Weakly-Supervised Audio-Visual Video Parsing with Prototype-based Pseudo-Labeling

Kranthi Kumar Rachavarapu
IIT Madras
kranthi.rachavarapu@gmail.com

Kalyan Ramakrishnan
University of Oxford
kalyanr@robots.ox.ac.uk

Rajagopalan A. N.
IIT Madras
raju@ee.iitm.ac.in

Abstract

In this paper, we address the weakly-supervised Audio-Visual Video Parsing (AVVP) problem, which aims at labeling events in a video as audible, visible, or both, and temporally localizing and classifying them into known categories. This is challenging since we only have access to video-level (weak) event labels when training but need to predict event labels at the segment (frame) level at test time. Recent methods employ multiple-instance learning (MIL) techniques that tend to focus solely on the most discriminative segments, resulting in frequent misclassifications. Our idea is to first construct several "prototype" features for each event class by clustering key segments identified for the event in the training data. We then assign pseudo labels to all training segments based on their feature similarities with these prototypes and re-train the model under weak and strong supervision. We facilitate this by structuring the feature space with contrastive learning using pseudo labels. Experiments show that we outperform existing methods for weakly-supervised AVVP. We also show that learning with weak and iteratively re-estimated pseudo labels can be interpreted as an expectation-maximization (EM) algorithm, providing further insight for our training procedure.

1. Introduction

In this paper, we explore the problem of weakly-supervised Audio-Visual Video Parsing (AVVP) [48], where the goal is to classify events in a video and localize them over time *and* modalities, given only video-level (*weak*) event labels for training. Such a formulation is attractive since it forgoes the need for expensive and tedious fine-grained labeling. Moreover, other weakly-supervised video-based temporal prediction tasks can be posed as special cases of AVVP. For instance, Audio-Visual Event Localization (AVEL) [47] considers events that are simultaneously audible and visible, whereas Temporal Action Localization (TAL) [10] considers only visible events. AVVP is challenging as an event could occur in just the audio stream, visual stream, or both at the same time. Audio-visual data is ubiquitous, with highly variable events in terms of appearance and duration.

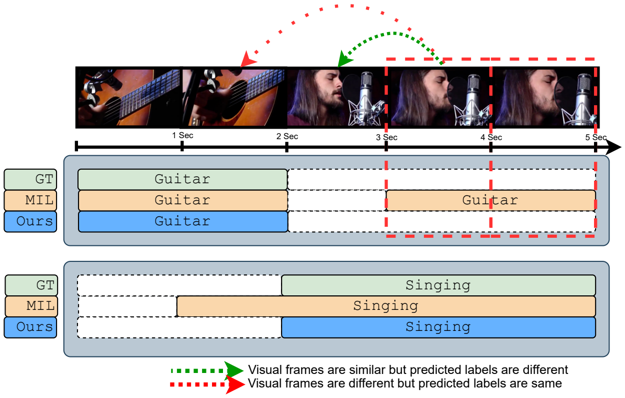
1 Sec
2 Sec
3 Sec
4 Sec
5 Sec
GT
MIL
Ours
Guitar
Guitar
Guitar
Guitar
GT
MIL
Ours
Singing
Singing
Singing
Visual frames are similar but predicted labels are different
Visual frames are different but predicted labels are same

Figure 1. Example predictions of a model trained using MIL under weak supervision [48] and our method. For clarity, we only show visual events. The MIL model detects the event "Guitar" correctly in the beginning, but there are some mistakes (red dotted boxes) later on. Moreover, although the last three frames are similar, they are labeled inconsistently by the model. Our method consistently localizes well. "GT" is the ground truth.

Recent methods for weakly-supervised AVVP [30, 48] use multiple-instance learning (MIL) [7] techniques that optimize for video-level labels during training and make frame-level predictions during evaluation. Here, video-level predictions are obtained by *pooling* (aggregating) the frame-level predictions. A drawback of MIL approaches is that a model may learn to focus solely on the *most discriminative* ("prominent") instances (here, audio/visual frames in a video) and still minimize the training loss successfully. This means such models may not reliably detect the full temporal extent of events. Fig. 1 shows an example where an MIL model [48] for AVVP fails to localize events accurately. Moreover, the model labels frames that are visually similar differently. Another reason for misclassifications in MIL models is that they typically contain a frame-level classifier with a single weight vector per event class. However, this may be insufficient to capture the intra-class variation for the event across time, modalities, and examples while training with weak labels.

We propose to address these issues with the following ideas. First, given that the MIL model can find discriminative frames that trigger the video-level labels, we can identify such frames and propagate their predicted labels to other frames in the training data based on similarities in the feature space, thus creating frame-level *pseudo* labels on the training data. Second, we can use their features to construct *multiple prototypes* per event class, ensuring a more robust representation to propagate pseudo labels with. Once this is done for the training data, we can re-train the model with strong supervision from pseudo labels and weak supervision from video-level labels. Fig. 1 shows an example where our method produces more accurate predictions than the MIL model. Because our method relies on feature similarities, we further enforce structure in the feature space via contrastive learning using pseudo labels.

We re-estimate pseudo labels on the training data after every epoch and train with them over the next epoch. We show that such a training procedure can be viewed through an expectation-maximization (EM) lens with the unknown frame-level labels as latent variables. This interpretation provides convergence guarantees and further insight. Our method achieves state-of-the-art performance for weakly-supervised AVVP and also works well on the related weakly-supervised Temporal Action Localization (TAL) task. We summarize our contributions below:

1. We introduce a prototype-based pseudo-labeling method for weakly-supervised AVVP that exploits similarities in feature space.
2. We show how to perform contrastive learning using pseudo labels to help semantically structure the feature space.
3. We present an expectation-maximization view of our general training procedure.
4. We show state-of-the-art results for weakly-supervised AVVP, and on several metrics for weakly-supervised TAL.

2. Related Work

Audio-Visual Video Parsing (AVVP). Tian et al. [48] formulated AVVP as a multimodal multiple-instance learning (MMIL) problem and proposed a hybrid attention network with a learned pooling function (HAN) to capture unimodal and cross-modal contexts. Most subsequent work has used the HAN architecture with other techniques to improve performance. Lamba et al. [30] utilized cross-modal information to learn better representations with adversarial and self-supervised losses. Wu et al. [52] estimated modality-level labels by exchanging the audio/visual streams between pairs of videos, followed by retraining under stronger supervision. Lin et al. [32] used cross-video and cross-modality signals, such as event co-occurrence across modalities, as additional sources of supervision. Mo et al. [36] leveraged multi-modal grouping to learn discriminative subspaces. Gao et al. [11] extract the "presence" and "absence" evidence for events from uni- and cross-modal information. While Fu et al. [9] focused on learning rate imbalance between modalities and Zhou et al. [60] utilized CLIP [41] to get segment-labels, our work leverages in-model feature similarities for pseudo labeling. PoiBin [40] modeled the number of positive frames for an event as a latent variable to improve localization. Different from previous methods, we estimate segment-level pseudo labels and re-train under full supervision.

Pseudo-Labeling refers to estimating labels for unlabeled data using the predictions of a trained model. This has been used to improve performance on several weakly-supervised tasks including object detection [1, 46, 59] and image classification [2, 13, 21]. A few works [34, 39, 58] have employed pseudo-labeling to improve performance on Temporal Action Localization (TAL), generating labels from model outputs or attention weights. Luo et al. [34] posed the TAL task as expectation-maximization, using pseudo-labels to approximate the E and M steps. Contrary to these, we propose a pseudo-labeling strategy using feature similarities between segments guided by the model predictions and weak labels for a video.

Prototype-based Classification. Prototype learning with a nearest-neighbor (non-parametric) classifier has inspired various machine learning techniques [5, 12, 15, 43, 44]. Recently, there has been increased interest in integrating this into deep learning methods, e.g., zero-shot [25], few-shot [45], unsupervised [54, 55], and supervised [16, 35, 53, 56] learning. Prototype learning has also been widely used for image segmentation [8, 50, 51, 61]. Recently, prototype-based methods have also been used for *semi-supervised* image [37] and text [57] classification. A few works have used prototype-based methods with multiple-instance learning. E.g., Rymarczyk et al. [42] processed large medical images as a bag of patches, and Huang et al. [22] learned prototypes for the TAL task with graph convolutional networks. These works adopt an *embedding-level* MIL approach ([24]) with one prototype per class, while we adopt *instance-level* MIL and construct multiple prototypes per event class.

3. Background

3.1. Problem Definition

In the audio-visual video parsing (AVVP) problem, we aim to identify audible or visible events in an unconstrained video and simultaneously localize them in time. Specifically, given a video x consisting T non-overlapping temporal segments $\{x_t = (x_t^a, x_t^v)\}_{t=1}^{T}$ of the audio (A) and visual (V) streams, our objective is to classify each segment into C possible events (e.g., singing, vehicle moving, etc.). At test time, we need to predict segment-level event labels $y_t = (\mathbf{y}_\mathbf{t}^\mathbf{a}, \mathbf{y}_\mathbf{t}^\mathbf{v})$ for each audio and visual segment, where $\mathbf{y}_\mathbf{t}^\mathbf{a}, \mathbf{y}_\mathbf{t}^\mathbf{v} \in \{0,1\}^C$ are segment-level audio and visual event labels, respectively.

Weak Supervision. In weakly-supervised AVVP, for each video x, we only have access to the corresponding video-level event label vector $\mathbf{w} = (w_1, \ldots, w_C) \in \{0,1\}^C$, where $w_c = 1$ if *any* of the segments in the video contains the c-th event, and $w_c = 0$ otherwise. Such *weak, video-level* labels only indicate whether an event occurs in the video or not. During evaluation, we still need to predict *segment-level* labels $\{(\mathbf{y}_\mathbf{t}^\mathbf{a}, \mathbf{y}_\mathbf{t}^\mathbf{v})\}_{t=1}^T$. Note that multiple events can simultaneously occur in a video, i.e., $\sum_c w_c \geq 1$, and each segment could have more than one event.

3.2. MIL Framework

We adopt an *instance-level* MIL approach [24] as done by prior work [30, 32, 36, 48, 52]: First, each instance (here, an audio/visual segment) is assigned a classification score (here, a C-dimensional vector). Then, the scores are aggregated by a *pooling* operation, and the aggregate is used in the loss. In our case, this looks like

$$\left[\{\mathbf{f}_\mathbf{t}^\mathbf{a}\}_{t=1}^T, \{\mathbf{f}_\mathbf{t}^\mathbf{v}\}_{t=1}^T\right] = f_\theta(x) \tag{1}$$

$$\mathbf{p}_\mathbf{t}^\mathbf{m} = g_\phi(\mathbf{f}_\mathbf{t}^\mathbf{m}) \in [0,1]^C, \ \ \forall t \in [1,T], \ m \in \{a, v\} \tag{2}$$

$$\hat{\mathbf{w}} = \sigma_{\theta,x}\left(\{\mathbf{p}_\mathbf{t}^\mathbf{a}\}_{t=1}^T, \{\mathbf{p}_\mathbf{t}^\mathbf{v}\}_{t=1}^T\right) \in [0,1]^C, \tag{3}$$

where f_θ is a feature extractor (encoder), g_ϕ is a feature classifier, $\sigma_{\theta,x}$ is a MIL-pooling operator. The predicted video-level probability vector $\hat{\mathbf{w}}$ is computed from segment-level classification scores $\{\mathbf{p}_t^m\}$, which are computed from segment features $\{\mathbf{f}_\mathbf{t}^\mathbf{m}\}$. We adopt the Hybrid Attention Network (HAN) [48] architecture to implement f_θ, g_ϕ, and σ. Here, f_θ consists of a transformer-based encoder with cross-modal attention and g_ϕ is a linear classifier with the sigmoid activation. Attention-based averaging is used for σ. We provide implementation details in the Supplementary.

This model is trained end-to-end with supervision at the video level by minimizing a weighted cross-entropy loss

$$\mathcal{L}_{\text{MIL}} = \text{CE}(\hat{\mathbf{w}}, \mathbf{w}) \tag{4}$$

with weights inversely proportional to the event distribution in the dataset. At test time, we get segment-level predictions from Eq. (2).

4. Prototype-based Pseudo-Labeling

While instance-level MIL enables finding key segments that trigger the video-level labels [24, 33], they often miss out on the full temporal extent of events in a video. However, if we identify such discriminative segments for each label in a training video, we could use them to build a set of *prototype* features that represent each class. A high degree of similarity of a segment with the prototype features for an event class means the segment probably contains the event. We use this idea to generate pseudo labels for all segments in a training video and then re-train the model under strong supervision. Fig. 2 shows an overview of our approach. We provide pseudocode in Algorithm 1.

4.1. Constructing Prototype Features

The segment-level linear classifier g_ϕ (Eq. (2)) can itself be considered a *prototype-based* classifier, where the prototype for each class is the corresponding weight vector in ϕ. However, this requires that an event have similar representations across modalities and examples in the training data. We instead construct multiple prototypes for each class in the following two steps.

Segment Selection. For each training video x, we find *positive* segments $\mathcal{P}_x(c)$ for a target event c (if $w_c = 1$) as those the model is highly confident contain the event. I.e.,

$$\mathcal{P}_x(c) = \begin{cases} \{x_t^m \in x | \mathbf{p}_\mathbf{t}^\mathbf{m}(c) \geq \gamma p^*(c)\} & \text{if } w_c = 1 \\ \phi & \text{if } w_c = 0, \end{cases} \tag{5}$$

where $\gamma \in (0,1]$ is a hyperparameter and $p^*(c) = \max_{t,m} \mathbf{p}_\mathbf{t}^\mathbf{m}(c)$ is the maximum predicted probability for the event over time *and* modalities for a given video. Clearly, $\mathcal{P}_X(c)$ will contain at least one audio/visual segment from video x when $w_c = 1$. We assume these segments contain event c with high probability (so are likely true positives). We also find *negative* segments $\mathcal{N}_x(c)$ for a target event c as those the model is highly confident do not contain the event when $w_c = 1$ (likely true negatives), or is highly confident do contain the event when $w_c = 0$ (definite false positives):

$$\mathcal{N}_x(c) = \begin{cases} \{x_t^m \in x \mid \mathbf{p}_\mathbf{t}^\mathbf{m}(c) \leq \beta p^*(c)\} & \text{if } w_c = 1 \\ \{x_t^m \in x \mid \mathbf{p}_\mathbf{t}^\mathbf{m}(c) = p^*(c) >= 0.5\} & \text{if } w_c = 0, \end{cases} \tag{6}$$

where $\beta \in (0,1)$ is a hyperparameter set to a small value to retrieve segments without the event c (if any). Note that $\mathcal{N}_x(c)$ may contain no segments at all even when $w_c = 1$, i.e., be a null set.

Finally, we aggregate the positive and negative sets over all videos in training data $\mathcal{D}$ to get prototypical segments as

$$\mathcal{P}(c) = \bigcup_{x \in \mathcal{D}} \mathcal{P}_x(c), \quad \mathcal{N}(c) = \bigcup_{x \in \mathcal{D}} \mathcal{N}_x(c). \tag{7}$$

Prototype Generation. We now generate multiple *prototypes* (feature vectors) for each event class by hypothesizing that having more than one prototype would help better capture the intra-class variation across modalities and training examples, and create a more robust representation.

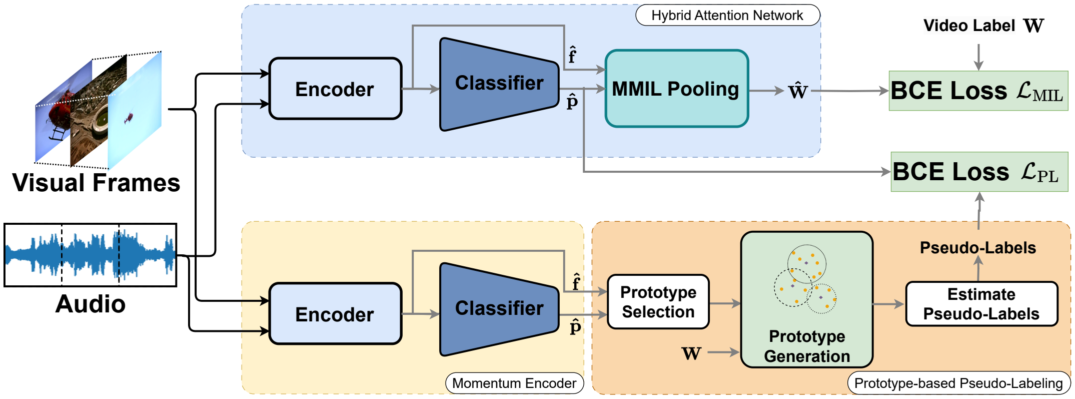

Figure 2. An overview of our method. The top branch is trained using MIL with weak labels *and* using full supervision with pseudo labels, which are generated by the bottom branch. This is done by first selecting key segments for each event from the training data and clustering them to construct a set of prototype features for the event, used to guide pseudo-label generation.

First, we extract features for all segments in $\mathcal{P}(c)$ and $\mathcal{N}(c)$ using Eq. (1), and call the resulting feature sets $\mathcal{F}_\mathcal{P}(c)$ and $\mathcal{F}_\mathcal{N}(c)$. Here, instead of f_θ, we use a momentum encoder [19] $f_{\tilde{\theta}}$ having the same architecture but whose weights are set (after each update to θ) as an exponential moving average (EMA) of θ over training steps: $\tilde{\theta} \leftarrow (1-\eta)\tilde{\theta} + \eta\theta$, where $\eta \in (0,1)$. The momentum encoder is more stable and less sensitive to noise from weak supervision, leading to slightly improved performance.

Next, we perform k-means clustering separately on $\mathcal{F}_\mathcal{P}(c)$ and $\mathcal{F}_\mathcal{N}(c)$ to get two sets of cluster centroids. We call the centroids *prototypes*:

$$\mathcal{C}_\mathcal{P}(c) = \{\mathbf{f}^+_\mathbf{i}\}_{i=1}^{k_p}, \quad \mathcal{C}_\mathcal{N}(c) = \{\mathbf{f}^-_\mathbf{j}\}_{j=1}^{k_n}. \tag{8}$$

Clustering for finding prototypes imposes a natural bottleneck [26], helping discard irrelevant information in the segments. We repeat this for each event class.

4.2. Estimating Pseudo Labels

We assign pseudo labels to each segment in a training video by considering its feature similarities with the prototypes for each class. We assume that if an audio/visual segment contains event c, it would be *on average* closer to $\mathcal{P}(c)$ in feature space than to $\mathcal{N}(c)$. For this, we explore two approaches: (i) *hard labeling*, which assigns the label of the closest-on-average set among the prototype sets for class c:

$$\mathbf{y}^{\mathbf{m}}_{\mathbf{t}}(c) = w_c \cdot \mathbb{1}\left[\frac{\sum_{\mathbf{f}^+\in\mathcal{C}_\mathcal{P}(c)} e^{(\mathbf{f}^+)^T\mathbf{f}^{\mathbf{m}}_{\mathbf{t}}}}{|\mathcal{C}_\mathcal{P}|} > \frac{\sum_{\mathbf{f}^-\in\mathcal{C}_\mathcal{N}(c)} e^{(\mathbf{f}^-)^T\mathbf{f}^{\mathbf{m}}_{\mathbf{t}}}}{|\mathcal{C}_\mathcal{N}|}\right], \tag{9}$$

and (ii) *soft labeling*, which instead assigns a probability score:

$$\mathbf{y}^{\mathbf{m}}_{\mathbf{t}}(c) = w_c \cdot \frac{\frac{1}{|\mathcal{C}_\mathcal{P}|}\sum_{\mathbf{f}^+\in\mathcal{C}_\mathcal{P}(c)} e^{(\mathbf{f}^+)^T\mathbf{f}^{\mathbf{m}}_{\mathbf{t}}/\tau}}{\sum_{\mathcal{S}\in\{\mathcal{C}_\mathcal{P},\mathcal{C}_\mathcal{N}\}} \frac{1}{|\mathcal{S}|}\sum_{\mathbf{f}'\in\mathcal{S}(c)} e^{(\mathbf{f}')^T\mathbf{f}^{\mathbf{m}}_{\mathbf{t}}/\tau}} \in (0,1), \tag{10}$$

where $\tau > 0$ is a temperature hyperparameter.

When training our model, we generate pseudo labels $\{\mathbf{y}^{\mathbf{m}}_{\mathbf{t}}\}$ after every epoch for each training video and use them to fully supervise the segment-level probabilities $\{\mathbf{p}^{\mathbf{m}}_{\mathbf{t}}\}$ in Eq. (2) using the binary cross-entropy loss

$$\mathcal{L}_{\text{PL}} = \text{CE}(\mathbf{p}^{\mathbf{m}}_{\mathbf{t}}, \mathbf{y}^{\mathbf{m}}_{\mathbf{t}}), \tag{11}$$

averaged over segments and training examples.

4.3. Contrastive Learning with Pseudo Labels

Given that our method exploits feature similarities between segments based on the events they may contain, we use contrastive learning based on pseudo labels to help semantically structure the feature space. Since multiple events can occur simultaneously, for each audio/visual segment, we consider a *similar* segment as one estimated to have *some* common event(s) and a *dissimilar* segment as one estimated to have no common events. Formally, given a batch of training examples, let $B = \{\mathbf{f}^{\mathbf{m}}_{\mathbf{t}} \to \mathbf{y}^{\mathbf{m}}_{\mathbf{t}}\}$ be a dictionary from features to the assigned pseudo labels for all segments and over *all* videos in the batch. For every feature $\mathbf{f} \in B$, the similar feature-set is then $\mathcal{S}(\mathbf{f}) = \{\mathbf{g} \in B \mid B(\mathbf{g}) \cap B(\mathbf{f}) \neq \phi\}$ and the dissimilar set is $\mathcal{D}(\mathbf{f}) = \{\mathbf{g} \in B \mid B(\mathbf{g}) \cap B(\mathbf{f}) = \phi\}$. To pull segments with a common predicted event closer in the feature space, we use the contrastive loss [38]

$$\mathcal{L}_{\text{CL}} = -\log \frac{\sum_{\mathbf{g}\in\mathcal{S}(\mathbf{f})} e^{\mathbf{g}^T\mathbf{f}/\tau_c}}{\sum_{\mathbf{g}\in\mathcal{S}(\mathbf{f})} e^{\mathbf{g}^T\mathbf{f}/\tau_c} + \sum_{\mathbf{h}\in\mathcal{D}(\mathbf{f})} e^{\mathbf{h}^T\mathbf{f}/\tau_c}}, \tag{12}$$

Method	Audio		Visual		Audio-Visual		Type@AV		Event@AV	
	Seg.	**Event**	**Seg.**	**Event**	**Seg.**	**Event**	**Seg.**	**Event**	**Seg.**	**Event**
AVE* [47]	47.2	40.4	37.1	34.7	35.4	31.6	39.9	35.5	41.6	36.5
AVSDN* [31]	47.8	34.1	52.0	46.3	37.1	26.5	45.7	35.6	50.8	37.7
HAN [48]	60.1	51.3	52.9	48.9	48.9	43.0	54.0	47.7	55.4	48.0
MA [52]	60.3	53.6	60.0	56.4	55.1	49.0	58.9	53.0	57.9	50.6
CVCM-MA [32]	60.8	53.8	63.5	58.9	57.0	49.5	60.5	54.0	59.5	52.1
MGN-MA [36]	60.2	50.9	61.9	59.7	55.5	49.6	59.2	53.4	58.7	49.9
JoMoLD [4]	61.3	53.9	63.8	59.9	57.2	49.6	60.8	54.5	59.9	52.5
PoiBin [40]	63.1	54.1	63.5	60.3	57.7	51.5	61.4	55.2	60.6	52.3
CMPAE [11]	64.2	56.6	66.4	63.7	59.2	51.8	63.3	57.4	62.8	55.7
Ours	**65.9**	**57.3**	**66.7**	**64.3**	**61.9**	**54.3**	**64.8**	**59.9**	**63.7**	**57.9**

Table 1. Results (% F1-scores) on all metrics defined in Tian et al. [48]. '*' indicates reproduced performance. The best and second best results are in **bold** and underline, respectively.

Algorithm 1: Pseudocode for our method

```
Init: model params $\psi = \{\theta, \phi\}$, EMA params $\psi_E = \{\theta_E, \phi_E\}$
while $\psi$ has not converged do
    $\# estimate\ segment - level\ pseudo\ labels$
    for $(x, \mathbf{w})$ in train set do
        $\{\mathbf{y_t^m}\} \leftarrow p_\theta(\cdot|\mathbf{w}, x)$ ;
    end
    $\# train$
    for $(x, \mathbf{w})$ in train set do
        $\{\mathbf{p_t^m}\} \leftarrow p_{\theta,\phi}(\cdot|x)$ ;
        $\hat{\mathbf{w}} = \sigma_{\theta,x}\left(\{\mathbf{p_t^m}\}\right)$;
        $\mathcal{L} = \mathcal{L}_{\text{MIL}} + \lambda_{\text{PL}}\ \mathcal{L}_{\text{PL}} + \lambda_{\text{CL}}\ \mathcal{L}_{\text{CL}}$;
        $\psi \leftarrow\ \psi - \alpha \nabla_\psi \mathcal{L}$ ;
        $\psi_E \leftarrow\ (1-\eta)\psi_E + \eta\psi$ ;
    end
end
```

averaged over all $\mathbf{f} \in B$, where $\tau_c > 0$ is a temperature hyperparameter.

4.4. Training and Inference

We train our model using the following loss in an end-to-end fashion:

$$\mathcal{L} = \mathcal{L}_{\text{MIL}} + \lambda_{\text{PL}}\ \mathcal{L}_{\text{PL}} + \lambda_{\text{CL}}\ \mathcal{L}_{\text{CL}}, \qquad (13)$$

where $\lambda_{\text{PL}}, \lambda_{\text{CL}} > 0$ are hyperparameters. During training, we generate pseudo labels once every epoch for all training examples using the best model so far. During inference, we predict the segment-level probabilities using the non-parametric prototypical pseudo-labeling (Eq. (10)) with video-level predictions from Eq. (3).

5. Experiments

5.1. Setup

LLP Dataset. We conduct experiments on the *Look, Listen and Parse* (LLP) dataset [48] which consists of 11849 YouTube videos, each of 10s duration, labeled into 25 event categories. These videos are unconstrained with a wide variety of scene content, including daily activities, music performances, and vehicle sounds. We use 10000 videos with weak (video-level) labels for training. The remaining 1849 fully-annotated videos (with segment-level labels) are used for validation and testing. The train/val/test split has been provided in the LLP dataset.

Evaluation Metrics. Following previous work, we use F1-scores on audio, visual, and audio-visual events as evaluation metrics. These are computed both at the segment and event level. We also include the aggregate metrics "Type@AV" and "Event@AV", again computed at the segment and event level. See Tian et al. [48] for a full explanation of metrics.

Implementation Details. We use ResNet-152 [18] pre-trained on ImageNet [6] and R(2+1)D-18 [49] pre-trained on Kinetics-400 [27] as visual feature extractors to generate a 512-d visual feature per segment. We use VGGish [20] pre-trained on AudioSet [14] to generate a 128-d audio feature per segment. We use the Adam optimizer [28] with a batch size of 16 and a learning rate $\alpha = 3e-4$ for 50 epochs. We take $\eta = 0.999$ for the momentum encoder, $\lambda_{\text{PL}} = \lambda_{\text{CL}} = 0.1$ in Eq. (13), $\gamma = 1$ in Eq. (5), $\beta = 0.2$ in Eq. (6), $\tau = 0.1$ in Eq. (10), and $\tau_c = 0.2$ in Eq. (12). We set $k_p = k_n = 10$ in Eq. (8). Hyperparameter values are based on validation performance.

5.2. Comparisons with Prior Work

For a fair comparison, all methods considered use the same pre-trained feature extractors and train/val/test split.

Quantitative. We compare methods in Table 1 on all the

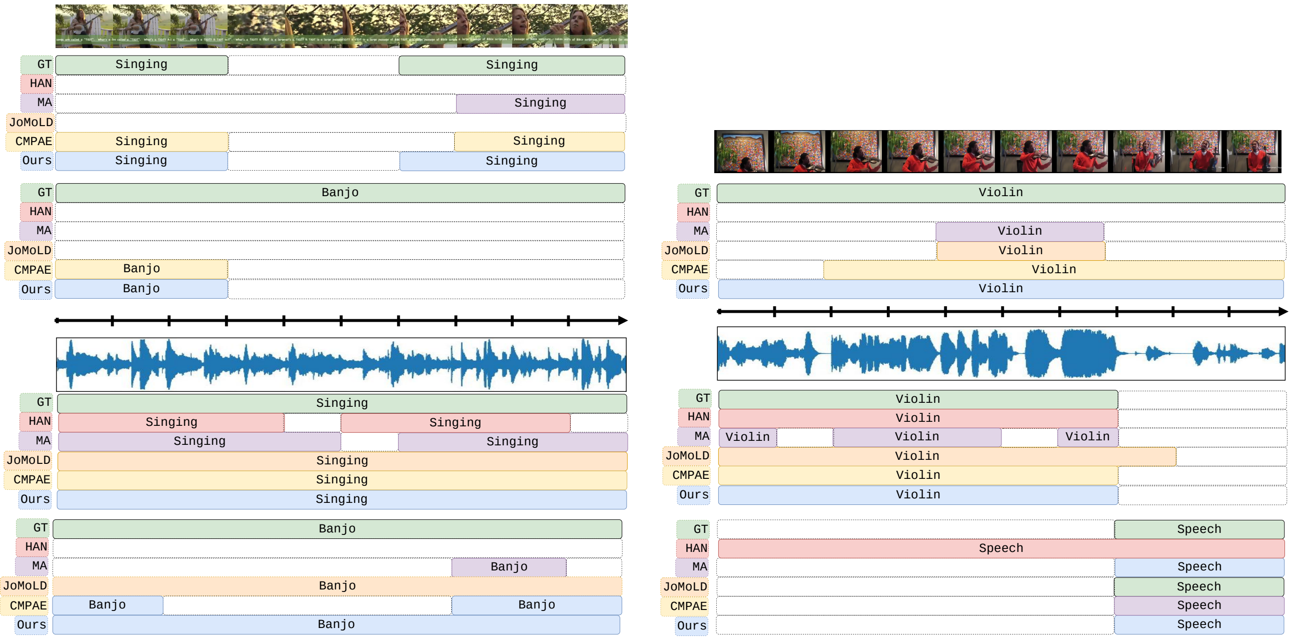

Figure 3. Audio-Visual Video Parsing results of our methods with HAN [48], MA [52], JoMoLD [4] and CMPAE [11] on two videos.

Method	Audio		Visual		Audio-Visual		Type@AV		Event@AV	
	Seg.	Event	Seg.	Event	Seg.	Event	Seg.	Event	Seg.	Event
Base	59.5	51.1	58.7	53.4	54.5	47.4	57.6	50.6	56.5	48.5
Base+NPL$_H$	58.9	52.1	58.5	49.8	55.7	48.2	57.8	50.1	55.7	46.6
Base+NPL$_S$	59.3	52.5	59.0	51.1	56.0	48.7	58.2	50.8	56.2	47.6
Base+PPL$_H$	62.5	57.4	62.9	59.6	59.6	54.0	61.9	56.3	60.3	55.7
Base+PPL$_S$	**65.9**	**57.3**	**66.7**	**64.3**	**61.9**	**54.3**	**64.8**	**59.9**	**63.7**	**57.9**

Table 2. Performance for different pseudo labeling strategies.

metrics listed earlier. We see that our method outperforms others on audio, audio-visual, Type@AV, and Event@AV scores at both the segment and event levels. In particular, we achieve an average improvement of 2.6 points for audio-visual events. Averaged over all metrics, our method achieves a 1.86 point improvement over the previous best method CMPAE [11]. Since we do not require additional learnable parameters to generate pseudo labels, this improvement is with a comparable number of parameters to previous methods.

Qualitative. We qualitatively compare our method with some previous methods on two examples in Fig. 3. Here, Fig. 3 (left) contains events "Singing" and "Banjo" in both modalities. Fig. 3 (Right) shows another scene with the event "Speech" in only the audio modality, and "Violin" occurring in both modalities. As evident, the common failure mode of previous methods is that they detect only a portion of the duration of events, while ours more commonly detects the full extent of events. As discussed in Sec. 1, this could be because of multiple-instance learning encouraging the model to pick out only the most discriminative segments containing an event, as opposed to all segments.

Computational Cost. When training, our approach takes 67.4s per epoch (17.1s for forward+backward propagation and 50.3s for prototype-based pseudo labeling). Inference time is 5.9ms on average per video, which is comparable to the baselines (2.9ms).

5.3. Analysis

Pseudo-Labeling Strategies. We report an ablation study in Table 2 to validate our specific pseudo-labeling method. Here, Base is the baseline MIL model explained in Sec. 3.2. Base+NPL refers to the base model trained with *naive* pseudo labeling, i.e., by directly using the segment-level predictions of the model as pseudo labels for re-training. In this case, H and S refer to using hard (binarized) and soft (continuous) pseudo labels, respectively. We observe no consistent improvement with naive pseudo labeling, likely due to the MIL model producing incorrect predictions for the non-discriminative segments for an event, leading to less reliable pseudo labels. Base+PPL refers to the base model trained with *prototype*-based pseudo labels as described in Sec. 4.2. In this case, H and S refer to using hard and soft pseudo labels, as defined in Eqs. (9) and (10), respectively. Our method significantly improves performance by generating more reliable pseudo labels throughout a training video. We also see that soft labeling is better than hard labeling. Since segment-level predictions may be noisy, we expect mistakes to affect performance more with hard rather than soft labels.

Reliability of Pseudo Labels. We now assess if the pseudo labels generated by our method are reliable for re-training. For this, we take the pre-trained baselines HAN [48] and MA [52], generate prototypes for each event from the train-

Method	Audio		Visual		Audio-Visual		Type@AV		Event@AV	
	Seg.	**Event**	**Seg.**	**Event**	**Seg.**	**Event**	**Seg.**	**Event**	**Seg.**	**Event**
HAN	60.1	51.3	52.9	48.9	48.9	43.0	54.0	47.7	55.4	48.0
+PPL$_{\text{Test}}$	**62.5**	**55.4**	**55.3**	**51.1**	**52.3**	**46.9**	**56.0**	**50.9**	**58.3**	**50.6**
MA	60.3	53.6	60.0	56.4	55.1	49.0	58.9	53.0	57.9	50.6
+PPL$_{\text{Test}}$	**61.7**	**55.4**	**61.8**	**57.9**	**57.5**	**51.6**	**60.6**	**55.0**	**59.4**	**52.6**

Table 3. Evaluating prototype-based pseudo labels on the test set.

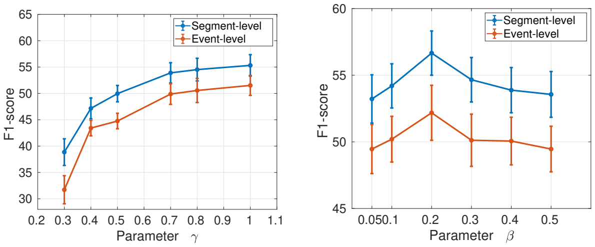

Figure 4. Performance for different γ and β choices.

Method	Audio		Visual		Audio-Visual		Type@AV		Event@AV	
	Seg.	**Event**	**Seg.**	**Event**	**Seg.**	**Event**	**Seg.**	**Event**	**Seg.**	**Event**
Base	59.5	51.1	58.7	53.4	54.5	47.4	57.6	50.6	56.5	48.5
No-CL	64.1	56.4	59.6	57.3	57.1	49.6	61.2	56.4	60.8	56.8
With-CL	**65.9**	**57.3**	**66.7**	**64.3**	**61.9**	**54.3**	**64.8**	**59.9**	**63.7**	**57.9**

Table 4. Effect of contrastive learning (CL) with pseudo labels.

ing data, and simply predict the prototype-based pseudo labels (Eq. (9)) on the test set for evaluation (i.e., discard the classifiers at test time). We call the resulting classifiers HAN+PPL$_{\text{Test}}$ and MA+PPL$_{\text{Test}}$ and show results in Table 3. With no other change to the models other than the classifiers, we see significant improvements in all metrics, showing that our pseudo labels are indeed reliable estimates of the true labels. Note that for the results in Table 1, we recompute pseudo labels on the training data every epoch, with a continually improving base model.

Effect of γ and β. The hyperparameters γ and β in Eqs. (5) and (6) roughly control the proportion of segments that will be selected as positives and negatives for assigning pseudo labels. We experiment with different values and plot the average of all segment/event-level metrics in Fig. 4. Performance is susceptible to γ, which controls positives, and is tolerant to a slight variation in β. The best performance is for $\gamma = 1$ and $\beta = 0.2$.

Effect of Contrastive Learning. We analyze the impact of contrastive learning using pseudo labels (Eq. (12)) by simply setting $\lambda_{\text{CL}} = 0$ in Eq. (13). Results are reported in Tab. 4. While only training with prototype-based pseudo labels (No-CL) significantly improves the baseline, adding a contrastive loss based on our pseudo labels (With-CL) further yields a large improvement.

Number of Clusters. We take $k_p = k_n$ and try values in $\{1, 2, ..., 100\}$. We plot the average of all segment/event-level metrics in Fig. 5. The best performance was for $k = 10$ clusters per class. Having fewer clusters helps capture more general features for an event instead of instance-specific details, but having too few hurts performance by discarding intra-class variation.

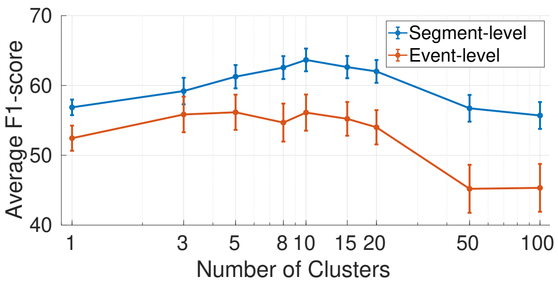

Figure 5. Performance for different choices of the number of clusters/prototypes per event.

Method	tIoU							Avg.		
	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]**
EM-MIL [34]	59.1	52.7	45.5	36.8	30.5	22.7	16.4	44.9	30.4	37.7
ASM-Loc [17]	71.2	65.5	57.1	46.8	36.6	25.2	13.4	55.4	35.8	45.1
DELU [3]	71.5	66.2	56.5	47.7	**40.5**	**27.2**	15.3	**56.5**	37.4	46.4
Ours	**72.7**	**66.9**	**57.9**	46.9	37.4	26.8	**20.1**	56.4	**37.8**	**46.9**

Table 5. Results (% mAP) for the TAL task at different tIoU thresholds.

5.4. Results for a Different Task

We conduct preliminary experiments for the weakly-supervised Temporal Action Localization (TAL) task [10] on the THUMOS14 dataset [23] to further validate our method. TAL aims to temporally localize and classify visual events in videos into known categories. We report mean average precisions (mAP) at different temporal intersection over union (tIoU) thresholds and the results are in Table 5. We achieve better or comparable performance w.r.t. the current state-of-the-art method DELU [3], and outperform previous methods. Check the supplementary for a full comparison.

6. Interpretation as Expectation-Maximization

We now show that iteratively re-estimating soft pseudo labels and training the model under weak + strong supervision (Eq. (13)) can be interpreted as an expectation-maximization (EM) algorithm, providing further insight for our training procedure. For this, we view the segment-level labels $y = \{(\mathbf{y_t^a}, \mathbf{y_t^v})\}_{t=1}^T$ for each video as *latent variables* since they are not observed when training under weak supervision. The observed variables are the video segments $x = \{(x_t^a, x_t^v)\}_{t=1}^T$ and weak labels $\mathbf{w}$. For clarity, assume a single event class, i.e., $\mathbf{w}, \mathbf{y_m^t} \in \{0, 1\}$.

Our goal is to maximize the data log-likelihood $\mathcal{L}(\theta) = \ln p_\theta(\mathbf{w}|x)$ over model parameters θ (assume just one data point x for clarity). Instead of maximizing this directly, we can attempt to maximize the variational lower bound [29] $\mathcal{L}_{\text{VLB}}$ for this likelihood (random variables are capitalized):

$$\begin{aligned}\mathcal{L}_{\text{VLB}}(\theta,\phi) = &\mathop{\mathbb{E}}_{Y\sim q_{\phi}(\cdot|\mathbf{w},x)}[\ln p_{\theta}(\mathbf{w}|Y,x)] \\ &- \mathrm{D}_{\text{KL}}\left[q_{\phi}(\cdot|\mathbf{w},x)||p_{\theta}(\cdot|x)\right], \quad (14)\end{aligned}$$

where $q_{\phi}(y|\mathbf{w},x)$ is *any* conditional distribution over labels y, with parameters ϕ, meant to approximate the true posterior $p_{\theta}(y|x)$. The EM algorithm alternates between two steps, first approximating the posterior by maximizing $\mathcal{L}_{\text{VLB}}$ over ϕ (E step), and then improving the data likelihood by maximizing it over θ (M step). Let $j \geq 1$ index the iterations (note, θ and ϕ here are not the same as in Sec. 3.2).

E Step. The j-th update is given by

$$q_{\phi_j}(y|\mathbf{w},x) = p_{\theta_{j-1}}(y|\mathbf{w},x), \quad \forall y. \quad (15)$$

Assuming conditional independence of the segment-level labels given the video and its weak labels, we can express the right-hand side as $p_{\theta}(y|\mathbf{w},x) = \prod_{t,m} p_{\theta}(\mathbf{y_t^m}|\mathbf{w},x)$, computed using

$$p_{\theta}(\mathbf{y_t^m}|\mathbf{w},x) := \begin{cases} p_{\theta}(\mathbf{y_t^m}|x) & \text{if } \mathbf{w}=1 \\ 1-\mathbf{y_t^m} & \text{if } \mathbf{w}=0, \end{cases} \quad (16)$$

where $p_{\theta}(\mathbf{y_t^m}|x)$ represents the prediction of the model for modality m and segment t in video x. Thus, $p_{\theta}(\mathbf{y_t^m}|\mathbf{w},x)$ becomes a *soft* pseudo label for the segment x_m^t.

M Step. The j-th update finds θ_j by maximizing $\mathcal{L}_{\text{VLB}}(\theta,\phi_j)$, which, due to Eq. (15), is equivalent to

$$\begin{aligned}\min_{\theta} \quad &-\sum_{y} p_{\theta_{j-1}}(y|\mathbf{w},x)\ln p_{\theta}(y|x) \\ &-\sum_{y} p_{\theta_{j-1}}(y|\mathbf{w},x)\ln p_{\theta}(\mathbf{w}|y,x), \quad (17)\end{aligned}$$

where $p_{\theta}(y|\mathbf{w},x)$ and $p_{\theta}(y|x)$ are computed as in Eq. (16). The **first term** in Eq. (17) can be expanded as the total cross-entropy between the segment-level predictions $p_{\theta}(\mathbf{y_t^m}|x)$ and the pseudo-label targets $p_{\theta_{j-1}}(\mathbf{y_t^m}|\mathbf{w},x)$, and is therefore equal to $\mathcal{L}_{\text{PL}}$ (see Eq. (13)). To understand the second term in Eq. (17), note that in our setting, $p_{\theta}(\mathbf{w}|y,x) = p_{\theta}(\mathbf{w}|x)$ since we make video-level predictions using MIL-pooling *without* access to the segment-level labels (y). The second term then reduces to $-\ln p_{\theta}(\mathbf{w}|x)$, which is simply the negative log-likelihood of the video-level label ($\mathbf{w}$). This means the **second term** is equal to $\mathcal{L}_{\text{MIL}}$ (see Eq. (13)). I.e., the M step is equivalently

$$\min_{\theta} \quad -\sum_{y} p_{\theta_{j-1}}(y|\mathbf{w},x)\ln p_{\theta}(y|x) - \ln p_{\theta}(\mathbf{w}|x). \quad (18)$$

If we simply maximized the data log-likelihood $\mathcal{L}(\theta)$ directly (no pseudo labels), this could converge to a solution that picks out just the most discriminative segments in each video, which may suffice to reduce the video-level loss $\mathcal{L}_{\text{MIL}}$. As we showed in this paper, using pseudo labels can help mitigate this, but the quality of the estimated pseudo labels has a crucial role. E.g., hard labeling is inferior to soft labeling since it does not attempt to compute the E step (Eq. (15)) exactly. While soft labeling *does* attempt an exact E step, better inductive biases (such as prototype-based labeling with contrastive learning) here can help reach a favorable solution, as reported in Sec. 5.3 and Table 2. We also compare with an off-the-shelf pseudo-labeling method [34] in the supplementary material and find that we significantly outperform it, showing that it is not pseudo-labeling per se that works for AVVP - *how* we estimate the pseudo labels is paramount.

7. Conclusion

We proposed a prototype-based pseudo-labeling method for weakly-supervised AVVP that outperforms existing methods. Some interesting findings are that (i) naive pseudo-labeling does not produce reliable training targets, (ii) contrastive learning using pseudo labels is an effective strategy, and (iii) training with weak and re-estimated soft pseudo labels is an EM algorithm.

We end with some limitations and suggestions for future work. First, performance is sensitive to hyperparameter β as shown in Fig. 4. This could be because identifying negatives for a class from the predictions of an MIL model is less likely to be accurate than identifying positives. Moreover, negative segments for an event would have a much larger variation in content. Second, events with larger intra-class variation may need more prototypes to represent, while we assume the same number to be optimal for all events for convenience. Third, while we can interpret our method as an EM algorithm, EM may only converge to a locally optimal solution. As discussed in Sec. 6, future work could explore inductive biases in the pseudo-labeling step that favor better solutions. Finally, it is not clear how many related video-/image-based prediction tasks our method can generalize to. While we provided results for TAL in Sec. 5.4, more evidence of adapting our method to related tasks would be useful in future work.

Acknowledgement

Support from Institute of Eminence (IoE) project No. SB22231269EEETWO005001 for Research Centre in Computer Vision is gratefully acknowledged.

References

[1] Hakan Bilen and Andrea Vedaldi. Weakly Supervised Deep Detection Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2846–2854, 2016. 2

[2] Ricardo Cabral, Fernando De la Torre, Joao Paulo Costeira, and Alexandre Bernardino. Matrix Completion for Weakly-Supervised Multi-Label Image Classification. *IEEE transactions on pattern analysis and machine intelligence*, 37(1): 121–135, 2014. 2

[3] Mengyuan Chen, Junyu Gao, Shicai Yang, and Changsheng Xu. Dual-evidential learning for weakly-supervised temporal action localization. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*, pages 192–208. Springer, 2022. 7

[4] Haoyue Cheng, Zhaoyang Liu, Hang Zhou, Chen Qian, Wayne Wu, and Limin Wang. Joint-modal label denoising for weakly-supervised audio-visual video parsing. In *Computer Vision – ECCV 2022*, pages 431–448, Cham, 2022. Springer Nature Switzerland. 5, 6

[5] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1): 21–27, 1967. 2

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5

[7] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the Multiple Instance Problem with Axis-Parallel Rectangles. *Artificial Intelligence*, 89(1):31–71, 1997. 1

[8] Nanqing Dong and Eric P Xing. Few-shot semantic segmentation with prototype learning. In *BMVC*, 2018. 2

[9] Jie Fu, Junyu Gao, Bing-Kun Bao, and Changsheng Xu. Multimodal imbalance-aware gradient modulation for weakly-supervised audio-visual video parsing. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023. 2

[10] Adrien Gaidon, Zaid Harchaoui, and Cordelia Schmid. Temporal localization of actions with actoms. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2782–2795, 2013. 1, 7

[11] Junyu Gao, Mengyuan Chen, and Changsheng Xu. Collecting cross-modal presence-absence evidence for weakly-supervised audio-visual event perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18827–18836, 2023. 2, 5, 6

[12] Salvador Garcia, Joaquin Derrac, Jose Cano, and Francisco Herrera. Prototype selection for nearest neighbor classification: Taxonomy and empirical study. *IEEE transactions on pattern analysis and machine intelligence*, 34(3):417–435, 2012. 2

[13] Weifeng Ge, Xiangru Lin, and Yizhou Yu. Weakly Supervised Complementary Parts Models for Fine-Grained Image Classification from the Bottom Up. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3034–3043, 2019. 2

[14] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780. IEEE, 2017. 5

[15] Jacob Goldberger, Geoffrey E Hinton, Sam Roweis, and Russ R Salakhutdinov. Neighbourhood components analysis. *Advances in neural information processing systems*, 17, 2004. 2

[16] Samantha Guerriero, Barbara Caputo, and Thomas Mensink. DeepNCM: Deep Nearest Class Mean Classifiers, 2018. 2

[17] Bo He, Xitong Yang, Le Kang, Zhiyu Cheng, Xin Zhou, and Abhinav Shrivastava. 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)*, pages 13925–13935, 2022. 7

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[19] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 4

[20] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. Cnn architectures for large-scale audio classification. In *2017 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 131–135. IEEE, 2017. 5

[21] Mengying Hu, Hu Han, Shiguang Shan, and Xilin Chen. Weakly Supervised Image Classification through Noise Regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11517–11525, 2019. 2

[22] Linjiang Huang, Yan Huang, Wanli Ouyang, and Liang Wang. Relational prototypical network for weakly supervised temporal action localization. In *Proceedings of the AAAI conference on artificial intelligence*, pages 11053–11060, 2020. 2

[23] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23, 2017. 7

[24] Maximilian Ilse, Jakub Tomczak, and Max Welling. Attention-based deep multiple instance learning. In *International conference on machine learning*, pages 2127–2136. PMLR, 2018. 2, 3

[25] Saumya Jetley, Bernardino Romera-Paredes, Sadeep Jayasumana, and Philip Torr. Prototypical priors: From improving classification to zero-shot learning. *arXiv preprint arXiv:1512.01192*, 2015. 2

[26] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9865–9874, 2019. 4

[27] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The Kinetics Human Action Video Dataset, 2017. 5

[28] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *ICLR (Poster)*, 2015. 5

[29] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes, 2022. 8

[30] Jatin Lamba, Jayaprakash Akula, Rishabh Dabral, Preethi Jyothi, Ganesh Ramakrishnan, et al. Cross-modal learning for audio-visual video parsing. *arXiv preprint arXiv:2104.04598*, 2021. 1, 2, 3

[31] Yan-Bo Lin, Yu-Jhe Li, and Yu-Chiang Frank Wang. Dual-modality seq2seq network for audio-visual event localization. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2002–2006. IEEE, 2019. 5

[32] Yan-Bo Lin, Hung-Yu Tseng, Hsin-Ying Lee, Yen-Yu Lin, and Ming-Hsuan Yang. Exploring cross-video and cross-modality signals for weakly-supervised audio-visual video parsing. *Advances in Neural Information Processing Systems*, 34, 2021. 2, 3, 5

[33] Guoqing Liu, Jianxin Wu, and Zhi-Hua Zhou. Key Instance Detection in Multi-Instance Learning. In *Proceedings of the Asian Conference on Machine Learning*, pages 253–268, Singapore Management University, Singapore, 2012. PMLR. 3

[34] Zhekun Luo, Devin Guillory, Baifeng Shi, Wei Ke, Fang Wan, Trevor Darrell, and Huijuan Xu. Weakly-supervised action localization with expectation-maximization multi-instance learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, pages 729–745. Springer, 2020. 2, 7, 8

[35] Pascal Mettes, Elise Van der Pol, and Cees Snoek. Hyperspherical prototype networks. *Advances in neural information processing systems*, 32, 2019. 2

[36] Shentong Mo and Yapeng Tian. Multi-modal grouping network for weakly-supervised audio-visual video parsing. In *Advances in Neural Information Processing Systems*, 2022. 2, 3, 5

[37] Islam Nassar, Munawar Hayat, Ehsan Abbasnejad, Hamid Rezatofighi, and Gholamreza Haffari. ProtoCon: Pseudo-Label Refinement via Online Clustering and Prototypical Consistency for Efficient Semi-Supervised Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11641–11650, 2023. 2

[38] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding, 2018. 4

[39] Alejandro Pardo, Humam Alwassel, Fabian Caba Heilbron, Ali Thabet, and Bernard Ghanem. RefineLoc: Iterative Refinement for Weakly-Supervised Action Localization. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3318–3327, 2021. 2

[40] Kranthi Kumar Rachavarapu and A N Rajagopalan. Boosting positive segments for weakly-supervised audio-visual video parsing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10192–10202, 2023. 2, 5

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[42] Dawid Rymarczyk, Aneta Kaczyńska, Jarosław Kraus, Adam Pardyl, and Bartosz Zieliński. Protomil: Multiple instance learning with prototypical parts for fine-grained interpretability. *arXiv e-prints*, pages arXiv–2108, 2021. 2

[43] Ruslan Salakhutdinov and Geoff Hinton. Learning a nonlinear embedding by preserving class neighbourhood structure. In *Artificial intelligence and statistics*, pages 412–419. PMLR, 2007. 2

[44] Atsushi Sato and Keiji Yamada. Generalized learning vector quantization. *Advances in neural information processing systems*, 8, 1995. 2

[45] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017. 2

[46] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple Instance Detection Network with Online Instance Classifier Refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2843–2851, 2017. 2

[47] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *ECCV*, pages 247–263, 2018. 1, 5

[48] Yapeng Tian, Dingzeyu Li, and Chenliang Xu. Unified multisensory perception: Weakly-supervised audio-visual video parsing. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 436–454. Springer, 2020. 1, 2, 3, 5, 6

[49] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 5

[50] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016. 2

[51] Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng. Panet: Few-shot image semantic segmentation with prototype alignment. In *proceedings of the IEEE/CVF international conference on computer vision*, pages 9197–9206, 2019. 2

[52] Yu Wu and Yi Yang. Exploring heterogeneous clues for weakly-supervised audio-visual video parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1326–1335, 2021. 2, 3, 5, 6

[53] Zhirong Wu, Alexei A Efros, and Stella X Yu. Improving generalization via scalable neighborhood component analysis. In *Proceedings of the european conference on computer vision (ECCV)*, pages 685–701, 2018. 2

[54] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018. 2

[55] Wenjia Xu, Yongqin Xian, Jiuniu Wang, Bernt Schiele, and Zeynep Akata. Attribute prototype network for zero-shot learning. *Advances in Neural Information Processing Systems*, 33:21969–21980, 2020. 2

[56] Hong-Ming Yang, Xu-Yao Zhang, Fei Yin, and Cheng-Lin Liu. Robust classification with convolutional prototype learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3474–3482, 2018. 2

[57] Weiyi Yang, Richong Zhang, Junfan Chen, Lihong Wang, and Jaein Kim. Prototype-Guided Pseudo Labeling for Semi-Supervised Text Classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16369–16382, Toronto, Canada, 2023. Association for Computational Linguistics. 2

[58] Yuanhao Zhai, Le Wang, Wei Tang, Qilin Zhang, Junsong Yuan, and Gang Hua. Two-stream consensus network for weakly-supervised temporal action localization. In *European conference on computer vision*, pages 37–54. Springer, 2020. 2

[59] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning Deep Features for Discriminative Localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 2

[60] Jinxing Zhou, Dan Guo, Yiran Zhong, and Meng Wang. Improving audio-visual video parsing with pseudo visual labels. *arXiv preprint arXiv:2303.02344*, 2023. 2

[61] Tianfei Zhou, Wenguan Wang, Ender Konukoglu, and Luc Van Gool. Rethinking semantic segmentation: A prototype view. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2582–2593, 2022. 2