to encode spatial-temporal video information. Two-stream network [31] adopted optical flow to learn temporal motion, which had been used in many latter works [4,31,36]. Many 3D convolutional networks [4,33,11] are also designed to learn action embeddings.

Beyond fully-supervised action recognition, a few works focus on self-supervised video feature learning [43] and few-shot action recognition [44]. In this paper, we focus on single-frame supervision for temporal action localization.

Point supervision. Bearman et al. [2] first utilized the point supervision for image semantic segmentation. Mettes et al. [22] extended it to spatio-temporal localization in video, where the action is pointed out by one spatial location in each action frame. We believe this is overkill for temporal localization, and demonstrate that single-frame supervision can achieve very promising results already. Recently, single-frame supervision has been used in [23] for video-level classification, but this work does not address identifying temporal boundaries. Note that Alwassel et al. [1] proposed to spot action in the video during inference time but targeted detecting one action instance per class in one video while our proposed single-frame localization task aims to detect every instance in one video.

Fully-supervised temporal action localization. Approaches of temporal action localization trained in full supervision have mainly followed a proposal-classification paradigm [6,8,12,18,30,28], where temporal proposals are generated first and then classified. Other categories of methods, including sequential decision-making [1] and single-shot detectors [17] have also been studied. Given full temporal boundary annotations, the proposal-classification methods usually filter out the background frames at the proposal stage via a binary actionness classifier. Activity completeness has also been studied in the temporal action localization task. Zhao et al. [42] used a structural temporal pyramid pooling followed by an explicit binary classifier to evaluate the completeness of an action instance. Yuan et al. [38] structured an action into three parts to model its temporal evolution. Chéron *et al.* [7] handled the spatio-temporal action localization with various supervisions. Long et al. [21] proposed a Gaussian kernel to dynamically optimize temporal scale of action proposals. However, these methods use fully temporal annotations, which are resource intensive.

Weakly-supervised temporal action localization. Multiple Instance Learning (MIL) has been widely used in weakly-supervised temporal action localization. Without temporal boundary annotations, temporal action score sequence has been widely used to generate action proposals [35,25,19,24]. Wang et al. [35] proposed UntrimmedNet composed of a classification module and a selection module to reason about the temporal duration of action instances. Nguyen *et al.* [25] introduced a sparsity regularization for video-level classification. Shou *et al.* [29] and Liu [20] investigated score contrast in the temporal dimension. Hide-and-Seek [32] randomly removed frame sequences during training to force the network to respond to multiple relevant parts. Liu *et al.* [19] proposed a multi-branch network to model the completeness of actions. Narayan *et al.* [24] introduced three-loss forms to guide the learning discriminative action features with enhanced localization capabilities. Nguyen [26] used attention modules to detect foreground and background for detecting actions. Despite the improvements over time, the performances of weakly-supervised methods are still inferior to the fully-supervised method.

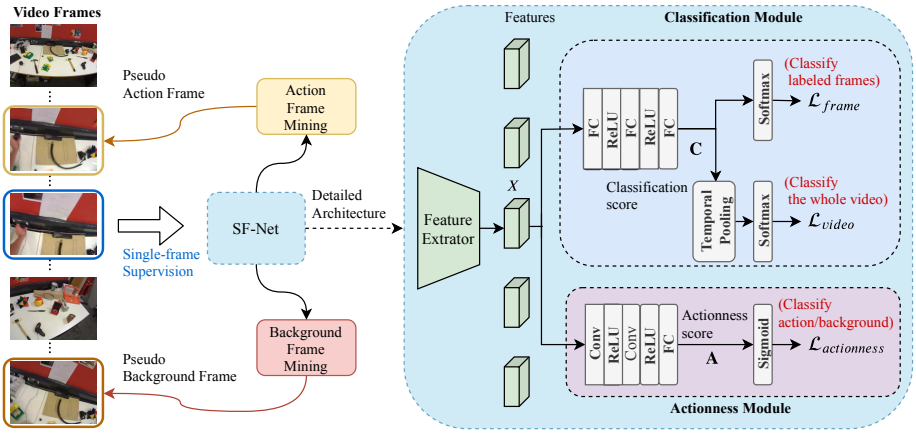

Fig. 2. Overall training framework of our proposed SF-Net. Given single-frame supervision, we employ two novel frame mining strategies to label pseudo action frames and background frames. The detailed architecture of SF-Net is shown on the right. SF-Net consists of a classification module to classify each labeled frame and the whole video, and an actionness module to predict the probability of each frame being action. The classification module and actionness module are trained jointly with three losses explained in Sec. 3.3.

3 Method

In this section, we define our tasks, present architecture of our SF-Net, and finally discuss details of training and inference, respectively.

3.1 Problem Definition

A training video can contain multiple action classes and multiple action instances. Unlike the full supervision setting, which provides temporal boundary annotation of each action instance, in our single-frame supervision setting, each instance only has one frame pointed out by annotator with timestamp t and action class y. Note that $y \in \{1, \ldots, N_c\}$ where N_c is the total number of classes and we use index 0 to represent the background class.

Given a testing video, we perform two temporal localization tasks: (1) **Segment localization**. We detect the start time and end time for each action instance with its action class prediction. (2) **Single-frame localization**. We output the timestamp of each detected action instance with its action class prediction. The evaluation metrics for these two tasks are explained in Sec. 4.

3.2 Framework

Overview. Our overall framework is presented in Fig. 2. During training, learning from single-frame supervision, SF-Net mines pseudo action and background

frames. Based on the labeled frames, we employ three losses to jointly train a classification module to classify each labeled frame and the whole video, and an actionness module to predict the probability of each frame being action. In the following, we outline the framework while details of frame mining strategies and different losses are described in Sec. 3.3.

Feature extraction. For a training batch of N videos, the features of all frames are extracted and stored in a feature tensor $X \in R^{N\times T\times D}$, where D is the feature dimension, and T is the number of frames. As different videos vary in the temporal length, we simply pad zeros when the number of frames in a video is less than T.

Classification module. The classification module outputs the score of being each action class for all frames in the input video. To classify each labeled frame, we feed X into three Fully-Connected (FC) layers to get the classification score $C \in \mathcal{R}^{N\times T\times N_c+1}$. The classification score C is then used to compute frame classification loss $\mathcal{L}_{frame}$. We also pool C temporally as described in [24] to compute video-level classification loss $\mathcal{L}_{video}$.

Actionness module. As shown in Fig. 2, our model has an actionness branch of identifying positive action frames. Different from the classification module, the actionness module only produces a scalar for each frame to denote the probability of being contained in an action segment. To predict an actionness score, we feed X into two temporal convolutional layers followed by one FC layer, resulting in an actionness score matrix $A \in \mathcal{R}^{N\times T}$. We apply sigmoid on A and then compute a binary classification loss $\mathcal{L}_{actionness}$.

3.3 Pseudo Label Mining and Training Objectives

Action classification at labeled frames. We use cross entropy loss for the action frame classification. As there are NT frames in the input batch of videos and most of the frames are unlabeled, we first filter the labeled frames for classification. Suppose we have K labeled frames where $K \ll NT$. We can get classification activations of K labeled frames from C. These scores are fed to a Softmax layer to get classification probability $\mathbf{p}^l \in \mathcal{R}^{K\times N_c+1}$ for all labeled frames. The classification loss of annotated frames in the batch of videos is formulated as:

$$\mathcal{L}^l_{frame} = -\frac{1}{K}\sum_{i}^{K}\mathbf{y}_i \log \mathbf{p}^l_i, \tag{1}$$

where the $\mathbf{p}^l_i$ denote the prediction for the i^{th} labeled action frame.

Pseudo labeling of frames. With only a single label per action instance, the total number of positive examples is quite small and may be difficult to learn from. While we do not use full temporal annotation, it is clear that actions are longer events spanning consecutive frames. To increase the temporal information available to the model, we design an action frame mining and a background frame mining strategy to introduce more frames into the training process.

(a) Action frame mining: We treat each labeled action frame as an anchor frame for each action instance. We first set the expand radius r to limit the maximum expansion distance to the anchor frame at t. Then we expand the past from $t-1$ frame and the future from $t+1$ frame, separately. Suppose the action class of the anchor frame is represented by y_i. If the current expanding frame has the same predicted label with the anchor frame, and the classification score at y_i class is higher than that score of the anchor frame multiplying a predefined value ξ, we then annotate this frame with label y_i and put it into the training pool. Otherwise, we stop the expansion process for the current anchor frame.

(b) Background frame mining: The background frames are also important and widely used in localization methods [19,26] to boost the model performance. Since there is no background label under the single-frame supervision, our proposed model manages to localize background frames from all the unlabeled frames in the N videos. At the beginning, we do not have supervision about where the background frames are. But explicitly introducing a background class can avoid forcing classifying a frame into one of the action classes. Our proposed background frame mining algorithm can offer us the supervision needed for training such a background class so as to improve the discriminability of the classifier. Suppose we try to mine ηK background frames, we first gather the classification scores of all unlabeled frames from C. The η is the ratio of background frames to labeled frames. These scores are then sorted along background class to select the top ηK scores $\mathbf{p}^b \in \mathcal{R}^{\eta K}$ as the score vector of the background frames. The pseudo background classification loss is calculated on the top ηK frames by,

$$\mathcal{L}^b_{frame} = -\frac{1}{\eta K}\sum \log\, \mathbf{p}^b, \tag{2}$$

The background frame classification loss assists the model with identifying irrelevant frames. Different from background mining in [19,26] which either require extra computation source to generate background frames or adopt a complicated loss for optimization, we mining background frames across multiple videos and use the classification loss for optimization. The selected pseudo background frames may have some noises in the initial training rounds. As the training evolves and the classifiers discriminability improves, we are able to reduce the noises and detect background frames more correctly. With the more correct background frames as supervision signals, the classifiers discriminability can be further boosted. In our experiments, we observed that this simple background mining strategy allows for better action localization results. We incorporate the background classification loss with the labeled frame classification loss to formulate the single-frame classification loss

$$\mathcal{L}_{frame} = \mathcal{L}^l_{frame} + \frac{1}{N_c}\mathcal{L}^b_{frame} \tag{3}$$

where N_c is the number of action classes to leverage the influence from background class.

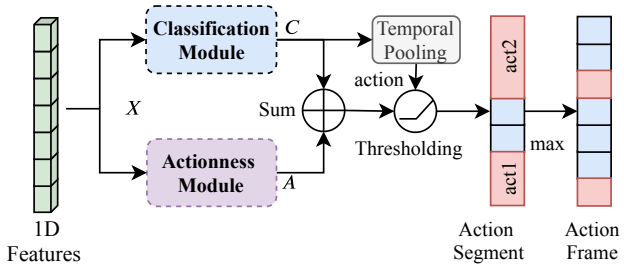

Fig. 3. The inference framework of SF-Net. The classification module outputs the classification score C of each frame for identifying possible target actions in the given video. The action module produces the actionness score determining the possibility of a frame containing target actions. The actionness score together with the classification score are used to generate action segment based on the threshold.

Actionness prediction at labeled frames. In fully-supervised TAL, various methods learn to generate action proposals that may contain the potential activities [37,18,6]. Motivated by this, we design the actionness module to make the model focus on frames relevant to target actions. Instead of producing the temporal segment in proposal methods, our actionness module produces the actionness score for each frame. The actionness module is in parallel with the classification module in our SF-Net. It offers extra information for temporal action localization. We first gather the actionness score $A^l \in \mathcal{R}^K$ of labeled frames in the training videos. The higher the value for a frame, the higher probability of that frame belongs to a target action. We also use the background frame mining strategy to get the actionness score $A^b \in \mathcal{R}^{\eta K}$. The actionness loss is calculated by,

$$\mathcal{L}_{actionness} = -\frac{1}{K}\sum \log \sigma(A^l) - \frac{1}{\eta K}\sum \log(1-\sigma(A^b)), \tag{4}$$

where σ is the sigmoid function to scale the actionness score to $[0, 1]$.

Full objective. We employ video-level loss as described in [24] to tackle the problem of multi-label action classification at video-level. For the i^{th} video, the top-k activations per category (where $k = T_i/8$) of the classification activation $C(i)$ are selected and then are averaged to obtain a class-specific encoding $r_i \in \mathcal{R}^{C+1}$ as in [27,24]. We average all the frame label predictions in the video v_i to get the video-level ground-truth $q_i \in \mathcal{R}^{N_c+1}$. The video-level loss is calculated by

$$\mathcal{L}_{video} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N_c} q_i(j) \log \frac{\exp(r_i(j))}{\sum_{N_c+1}\exp(r_i(k))}, \tag{5}$$

where $q_i(j)$ is the j^{th} value of q_i representing the probability mass of video v_i belong to j^{th} class.

Consequently, the total training objective for our proposed method is

$$\mathcal{L} = \mathcal{L}_{frame} + \alpha\mathcal{L}_{video} + \beta\mathcal{L}_{actionness}, \tag{6}$$

where $\mathcal{L}_{frame}$, $\mathcal{L}_{video}$, and $\mathcal{L}_{actionness}$, denote the frame classification loss, video classification loss, and actionness loss, respectively. α and β are the hyper-parameters leveraging different losses.

3.4 Inference

During the test stage, we need to give the temporal boundary for each detected action. We follow previous weakly-supervised work [25] to predict video-level labels by temporally pooling and thresholding on the classification score. As shown in Fig. 3, we first obtain the classification score C and actionness score A by feeding input features of a video to the classification module and actionness module. Towards segment localization, we follow the thresholding strategy in [25,26] to keep the action frames above the threshold and consecutive action frames constitute an action segment. For each predicted video-level action class, we localize each action segment by detecting an interval that the sum of classification score and actionness score exceeds the preset threshold at every frame inside the interval. We simply set the confidence score of the detected segment to the sum of its highest frame classification score and the actionness score. Towards single frame localization, for the action instance, we choose the frame with the maximum activation score in the detected segment as the localized action frame.

4 Experiment

4.1 Datasets

THUMOS14. There are 1010 validation and 1574 test videos from 101 action categories in THUMOS14 [13]. Out of these, 20 categories have temporal annotations in 200 validation and 213 test videos. The dataset is challenging, as it contains an average of 15 activity instances per video. Similar to [14, 16], we use the validation set for training and test set for evaluating our framework.
GTEA. There are 28 videos of 7 fine-grained types of daily activities in a kitchen contained in GTEA [15]. An activity is performed by four different subjects, and each video contains about 1800 RGB frames, showing a sequence of 7 actions, including the background action.
BEOID. There are 58 videos in BEOID [9]. There is an average of 12.5 action instances per video. The average length is about 60s, and there are 30 action classes in total. We randomly split the untrimmed videos in an 80-20% proportion for training and testing, as described in [23].

Table 1. Comparison between different methods for simulating single-frame supervision on THUMOS14. "Annotation" means that the model uses human annotated frame for training. "TS" denotes that the single-frame is sampled from action instances using a uniform distribution, while "TS in GT" is using a Gaussian distribution near the mid timestamp of each activity. The AVG for segment localization is the average mAP from IoU 0.1 to 0.7.

Position	mAP@hit	Segment mAP@IoU			
		0.3	0.5	0.7	AVG
Annotation	**60.2**±0.70	**53.3**±0.30	28.8±0.57	9.7±0.35	**40.6**±0.40
TS	57.6±0.60	52.0±0.35	**30.2**±0.48	**11.8**±0.35	40.5±0.28
TS in GT	52.8±0.85	47.4±0.72	26.2±0.64	9.1±0.41	36.7±0.52

4.2 Implementation Details

We use I3D network [4] trained on the Kinetics [5] to extract video features. For the RGB stream, we rescale the smallest dimension of a frame to 256 and perform the center crop of size 224×224. For the flow stream, we apply the TV-L1 optical flow algorithm [39]. We follow the two-stream fusion operation in [24] to integrate predictions from both appearance (RGB) and motion (Flow) branches. The inputs to the I3D models are stacks of 16 frames.

On all datasets, we set the learning rate to 10^{-3} for all experiments, and the model is trained with a batch size of 32 using the Adam [14]. Loss weight hyper-parameters α and β are set to 1. The model performance is not sensitive to these hyper-parameters. For the hyper-parameter η used in mining background frames, we set it to 5 on THUMOS14 and set it to 1 on the other two datasets. The number of iterations is set to 500, 2000 and 5000 for GTEA, BEOID and THUMOS14, respectively.

4.3 Evaluation Metrics

(1) **Segment localization**: We follow the standard protocol, provided with the three datasets, for evaluation. The evaluation protocol is based on mean Average Precision (mAP) for different intersection over union (IoU) values for the action localization task.
(2) **Single-frame localization**: We also use mAP to compare performances. Instead of measuring IoU, the predicted single-frame is regarded as correct when it lies in the temporal area of the ground-truth segment, and the class label is correct. We use mAP@hit to denote the mean average precision of selected action frame falling in the correct action segment.

4.4 Annotation Analysis

Single-frame supervision simulation. First, to simulate the single-frame supervision based on ground-truth boundary annotations existed in the above three datasets, we explore the different strategies to sample a single-frame for

Table 2. Single-frame annotation differences between different annotators on three datasets. We show the number of action segments annotated by Annotator 1, Annotator 2, Annotator 3, and Annotator 4. In the last column, we report the total number of the ground-truth action segments for each dataset.

Datasets	Annotator 1	Annotator 2	Annotator 3	Annotator 4	# of total segments
GTEA	369	366	377	367	367
BEOID	604	602	589	599	594
THUMOS14	3014	2920	2980	2986	3007

each action instance. We follow the strategy in [23] to generate single-frame annotations with uniform and Gaussian distribution (**Denoted by TS and TS in GT**). We report the segment localization at different IoU thresholds and frame localization results on THUMOS14 in Table 1. The model with each single-frame annotation is trained five times. The mean and standard deviation of mAP is reported in the Table. Compared to models trained on sampled frames, the model trained on human annotated frames achieves the highest mAP@hit. As the the action frame is the frame with the largest prediction score in the prediction segment, the model with higher mAP@hit can assist with localizing action timestamp more accurately when people need to retrieve the frame of target actions. When sampling frames are from near middle timestamps to the action segment (TS in GT), the model performs inferior to other models as these frames may not contain informative elements of complete actions. For the segment localization result, the model trained on truly single-frame annotations achieves higher mAP at small IoU thresholds, and the model trained on frames sampled uniformly from the action instance gets higher mAP at larger IoU thresholds. It may be originated by sampled frames of uniform distribution containing more boundary information for the given action instances.

Single-frame annotation. We also invite four annotators with different backgrounds to label a single frame for each action segment on three datasets. More details of annotation process can be found in the supplementary material. In Table 2, we have shown the action instances of different datasets annotated by different annotators. The ground-truth in the Table denotes the action instances annotated in the fully-supervised setting. From the Table, we obtain that the number of action instances by different annotators have a very low variance. The number of labeled frames is very close to the number of action segments in the fully-supervised setting. This indicates that annotators have common justification for the target actions and hardly miss the action instance despite that they only pause once to annotate single-frame of each action.

We also present the distribution of the relative position of single-frame annotation to the corresponding action segment. As shown in Fig. 4, there are rare frames outside of the temporal range of action instances from the ground-truth in the fully-supervised setting. As the number of annotated single frames is almost the same as the number of action segments, we can draw the inference that

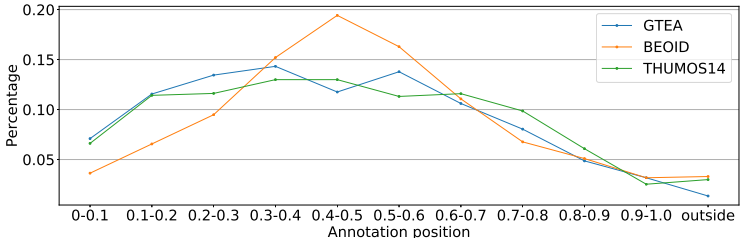

Fig. 4. Statistics of human annotated single-frame on three datasets. X-axis: single-frame falls in the relative portion of the whole action; Y-axis: percentage of annotated frames. We use different colors to denote annotation distribution on different datasets.

the single frame annotation includes all almost potential action instances. We obtain that annotators prefer to label frames near to the middle part of action instances. This indicates that humans can identify an action without watching the whole action segment. On the other hand, this will significantly reduce the annotation time compared with fully-supervised annotation as we can quickly skip the current action instance after single-frame annotation.

Annotation speed for different supervision. To measure the required annotation resource for different supervision, we conduct a study on GTEA. Four annotators are trained to be familiar with action classes in GTEA. We ask the annotator to indicate the video-level label, single-frame label and temporal boundary label of 21 videos lasting 93 minutes long. While watching, the annotator is able to skim quickly, pause, and go to any timestamp. On average, the annotation time used by each person to annotate 1-minute video is 45s for the video-level label, 50s for the single-frame label, and 300s for the segment label. The annotation time for single-frame label is close to the annotation time for video-level label but much fewer than time for the fully-supervised annotation.

4.5 Analysis

Effectiveness of each module, loss, and supervision. To analyze the contribution of the classification module, actionness module, background frame mining strategy, and the action frame mining strategy, we perform a set of ablation studies on THUMOS14, GTEA and BEOID datasets. The segment localization mAP at different thresholds is presented in Table 3. We also compare the model with only weak supervision and the model with full supervision. The model with weak supervision is implemented based on [24].

We observe that the model with single-frame supervision outperforms the weakly-supervised model. And large performance gain is obtained on GTEA and BEOID datasets as the single video often contains multiple action classes, while action classes in one video are fewer in THUMOS14. Both background frame mining strategy and action frame mining strategy boost the performance

Table 3. Segment localization mAP results at different IoU thresholds on three datasets. Weak denotes that only video-level labels are used for training. All action frames are used in the full supervision approach. SF uses extra single frame supervision with frame level classification loss. SFB means that pseudo background frames are added into the training, while the SFBA adopts the actionness module, and the SFBAE indicates the action frame mining strategy added in the model. For models trained on single-frame annotations, we report mean and standard deviation results of five runs. AVG is the average mAP from IoU 0.1 to 0.7.

Dataset	Models	mAP@IoU				
		0.1	0.3	0.5	0.7	AVG
GTEA	Full	58.1	40.0	22.2	14.8	31.5
	Weak	14.0	9.7	4.0	3.4	7.0
	SF	50.0±1.42	35.6±2.61	**21.6**±1.67	**17.7**±0.96	30.5±1.23
	SFB	52.9±3.84	34.9±4.72	17.2±3.46	11.0±2.52	28.0±3.53
	SFBA	52.6±5.32	32.7±3.07	15.3±3.63	8.5±1.95	26.4±3.61
	SFBAE	**58.0**±2.83	**37.9**±3.18	19.3±1.03	11.9±3.89	**31.0**±1.63
BEOID	Full	65.1	38.6	22.9	7.9	33.6
	Weak	22.5	11.8	1.4	0.3	8.7
	SF	54.1±2.48	24.1±2.37	6.7±1.72	1.5±0.84	19.7±1.25
	SFB	57.2±3.21	26.8±1.77	9.3±1.94	1.7±0.68	21.7±1.43
	SFBA	**62.9**±1.68	36.1±3.17	12.2±3.15	2.2±2.07	27.1±1.44
	SFBAE	**62.9**±1.39	**40.6**±1.8	**16.7**±3.56	**3.5**±0.25	**30.1**±1.22
THUMOS14	Full	68.7	54.5	34.4	16.7	43.8
	Weak	55.3	40.4	20.4	7.3	30.8
	SF	58.6±0.56	41.3±0.62	20.4±0.55	6.9±0.33	31.7±0.41
	SFB	60.8±0.65	44.5±0.37	22.9±0.38	7.8±0.46	33.9±0.31
	SFBA	68.7±0.33	52.3±1.21	28.2±0.42	**9.7**±0.51	39.9±0.43
	SFBAE	**70.0**±0.64	**53.3**±0.3	**28.8**±0.57	**9.7**±0.35	**40.6**±0.40

on BEOID and THUMOS14 by putting more frames into the training, the performance on GTEA decreases mainly due to that GTEA contains almost no background frame. In this case, it is not helpful to employ background mining and the actionness module which aims for distinguishing background against action. The actionness module works well for the BEOID and THUMOS14 datasets, although the actionness module only produces one score for each frame.

Comparisons with state-of-the-art. Experimental results on THUMOS14 testing set are shown in Table 4. Our proposed single-frame action localization method is compared to existing methods for weakly-supervised temporal action localization, as well as several fully-supervised ones. Our model outperforms the previous weakly-supervised methods at all IoU thresholds regardless of the choice of feature extraction network. The gain is substantial even though only one single-frame for each action instance is provided. The model trained on human annotated frames achieves higher mAP at lower IoU compared to model trained

Table 4. Segment localization results on THUMOS14 dataset. The mAP values at different IoU thresholds are reported, and the column AVG indicates the average mAP at IoU thresholds from 0.1 to 0.5. * denotes the single-frame labels are simulated based on the ground-truth annotations. # denotes single-frame labels are manually annotated by human annotators.

Supervision	Method	mAP @IoU							
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	AVG
Full	S-CNN [30]	47.7	43.5	36.3	28.7	19.0	-	5.3	35.0
Full	CDC [28]	-	-	40.1	29.4	23.3	-	7.9	-
Full	R-C3D [37]	54.5	51.5	44.8	35.6	28.9	-	-	43.1
Full	SSN [42]	60.3	56.2	50.6	40.8	29.1	-	-	47.4
Full	Faster- [6]	59.8	57.1	53.2	48.5	42.8	**33.8**	**20.8**	52.3
Full	BMN [16]	-	-	56.0	47.4	38.8	29.7	20.5	-
Full	P-GCN [40]	**69.5**	**67.8**	**63.6**	**57.8**	**49.1**	-	-	**61.6**
Weak	Hide-and-Seek [32]	36.4	27.8	19.5	12.7	6.8	-	-	20.6
Weak	UntrimmedNet [35]	44.4	37.7	28.2	21.1	13.7	-	-	29.0
Weak	W-TALC [10]	49.0	42.8	32.0	26.0	18.8	-	6.2	33.7
Weak	AutoLoc [29]	-	-	35.8	29.0	21.2	13.4	5.8	-
Weak	STPN [25]	52.0	44.7	35.5	25.8	16.9	9.9	4.3	35.0
Weak	W-TALC [27]	55.2	49.6	40.1	31.1	22.8	-	7.6	39.7
Weak	Liu *et al.* [19]	57.4	50.8	41.2	32.1	23.1	15.0	7.0	40.9
Weak	Nguyen *et al.* [26]	**60.4**	**56.0**	**46.6**	**37.5**	**26.8**	**17.6**	**9.0**	**45.5**
Weak	3C-Net [24]	59.1	53.5	44.2	34.1	26.6	-	8.1	43.5
Single-frame simulation*	Moltisanti *et al.* [23]	24.3	19.9	15.9	12.5	9.0	-	-	16.3
Single-frame simulation*	SF-Net	68.3	62.3	52.8	**42.2**	**30.5**	**20.6**	**12.0**	51.2
Single-frame#	SF-Net	**71.0**	**63.4**	**53.2**	40.7	29.3	18.4	9.6	**51.5**

on sampling frames uniformly from action segments. The differences come from the fact that the uniform sampling frames from ground-truth action segments contain more information about temporal boundaries for different actions. As there are many background frames in the THUMOS14 dataset, the single frame supervision assists the proposed model with localizing potential action frames among the whole video. Note that the supervised methods have the regression module to refine the action boundary, while we simply threshold on the score sequence and still achieve comparable results.

5 Conclusions

In this paper, we have investigated how to leverage single-frame supervision to train temporal action localization models for both segment localization and single-frame localization during inference. Our SF-Net makes full use of single-frame supervision by predicting actionness score, pseudo background frame mining and pseudo action frame mining. SF-Net significantly outperforms weakly-supervised methods in terms of both segment localization and single-frame localization on three standard benchmarks.

Acknowledgements. The authors from UTS were partially supported by ARC DP200100938 and Facebook.

References

1. Alwassel, H., Caba Heilbron, F., Ghanem, B.: Action search: Spotting actions in videos and its application to temporal action localization. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 251–266 (2018)
2. Bearman, A., Russakovsky, O., Ferrari, V., Fei-Fei, L.: Whats the point: Semantic segmentation with point supervision. In: European conference on computer vision. pp. 549–565. Springer (2016)
3. Caba Heilbron, F., Escorcia, V., Ghanem, B., Carlos Niebles, J.: Activitynet: A large-scale video benchmark for human activity understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 961–970 (2015)
4. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
5. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
6. Chao, Y.W., Vijayanarasimhan, S., Seybold, B., Ross, D.A., Deng, J., Sukthankar, R.: Rethinking the faster r-cnn architecture for temporal action localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1130–1139 (2018)
7. Chéron, G., Alayrac, J.B., Laptev, I., Schmid, C.: A flexible model for training action localization with varying levels of supervision. In: Advances in Neural Information Processing Systems. pp. 942–953 (2018)
8. Dai, X., Singh, B., Zhang, G., Davis, L.S., Qiu Chen, Y.: Temporal context network for activity localization in videos. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 5793–5802 (2017)
9. Damen, D., Leelasawassuk, T., Haines, O., Calway, A., Mayol-Cuevas, W.W.: You-do, i-learn: Discovering task relevant objects and their modes of interaction from multi-user egocentric video. In: BMVC. vol. 2, p. 3 (2014)
10. Ding, L., Xu, C.: Weakly-supervised action segmentation with iterative soft boundary assignment. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6508–6516 (2018)
11. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
12. Gao, J., Yang, Z., Nevatia, R.: Cascaded boundary regression for temporal action detection. arXiv preprint arXiv:1705.01180 (2017)
13. Idrees, H., Zamir, A.R., Jiang, Y.G., Gorban, A., Laptev, I., Sukthankar, R., Shah, M.: The thumos challenge on action recognition for videos in the wild. Computer Vision and Image Understanding **155**, 1–23 (2017)
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
15. Lei, P., Todorovic, S.: Temporal deformable residual networks for action segmentation in videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6742–6751 (2018)
16. Lin, T., Liu, X., Li, X., Ding, E., Wen, S.: Bmn: Boundary-matching network for temporal action proposal generation. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)

17. Lin, T., Zhao, X., Shou, Z.: Single shot temporal action detection. In: Proceedings of the 25th ACM international conference on Multimedia. pp. 988–996. ACM (2017)
18. Lin, T., Zhao, X., Su, H., Wang, C., Yang, M.: Bsn: Boundary sensitive network for temporal action proposal generation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 3–19 (2018)
19. Liu, D., Jiang, T., Wang, Y.: Completeness modeling and context separation for weakly supervised temporal action localization. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
20. Liu, Z., Wang, L., Zhang, Q., Gao, Z., Niu, Z., Zheng, N., Hua, G.: Weakly supervised temporal action localization through contrast based evaluation networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3899–3908 (2019)
21. Long, F., Yao, T., Qiu, Z., Tian, X., Luo, J., Mei, T.: Gaussian temporal awareness networks for action localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 344–353 (2019)
22. Mettes, P., Van Gemert, J.C., Snoek, C.G.: Spot on: Action localization from pointly-supervised proposals. In: European conference on computer vision. pp. 437–453. Springer (2016)
23. Moltisanti, D., Fidler, S., Damen, D.: Action recognition from single timestamp supervision in untrimmed videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9915–9924 (2019)
24. Narayan, S., Cholakkal, H., Khan, F.S., Shao, L.: 3c-net: Category count and center loss for weakly-supervised action localization. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
25. Nguyen, P., Liu, T., Prasad, G., Han, B.: Weakly supervised action localization by sparse temporal pooling network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6752–6761 (2018)
26. Nguyen, P.X., Ramanan, D., Fowlkes, C.C.: Weakly-supervised action localization with background modeling. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
27. Paul, S., Roy, S., Roy-Chowdhury, A.K.: W-talc: Weakly-supervised temporal activity localization and classification. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 563–579 (2018)
28. Shou, Z., Chan, J., Zareian, A., Miyazawa, K., Chang, S.F.: Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
29. Shou, Z., Gao, H., Zhang, L., Miyazawa, K., Chang, S.F.: Autoloc: Weakly-supervised temporal action localization in untrimmed videos. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 154–171 (2018)
30. Shou, Z., Wang, D., Chang, S.F.: Temporal action localization in untrimmed videos via multi-stage cnns. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
31. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in neural information processing systems. pp. 568–576 (2014)
32. Singh, K.K., Lee, Y.J.: Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 3544–3553. IEEE (2017)

33. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: The IEEE International Conference on Computer Vision (ICCV) (December 2015)
34. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: The IEEE International Conference on Computer Vision (ICCV) (December 2013)
35. Wang, L., Xiong, Y., Lin, D., Van Gool, L.: Untrimmednets for weakly supervised action recognition and detection. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 4325–4334 (2017)
36. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Val Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: ECCV (2016)
37. Xu, H., Das, A., Saenko, K.: R-c3d: Region convolutional 3d network for temporal activity detection. In: Proceedings of the IEEE international conference on computer vision. pp. 5783–5792 (2017)
38. Yuan, Z., Stroud, J.C., Lu, T., Deng, J.: Temporal action localization by structured maximal sums. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3684–3692 (2017)
39. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime tv-l 1 optical flow. In: Joint pattern recognition symposium. pp. 214–223. Springer (2007)
40. Zeng, R., Huang, W., Tan, M., Rong, Y., Zhao, P., Huang, J., Gan, C.: Graph convolutional networks for temporal action localization. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
41. Zhao, H., Torralba, A., Torresani, L., Yan, Z.: Hacs: Human action clips and segments dataset for recognition and temporal localization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 8668–8678 (2019)
42. Zhao, Y., Xiong, Y., Wang, L., Wu, Z., Tang, X., Lin, D.: Temporal action detection with structured segment networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2914–2923 (2017)
43. Zhu, L., Yang, Y.: Actbert: Learning global-local video-text representations. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
44. Zhu, L., Yang, Y.: Label independent memory for semi-supervised few-shot video classification. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020). https://doi.org/10.1109/TPAMI.2020.3007511

Appendix

A Single-frame Annotation

We invite four annotators with different backgrounds to label single-frames for all actions intances. Before annotating each dataset, four annotators have watched a few video examples containing different actions to be familiar with action classes. They are asked to annotate one single frame for each target action instance while watching the video by our designed annotation tool. Specifically, they are required to pause the video when they identify an action instance and choose the action class that the paused frame belongs to. Once they have chosen the action class, they need to continue watching the video and record the frames for the next target action instances. After watching the whole video, the annotator should press the generation button and the annotation tool will then automatically produce the timestamps and action classes of all operated frames for the given video. Compared to the annotation process in the weakly-supervised setting, this results into almost no extra time cost since the timestamps are automatically generated. The single-frame annotation process is much faster than annotating the temporal boundary of each action in which the annotator often watches the video many times to define the start and end timestamp of a given action.

A.1 Annotation guideline

Different people may have different understandings of what constitutes a given action. To reduce the ambiguity, we prepare a detailed annotation guideline, which includes both clear action definitions as well as positive/negative examples with detailed clarifications for each action. For each action, we give (1) textual action definition for single-frame annotation, (2) positive single-frame annotations, and (3) segmented action instances for annotator to be familiar with.

A.2 Annotation tool

Our annotation tool supports automatically recording timestamp for annotating single-frame. This makes the annotation process faster when annotators notice an action and ready to label the paused frame. The interface of our annotation tool is presented in Figure 5. After watching a whole video, the annotator can press the generate button, the annotation results will be automatically saved into a csv file. When annotators think they made a wrong annotation, they can delete it at any time while watching the video. We have shown the one annotation example in the supplementary file. We have uploaded a video in the supplementary file to show how to annotate single-frame while watching the video.

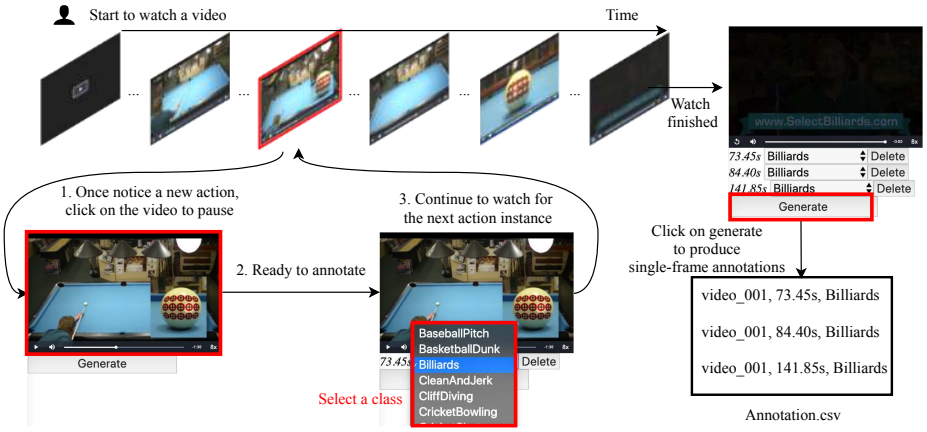

Fig. 5. Interface for annotating a single frame. First step is to pause the video when annotators notice an action while watching the video. The second step is to select the target class for the paused frame. After annotating an action instance, the annotator can click the video to keep watching the video for the next action instance. Note that the time is automatically generated by the annotation tool. After watching a whole video, the annotator can press the generate button to save all records into a csv file.

A.3 Quality control

We make two efforts to improve the annotation quality. First of all, each video is labeled by four annotators, and the annotated single-frames of a view are randomly selected during experiments to reduce annotation bias. Secondly, we train annotators before annotating videos and make sure that they can notice target actions while watching the video.

B Action Frame Mining

The action frame mining strategy is described in Algorithm 1. We treat the labeled action frame as the anchor frame and expand frames around it. We use a threshold ξ and the label consistency with neighbors to decide whether to add the unlabeled frame or not. The expanded frames are annotated with the same label as the anchor frame. As shown in Algorithm 1, we expand the frames at $t-1$ to the anchor frame. We first gather the classification score of three frames around $t-1$ frame. We then calculate the prediction classes for these three frames. If they all have the same predicted class and the classification score for the current frame at class y is above a threshold, we choose to put the current frame into the training frame set S. For all experiments in the current paper, we set $\xi = 0.9$ for fair comparison.

Algorithm 1 Action Frame Mining

Input: video classification activation $C \in \mathcal{R}^{T \times N_c+1}$, labeled action frame at time t belonging to action class y, expand radius $r = 5$, threshold $\xi = 0.9$.
Output: expanded frames set $\mathcal{S}$
gather classification score $C(t)$ for the anchor frame
$\mathcal{S} \leftarrow \{(t, y)\}$
function EXPAND(s) ;
    for $j \leftarrow 1; j \leq r$ **do**
        $\hat{y}_{past} \leftarrow \text{argmin } C(t + (j-1)s)$
        $\hat{y}_{current} \leftarrow \text{argmin } C(t + js)$
        $\hat{y}_{future} \leftarrow \text{argmin } C(t + (j+1)s)$
        if $\hat{y}_{past} == \hat{y}_{current} == \hat{y}_{future}$ and $C(t + js)_y \geq \xi C(t)_y$ **then**
            $\mathcal{S} \leftarrow (t + js, y)$
        end if
        $j \leftarrow j + s$
    end for
end function
EXPAND(-1)
EXPAND(1)
Return $\mathcal{S}$

Table 5. Classification accuracy and class-agnostic localization AP on THUMOS14.

	Classification	Class-agnostic localization		
	mAP	AP@IoU=0.3	AP@IoU=0.5	AP@IoU=0.7
Ours w/o single-frame	97.8	42.1	18.1	5.5
Ours w/ single-frame	**98.5**	**58.8**	**32.4**	**9.4**

C Evaluate Classification & Localization Independently

We evaluate our single-frame supervised model and weakly-supervised model in terms of classification and localization independently. We adopt mean average precision (mAP) in [36] to evaluate the video-level classification performance and AP at different IoU thresholds to evaluate the class-agnostic localization quality regardless of the action class. We report the video-level classification mAP in Table 5, showing only marginal gain as expected. This is because THUMOS14 only contains one or two action classes in a single video which makes the video be easily classified into the target action category. We also evaluates boundary detection AP regardless of the label in Table 5, showing large gain after adding single-frame supervision.

D Sensitivity Analysis

D.1 Background Ratio

Table 6 shows the results with respect to different background ratios η on THUMOS14. The mean and standard deviation of segment and frame metrics are

Table 6. The background η analysis on THUMOS14. AVG is the average mAP at IoU 0.1 to 0.7.

η	mAP@hit	mAP@IoU					
		0.1	0.3	0.5	0.6	0.7	AVG
0.0	44.4±0.56	58.6±0.55	41.1±0.80	20.2±0.69	12.9±0.58	7.3±0.10	31.7±0.47
1.0	57.7±0.41	68.3±0.37	51.1±0.57	28.2±0.52	17.7±0.09	9.4±0.31	39.3±0.13
3.0	60.6±1.36	71.0±1.21	53.8±0.71	29.3±1.14	18.9±0.88	9.4±0.43	41.1±0.80
5.0	60.6±0.85	70.6±0.92	53.7±1.21	29.1±0.39	19.1±1.31	10.2±0.84	41.1±0.78
7.0	60.9±0.56	70.7±0.08	54.3±1.18	29.5±0.13	19.0±0.50	10.1±0.27	41.3±0.44
9.0	60.2±1.12	70.3±0.83	53.4±0.8	29.6±0.58	18.8±0.99	10.1±0.37	41.0±0.60

Table 7. The loss coefficients analysis on THUMOS14. AVG is the average mAP at IoU 0.1 to 0.7.

parameter	mAP@hit	Segment mAP@IoU					
		0.1	0.3	0.5	0.6	0.7	AVG
$\alpha = 0.2$	61.9±0.34	71.6±0.73	54.2±1.31	29.3±0.47	18.4±0.62	9.7±0.35	41.3±0.56
$\alpha = 0.5$	61.9±0.68	71.8±0.36	54.4±0.68	30.2±0.41	19.3±0.92	10.2±1.14	41.9±0.47
$\alpha = 0.8$	60.7±0.95	71.0±0.40	53.8±0.64	29.4±0.26	19.0±0.23	10.0±0.25	41.2±0.22
$\beta = 0.2$	60.6±1.55	70.5±1.21	53.2±1.09	29.4±0.64	18.8±0.71	9.7±0.33	41.0±0.67
$\beta = 0.5$	60.2±0.69	70.5±0.55	53.7±0.71	29.4±0.16	18.8±0.47	10.0±0.34	41.1±0.42
$\beta = 0.8$	60.8±1.05	70.6±0.50	53.8±1.47	29.6±0.34	18.9±0.36	10.0±0.37	41.2±0.55

reported. We ran each experiment three times. The single-frame annotation for each video is randomly sampled from annotations by four annotators. From the table 6, we find that our proposed SF-Net boosts the segment and frame evaluation metrics on THUMOS14 dataset with background mining. The model becomes stable when the η is set in range from 3 to 9.

D.2 Loss coefficients

We also conduct experiments to analyze the hyper-parameters of each loss item on the THUMOS14 in Table 7. The mean and standard deviation of segment and frame metrics are reported. We ran each experiment three times. The single-frame annotation for each video is randomly sampled from annotations by four annotators. The default values of α and β are 1. We change one hyper-parameter and fix the other one. From the Table 7, we observe that our model is not sensitive to the hyper-parameters.

E ActivityNet Simulation Experiment

We conduct experiments on ActiviytNet1.2 by randomly sampling single-frame annotations from ground truth temporal boundaries. Table 8 presents the re-

Supervision	Method	mAP @IoU			
		0.5	0.7	0.9	AVG
Full	CDC [28]	45.3	-	-	23.8
Full	SSN [42]	41.3	30.4	13.2	28.3
Weak	UntrimmedNet [35]	7.4	3.9	1.2	3.6
Weak	AutoLoc [29]	27.3	17.5	6.8	16.0
Weak	W-TALC [10]	37.0	14.6	-	18.0
Weak	Liu *et al.* [19]	36.8	-	-	**22.4**
Weak	3C-Net [24]	**37.2**	**23.7**	**9.2**	21.7
Single-frame	SF-Net (**Ours**)	37.8	24.6	10.3	22.8

Table 8. Segment localization results on ActivityNet1.2 validation set. The AVG indicates the average mAP from IoU 0.5 to 0.95.

sults on ActivityNet1.2 validation set. In this experiment, the annotations are generated by randomly sampling single frame from ground truth segments. We follow the standard evaluation protocal [3] by reporting the mean mAP scores at different thresholds (0.5:0.05:0.95). On the large scale dataset, our proposed method can still obtain a performance gain with single frame supervision.

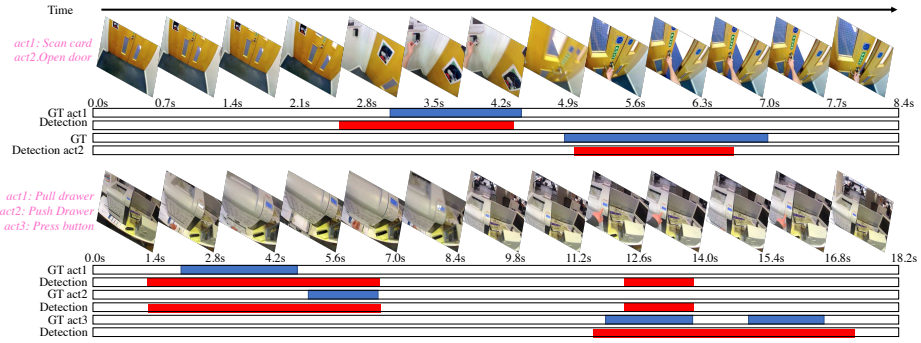

Fig. 6. Qualitative Results on BEOID dataset. GT denotes the ground truth and the action segment is marked with blue. Our proposed method detects all the action instances in the videos.

F Qualitative Results

We present the qualitative results on BEOID dataset in Figure 6. The first example has two action instances: *scan card* and *open door*. Our model localizes every action instance and classifies each action instance into the correct category.

The temporal boundary for each instance is also close to the ground-truth annotation despite that we do not have any temporal boundary information during training. For the second example, there are three different actions and total four action instances. Our SF-Net has detected all the positive instances in the videos. The drawback is that the number of detected segments for each action class is greater than the number of ground truth segments. To better distinguish actions of different classes, the model should encode the fine-grained action information from the target action area instead of the 1D feature directly extracted from the whole frame. We will consider this in the future work.

← Go to **ECCV 2020 Conference** homepage (/group?id=thecvf.com/ECCV/2020/Conference)

SF-Net: Single-Frame Supervision for Temporal Action Localization

(/pdf?id=7YCdhpXnZx)

Anonymous

06 Mar 2020 (modified: 17 Jul 2020) ECCV 2020 Conference Blind Submission Readers: Paper2314 Authors, Paper2314 Reviewers, Paper2314 Area Chairs, Program Chairs Show Revisions (/revisions?id=7YCdhpXnZx)

Abstract: In this paper, we study an intermediate form of supervision, i.e., single-frame supervision, for temporal action localization (TAL). To obtain the single-frame supervision, the annotators are asked to identify only a single frame within the temporal window of an action. This can significantly reduce the labor cost of obtaining full supervision which requires annotating the action boundary. Compared to the weak supervision that only annotates the video-level label, the single-frame supervision introduces extra temporal action signals while maintaining low annotation overhead. To make full use of such single-frame supervision, we propose a unified system called SF-Net. First, we propose to predict an actionness score for each video frame. Along with a typical category score, the actionness score can provide comprehensive information about the occurrence of a potential action and aid the temporal boundary refinement during inference. Second, we mine pseudo action and background frames based on the single-frame annotations. We identify pseudo action frames by adaptively expanding each annotated single frame to its nearby, contextual frames and we mine pseudo background frames from all the unannotated frames across multiple videos. Together with the ground-truth labeled frames, these pseudo-labeled frames are further used for training the classifier.

Subject Areas: Action Recognition, Understanding

Author Agreement: All authors agree with the author guidelines of ECCV 2020.

TPMS Agreement: All authors agree that the manuscript can be processed by TPMS for paper matching.

Supplementary Material: zip (/attachment?id=7YCdhpXnZx&name=supplementary_material)

Source: zip (/attachment?id=7YCdhpXnZx&name=source)

First Author Is A Student: Yes

Copyright: pdf (/attachment?id=7YCdhpXnZx&name=copyright)

Revealed to Fan Ma, Linchao Zhu, Yi Yang, Shengxin Zha, Gourab Kundu, Matt Feiszli, Zheng Shou

24 Feb 2020 (modified: 17 Jul 2020) ECCV 2020 Conference Submission

Authors: *Fan Ma (/profile?id=~Fan_Ma2), Linchao Zhu (/profile?id=~Linchao_Zhu1), Yi Yang (/profile?id=~Yi_Yang13), Shengxin Zha (/profile?id=~Shengxin_Zha2), Gourab Kundu (/profile?id=~Gourab_Kundu2), Matt Feiszli (/profile?id=~Matt_Feiszli1), Zheng Shou (/profile?id=~Zheng_Shou1)*

9 Replies Add **Withdraw**

Show all from everybody

[−] ## Final Decision

ECCV 2020 Conference Paper2314 Area Chairs, ECCV 2020 Conference Program Chairs

03 Jul 2020 ECCV 2020 Conference Paper2314 Decision Readers: Program Chairs, Paper2314 Area Chairs, Paper2314 Reviewers, Paper2314 Authors

Decision: Accept (Spotlight)

Comment: This paper received pretty positive reviews: 1 strong accept, 2 weak accepts and 1 borderline accept. Using single-frame supervision for action localization is an extension of point supervision for weakly supervised object detection, and novelty is not very impressive. However, the benefit of single-frame supervision in videos can be more significant than that of point supervision in images considering relative annotation costs with respect to full supervision. So, the AC panel recommends accepting this paper.

[−] ## Official Review of Paper2314 by AnonReviewer2

ECCV 2020 Conference Paper2314 AnonReviewer2

22 May 2020 (modified: 03 Jul 2020) ECCV 2020 Conference Paper2314 Official Review Readers: Program Chairs, Paper2314 Area Chairs, Paper2314 Reviewers, Paper2314 Authors

Summary Of Contributions: This paper proposes a new task, single-frame supervision for temporal action localization. The single-frame annotation speed is similar to the video-level class annotation, but the performance can be much better than the latter one. To make use of the single-frame annotation, this paper proposes a method to mine pseudo action and background frames based on the annotated single action frame.

Strengths: The proposed new task is interest and worthwhile for the community. The experimental results also show a better performance than the weakly-supervised temporal action localization with only video-level action labels.

Weaknesses: At the inference, why use the sum of classification score and actionness score but not the product?

In Table 2, I don't understand why the "TS in GT" performs much worse compared with "Annotation".

In Table 3, it would be better to also show the SFE and SFBE.

Suggestion To Authors: Perhaps you can write the annotation simulation first and then the annotation by annotators. It is very straightforward for a reader to think about generate the annotations with the ground truth boundary annotations.

Preliminary Rating: 4: Borderline accept

Preliminary Rating Justification: This paper proposes a new task that only using single-frame annotation for each action to do the temporal action localization. The annotation process is practical in real-world applications, and I think it is worthwhile for the community.

Confidence: 4: High, published similar work

Final Rating: 5: Weak accept

Final Rating Justification: I appreciate the idea of using a single frame annotation for temporal action localization. It provides new insight into the vision research community and also practical usage in the industry. Moreover, the authors' response is quite satisfactory. The discussion of Logit Sum, Softmax Sum, Softmax product is great and should definitely be added to the final version. I also do hope the author will address all the questions raised by all the reviewers in the final version.

BTW, "Marginalized Average Attentional Network for Weakly-Supervised Learning" (ICLR 2019) should be cited and compared.

[−] Rebuttal by Paper2314 Authors

ECCV 2020 Conference Paper2314 Authors *Fan Ma (/profile?id=~Fan_Ma2) (privately revealed to you)*

29 May 2020 (modified: 30 May 2020) ECCV 2020 Conference Paper2314 AnonReviewer2 Rebuttal Readers: Program Chairs, Paper2314 Area Chairs, Paper2314 Reviewers, Paper2314 Authors

Rebuttal:

We appreciate R2 for the recognition of this paper and the valuable suggestions. We respond to the questions raised by R2 in the following.

(1) Why use the sum of classification score and actionness score but not the product?

Response: Note that for the classification score C and actionness score A, they are logits which are before softmax and thus could be positive or negative or 0. Also, higher the score, higher probability of being the corresponding class. When using “sum”, this monotonicity still holds for C+A. But when using “product” C*A, this monotonicity does not hold (e.g. consider -1×-2=2, -1×1=-1, 1×1=1)

Furthermore, we have experimented with setting the scores to be values after softmax and in this case we explored various choices including sum, product, etc. while all of them underperform the chosen strategy in the paper (i.e. sum of logits before softmax). We will include such discussion in the final paper.

Method	Logit Sum	Softmax Sum	Softmax product
mAP@IoU=0.1	70.7	63.4	59.6
mAP@IoU=0.5	29.3	21.1	13.6

(2) In Table 2, why the “TS in GT” performs much worse compared with “Annotation”.

Response: The sampling strategy “TS in GT” is proposed in [20]: the annotated frame is sampled from a normal distribution with the mean at the center timestamp and the standard deviation set to 1 second. Therefore, most annotated frames are located around the action instance center, providing limited information about how the action looks in other temporal parts of action (e.g. beginning part). But for “Annotation”, the annotated frames are distributed much more diversely over the whole action duration as shown in Fig. 4. Therefore, “TS in GT” performs much worse compared with “Annotation”.

(3) In Table 3, it would be better to also show the SFE and SFBE.

Response: We have conducted the experiment on THUMOS14 for one run and the result is shown as follows.We will put more results in the final manuscript.

Method	0.1	0.3	0.5	0.7	Avg(0.1:0.7)
SFE	58.4	41.2	19.8	7.4	31.4
SFBE	63.3	47.5	25.1	8.4	36.3

[−] Official Review of Paper2314 by AnonReviewer4

ECCV 2020 Conference Paper2314 AnonReviewer4

21 May 2020 (modified: 03 Jul 2020) ECCV 2020 Conference Paper2314 Official Review Readers: Program Chairs, Paper2314 Area Chairs, Paper2314 Reviewers, Paper2314 Authors

Summary Of Contributions: This paper proposes a novel temporal action localization(TAL) framework where it only receives a single-frame annotation. To be specific, the authors propose algorithms to aggregate pseudo ground truth frames from the single-frame supervision by 1) expanding single-frame annotation to neighbor frames, 2) mining background-likely frames.

For the experiments, this paper provides an analysis of annotation cost for the video, which is not explored much. The proposed method achieves comparable performance to fully-supervised methods. Moreover, the paper proposes a novel task, called single frame localization, which aims to localize a single frame from action instance.

Strengths:

- The novelty of task:

This paper proposes a novel problem setting for temporal action localization, where only the single-frame annotation is given and a novel task, which is localizing a single frame of action. Both of the proposed tasks seem practical in that the performance is similar to the fully-supervised methods while the annotation cost is much lower.

- Analysis of annotation cost for video

The authors carefully design the annotation tool and the annotation process. Moreover, the paper provides an analysis of the annotation cost by comparing full-annotation, weak-annotation, and single-frame annotation, which is not explored much so far.

Weaknesses:

- Architecture:

The actionness module seems similar to the attention module proposed by many other weakly supervised TAL methods[1,2,3,4] with slight modification. I think it is not fair to say that the authors designed the module motivated by proposal based methods as stated in line 304.

- Algorithms:

Even though the background mining algorithm, where the model simply takes top-k frames, seems well suited for the proposed framework, it is hard to say that it is a “novel algorithm”. Many other weakly supervised TAL methods take this strategy for taking action instances[5,6,7,8]. The authors only adopt this strategy to background class.

- Experiments :
1) I'm not sure whether the single frame supervision could still show comparable performance to fully-supervised frameworks for the large scale datasets, where the given dataset has lots of similar classes. 3 benchmarks(THUMOS'14, GTEA, BEOID) are small datasets to argue the generality of the proposed framework. Most of the action localization researches[1-8] conduct their experiments on activityNet, which has many more action categories(100 for 1.2 version/200 for 1.3 version), and has more similar action classes within the datasets. I'm aware that annotating the whole datasets might burdensome. However, since the datasets already have the full annotation, the experiments could be conducted by "single-frame simulation", as the authors did in Table 4.

2) Analysis of the single-frame localization task should be more elaborated. The paper only contains the performance(mAP@hit) of the proposed method. The comparison with other methods(fully or weakly supervised methods) should be considered.

3) I wonder why the performance of the proposed method for THUMOS'14 reported in Table 4 is different from Table 3.

4) Compare to the modern full supervision methods in Table 4, the fully-supervised version of the proposed method used in Table 3 seems pretty works well on THUMOS'14. I think the result is weird since the fully-supervised version of the proposed method is mere frame-wise classification. I've never seen the researches that frame-wise classification is comparable to the proposal-based methods for fully-supervised TAL. Please elaborate more on the fully-supervised version of the proposed method. Moreover, the recent fully-supervised methods[9,10] should be compared together in Table 4.

Refs:
[1]: Nguyen et. al, Weakly Supervised Action Localization by Sparse Temporal Pooling Network, CVPR, 2018
[2]: Liu et. al, Completeness Modeling and Context Separation for Weakly Supervised Temporal Action Localization, CVPR, 2019
[3]: Liu et. al, Weakly Supervised Temporal Action Localization through Contrast based Evaluation Networks, ICCV, 2019
[4]: Nguyen et. al, Weakly-supervised Action Localization with Background Modeling, ICCV, 2019
[5]: Wang et. al, UntrimmedNets for Weakly Supervised Action Recognition and Detection, CVPR, 2017
[6]: Paul et. al, W-TALC: Weakly-supervised Temporal Activity Localization and Classification, ECCV, 2018
[7]: Narayan et. al, 3C-Net: Category Count and Center Loss for Weakly-Supervised Action Localization, ICCV 2019
[8]: Lee et. al, Background Suppression Network for Weakly-supervised Temporal Action Localization, AAAI, 2020
[9]: Zheng et. al, Graph Convolutional Networks for Temporal Action Localization, ICCV, 2019
[10]: Lin et. al, BMN: Boundary-Matching Network for Temporal Action Proposal Generation, ICCV, 2019

Suggestion To Authors:
- The paper [11] should be cited or mentioned in related works. The paper handles the spatio-temporal action localization with various supervision including a single timestamp.
- Typo: In Figure 2 in supp. material, the position of the axis name is miss-placed(GT act2 Detection).
- For the qualitative results, it might be better if we could see how the initial supervision extends to the frames nearby gradually.
- I wonder if the annotation tool or the annotations will be released to the public.
- I wonder why the line 7~9 in Algorithm 1(supp. material) states "argmin", not argmax.

Refs:
[11]: Chéron et. al, A flexible model for training action localization with varying levels of supervision, NeurIPS, 2018

Preliminary Rating: 3: Borderline reject

Preliminary Rating Justification: The proposed setting(single-frame supervision) and the proposed task(single-frame localization) are novel and seem practical in the sense of high performance and low annotation cost. However, I have some concerns about the proposed framework and the experiments as I mentioned in the weakness section. Therefore, I rate the paper as borderline reject for the preliminary rating.

Confidence: 2: Low, read similar work

Final Rating: 4: Borderline accept

Final Rating Justification: Basically, the idea of using a single frame only to detect action in the video is practically useful in the view of annotation cost.
Moreover, the authors addressed all of my concerns and their response is quite satisfactory.
Although I still have some doubts about the novelty of the mining algorithm itself, the authors' claim that they mine the pseudo-background frames for the first time makes sense to some extent.
And the quality of this research is above the borderline in overall.
Therefore, I change my rating to acceptance.

[−] ## Rebuttal by Paper2314 Authors

ECCV 2020 Conference Paper2314 Authors *Mike Zheng Shou (/profile?id=~Mike_Zheng_Shou1) (privately revealed to you)*

29 May 2020 (modified: 30 May 2020) ECCV 2020 Conference Paper2314 AnonReviewer4 Rebuttal Readers: Program Chairs, Paper2314 Area Chairs, Paper2314 Reviewers, Paper2314 Authors

Rebuttal:
Thanks for acknowledging our novelty and providing useful suggestions. We address your concerns as follows.

Q1: The architecture of the attention module in [1-4] is similar to our actionness module.. motivated.. line 304.. A: Indeed the network architectures are similar and we will add [1-4] in line 304. But note that the purpose of introducing and the way of utilizing such an actionness module are totally different. [1-4] used the output of attention module to weighted sum the frame-level classification scores to obtain the video-level label. In our actionness module, its output is exactly what we need and is directly used for comparing with the frame-level label during training which uniquely exists in our task.

Q2: the novelty of the background mining algorithm. A: (1) We are the first to propose the idea of mining pseudo background frames. This makes it possible to introduce and train a background class in the classifier to improve its discriminability. [5-8] do not mine pseudo frames. (2) during training, we alternate between choosing pseudo background frames and training a frame-level classifier to boost each other gradually. [5-8] do not train a frame-level classifier let alone the boosting process. (3) When setting k in [5-8], k is proportional to the number of all frames; to fit our task, we propose to set k proportional to the number of action instances, i.e. labeled frames (line 282).

Q3: ActivityNet. A: See Q4 to Reviewer3.

Q4: For the single-frame localization task, add results for fully and weakly.. A: Sure, we will add them. When using the same backbone, the results are as follows:

Dataset	Weakly	Fully	Single-Frame
THUMOS14	39.5	55.2	60.2
BEOID	7.8	34.3	47.4
GTEA	7.2	29.2	34.5

Q5: Why ... Table 4 is different from Table 3. A: In Table 3, we conduct five runs and report average. But in Table 4 when comparing with SOTA methods, we follow them to only report one run.

Q6: Why is the fully-supervised version of the proposed method comparable to the proposal-based methods for fully-supervised TAL? A: When IoU is high, our fully-supervised version is worse than proposal-based fully-supervised methods as expected. When IoU is low, they are comparable because the detected segment can be regarded as correct even when the boundary is not precise.

Sure, we will add [9,10] in Table 4.

Q7: Suggestion To Authors. A: Thank you for your suggestions. We will also revise accordingly. We will release the annotation files and tools.

[−] **Official Review of Paper2314 by AnonReviewer3**

ECCV 2020 Conference Paper2314 AnonReviewer3

13 May 2020 (modified: 03 Jul 2020) ECCV 2020 Conference Paper2314 Official Review Readers: Program Chairs, Paper2314 Area Chairs, Paper2314 Reviewers, Paper2314 Authors

Summary Of Contributions: The paper address the gap between weakly-supervised and fully-supervised action localization by studying the effect of single-frame supervision. The paper demonstrates the effectiveness of this approach on three different benchmarks with detailed explanation or analyses.

Strengths: -- The writing is clear and easy to understand.

-- The thesis and the story are well-founded and clearly stated.
The paper did well on laying out the practical short-comings of current fully-supervised and weakly-supervised approach. They even go further to analyze the single-frame annotations. This is greatly appreciated. This alone is a wonder contribution to the community. Figure 1 lays out perfectly the pain points of the current settings.

-- A lot of papers say that temporal boundaries are expensive and weakly-supervision are cheaper, but never go into details. This paper explore in details and more.

-- The value of annotation analysis, section 4.4.
This section is excellent. They offer rare information and experience that are not available in any of the recent publication in the field of action localization. By studying the behaviors of annotators for single-frame annotations and comparing the annotation speed against under fully-supervision, it gives specific insights into the difficulties of data annotations.

Weaknesses: -- Four annotators would be a bit biased.
Have you tried the annotation with MTurk and exposed this issue to a larger set of annotators? I am concerned about the noisiness given a small pool of annotators.

-- What happened to the annotation when the action already passed? Hence the frame is outside of the action?

-- annotation time for single-frame label is close to the annotation time for video-level label
This is true for the case of 1-min video. However, I doubt this statement holds for much longer video (10-20 min videos). Unless the authors can provide further evidence for different settings, longer videos with multiple actions, these kind of statements can be misleading.

-- The lack of performance benchmarking on larger-scale datasets.
It would be interesting to know how single-frame supervision approach is validated on larger benchmark such as ActivityNet 1.2 or 1.3. The datasets in the paper are relatively small. For the larger datasets, the single-frame supervision simulation would suffice.

-- Higher IoU sees less improvement for the lower IoU.
This seems like the methods are able to localize more actions, but however the action boundaries are still not well-located.

Suggestion To Authors: Have you tried regression the actual boundaries from the given single-frame annotations? Even with the new annotations, your model still resembles much of the weakly-supervised approach. It'd be interesting to see how the learning process and inference change with the new type of information, maybe better boundary refinement, or you can grow the action from the seed frames. Something similar to tracking for 2D objects.

Preliminary Rating: 6: Strong accept

Preliminary Rating Justification: I recommend a strong accept. The writing is clear and easy to understand. The paper offers a lot of values to the field of action localization. The paper addresses multiple practical pain points dealing with action annotations. They thoroughly explore a third option in this field. The motivation is solid and the execution is great. This paper provides a lot of information for future research and offers a lot of new knowledge to the field. The paper also has great practical implications for people who want to implement and deploy action localization system. This paper would be a great addition to the community.

However, there are still some shortcomings listed above. However I am also leaning toward a strong accept for this paper given the impact of the contributions.

Confidence: 4: High, published similar work

Final Rating: 6: Strong accept

Final Rating Justification: After reading the feedback from the author and other reviewers, I decide to keep my rating as a strong accept. This paper is a wonderful addition to the action localization literature. The paper stresses the current pain points and problems of the problem in term of practical

applicability. They actually perform actual annotations to provide useful reference point and data for future works or anyone who really wants to implement a real-world action localization system.

I do hope the authors expand on their current works on the annotation experiments and address other reviewers' concerns in the camera-ready version.

[−] **Rebuttal by Paper2314 Authors**

ECCV 2020 Conference Paper2314 Authors *Mike Zheng Shou (/profile?id=~Mike_Zheng_Shou1) (privately revealed to you)*

29 May 2020 (modified: 30 May 2020) ECCV 2020 Conference Paper2314 AnonReviewer3 Rebuttal Readers: Program Chairs, Paper2314 Area Chairs, Paper2314 Reviewers, Paper2314 Authors

Rebuttal:

We appreciate your recognition and the insightful suggestions. To each question in weakness:

Q1: Four annotators..a bit biased. A: To mitigate the bias, these four annotators are of different backgrounds: 2 males and 2 females who have different careers. Ideally without budget limit, it would be better to have more annotators to make the annotations more diverse and our model can achieve even better performances.

For the TS simulation in Table 2 (get ground truth frames by uniform sampling) -- this can be analogical to the extreme case that the annotation results are very diverse and have no bias about the annotated frame position. This TS simulation gives comparable performance to using our current annotations from 4 people. Thus ours shall have negligible bias.

Q2: What happened to the annotation when the action already passed? A: Yes. Fig. 4 x-axis also has one bin for “outside” whose percentage is small. Also, we find that these outside frames are very visually similar to the preceding action frames.

Q3: annotation time...close... True for 1-min videos but how about long videos (10-20mins)? A: Thank you for raising this insightful discussion. We denote that given a video, t_sf: annotation time of single-frame supervision, t_w: annotation time of video-level label for one video.

The annotation overhead is (t_sf - t_w), which mainly comes from “pause and annotate the action frame for each instance” and thus is proportional to the number of action instances contained in the whole video. This is true not only for short video but also long video, because we still have to watch the whole video to decide the video-level label. When the number of instances in a long video is small, the annotation overhead (t_sf - t_w) is negligible compared to t_w for watching the whole video.

So whether (t_sf - t_w) is negligible really depends whether the video contains a lot of instances. On THUMOS14, the average number of instances per video is 15. Some videos indeed are over 10 mins and we find their (t_sf - t_w) is actually close to t_weakly. We will add such discussions in the final paper.

Q4: larger-scale datasets. A: Per requested, we simulation labeled frames by uniformly sampling from the GT instances on ActivityNet1.2 as in the following. Our SF-Net can still outperform weakly-supervised methods.

Methods	0.5	0.7	0.9	Avg(0.5:0.95)
AutoLoc	27.3	17.5	6.8	16.0
W-TALC	37.0	14.6	-	18.0
3CNet	37.2	23.7	9.2	21.7
SF-Net	37.8	24.6	10.3	22.8

[−] **Official Review of Paper2314 by AnonReviewer1**

ECCV 2020 Conference Paper2314 AnonReviewer1

11 May 2020 (modified: 03 Jul 2020) ECCV 2020 Conference Paper2314 Official Review Readers: Program Chairs, Paper2314 Area Chairs, Paper2314 Reviewers, Paper2314 Authors

Summary Of Contributions: This paper proposes a temporal action location method based on single-frame supervision. Their method mines pseudo action and background frames based on single-frame annotations. These pseudo-labeled frames are used for training the classifier. The experimental results on three public datasets are good and compared with other approaches.

Strengths: (1) The paper is well written and easy to understand.
(2) The pseudo mining and training objectives are reasonable.
(3) The experimental results are good.
(4) The annotations of three datasets should be interesting for the vision community.

Weaknesses: (1) About the background frame mining, it would be better if the authors can explain more why their proposed method would work.
(2) During the inference stage in section 3.4, the action segment detection is not clear. Not sure whether the authors use the sliding windows as the initial segment candidates.
(3) The explanations of some variables or parameters are not clear. For example, at line 272, about the variable \xi, how to set this parameter? The parameter \eta at line 363 and line 279 should be different. The authors should rename one of them.

Suggestion To Authors: The authors can improve the paper based on comments in the weakness section. I will suggest the authors release their single-frame annotations for three public datasets.

Preliminary Rating: 4: Borderline accept

Preliminary Rating Justification: The topic of single-frame supervision is interesting. The proposed method is evaluated on three public datasets and the results are good. The main concern is the background frame mining method. Not sure whether it will work and the authors can explain more in the rebuttal.

Confidence: 3: Medium, published weakly related work

Final Rating: 4: Borderline accept

Final Rating Justification: The authors have responded to my questions in the reviews. This paper can be accepted.

[−] **Rebuttal by Paper2314 Authors**

ECCV 2020 Conference Paper2314 Authors Fan Ma (/profile?id=~Fan_Ma2) (privately revealed to you)

29 May 2020 (modified: 30 May 2020) ECCV 2020 Conference Paper2314 AnonReviewer1 Rebuttal Readers: Program Chairs, Paper2314 Area Chairs, Paper2314 Reviewers, Paper2314 Authors

Rebuttal:

Thank you for the helpful suggestions and we will revise the paper accordingly.

(1) About the background frame mining, it would be better if the authors can explain more why their proposed method would work.

Response: (a) At the beginning, we do not have supervision about where the background frames are. But explicitly introducing a background class can avoid forcing classifying a frame into one of the action classes. Our proposed background frame mining algorithm can offer us the supervision needed for training such a background class so as to improve the discriminability of the classifier. (b) During each iteration, our proposed algorithm selects these frames of high background class scores as pseudo ground truth. Despite these frames having high scores, often the scores are not as high as 1; after training with pseudo ground truth, these frames will be likely to have higher background class scores (getting closer to 1) because the classifier's discriminability gets improved. (c) At the beginning of training, the selected ground truth background frames would have some noises/mistakes; as the training evolves and the classifier's discriminability improves, we are able to reduce the noises and detect background frames more correctly; with more correct background frames as supervision signals, the classifier's discriminability can be further boosted. (d) Ablation study in Table 3 also demonstrates that our background frame mining indeed improves the model performance. We will add the above discussions in the final paper.

(2) During the inference stage in section 3.4, the action segment detection is not clear. Not sure whether the authors use the sliding windows as the initial segment candidates.

Response: Thank you for pointing this out and we will make it clear in the final version. To detect segments, we do not use sliding windows. We simply follow the thresholding strategy like in [21, 23] to keep the action frames above the threshold and consecutive action frames constitute an action segment.

(3) The explanations of some variables or parameters.

Response: According to supplementary material line 72, we set ξ at 0.9 in all experiments. Yes, η should be different. We will rename one of them in the final revision. Thank you for pointing it out.

Suggestions: I will suggest the authors release their single-frame annotations for three public datasets.

Response: For sure we will release the annotations for all three datasets.