Mining and Unifying Heterogeneous Contrastive Relations for Weakly-Supervised Actor-Action Segmentation

Bin Duan[1] Hao Tang[2] Changchang Sun[1] Ye Zhu[3] Yan Yan[1]
[1]Illinois Institute of Technology [2]Carnegie Mellon University [3]Princeton University
{bduan2, csun39}@hawk.iit.edu, bjdxtanghao@gmail.com, yezhu@princeton.edu, yyan34@iit.edu

Abstract

We introduce a novel weakly-supervised video actor-action segmentation (VAAS) framework, where only video-level tags are available. Previous VAAS methods follow a synthesize-and-refine scheme, i.e., they first synthesize the pseudo-segmentation and recursively refine the segmentation. However, this process requires significant time costs and heavily relies on the quality of the initial segmentation. Unlike existing works, our method hierarchically mines contrastive relations to supplement each other for learning a visually-plausible segmentation model. Specifically, three contrastive relations are abstracted from the pixel-level and frame-level, i.e., low-level edge-aware, class-activation map aware, and semantic tag-aware relations. Then, the discovered contrastive relations are unified into a universal objective for training the segmentation model, regardless of their heterogeneity. Moreover, we incorporate motion cues and unlabeled samples to increase the discriminative power and robustness of the segmentation model. Extensive experiments indicate that our proposed method produces reasonable segmentation.

1. Introduction

Video **A**ctor-**A**ction **S**egmentation (VAAS) involves performing segmentation on both actors and actions within a video, as demonstrated in the case of *baby crawling*, where *baby* represents the actor and *crawling* represents the action. Unlike traditional video object segmentation, which focuses solely on foreground object segmentation in a scene, and instance segmentation, which concentrates on separating and identifying individual objects, VAAS stands out due to its unique capability of segmenting different actors engaged in different actions simultaneously. This means that a VAAS method must distinguish between an actor (object) and two different semantic elements: actor-semantics and action-semantics. This distinctive requirement for discriminating between actions sets VAAS apart from other object segmentation techniques, making it a notably challenging task. Existing methods deal with the VAAS task through various approaches, such as joint learning [29, 31], supervoxel-based method [53], 2D/3D FCN [43], text-guidance instead of the original video label [18]. Despite the promising segmentation results achieved by these fully-supervised methods, the lack of pixel-level annotations has limited the practical applications of VAAS in real-world scenarios.

To address the scarcity of pixel-level annotations, there are a few works in the literature [8, 56] investigates VAAS in the weakly-supervised setting where only video-level actor-action tags are accessible. That is, we can only access the actor-action labels during training without fine-grained pixel-level annotations, making it more complex than the aforementioned fully-supervised methods. The seminal work of Yan *et al.* [56] introduced a set of ranking-supporting vector machines to replace classifiers. They considered each superpixel within the frame as the same category, which inevitably confuses the model due to this rough labeling. Another work by Chen *et al.* [8] proposed a 3D GCAM [44] to synthesize pixel-level pseudo-segmentation and then iteratively refined the initial pseudo-segmentation.

These weakly-supervised VAAS methods [8, 56] and even more general video object segmentation [25, 27, 46, 59], adhere to a similar synthesize-and-refine approach. This approach involves synthesizing pseudo-segmentations and iteratively refining them. However, this process entails significant time costs and heavily depends on the quality of the initial segmentation. Different from this intricate synthesize-and-refine paradigm, we propose to train a network for direct video segmentation without additional refinements. We demonstrate that heterogeneous contrastive relations can be extracted from raw frames, offering supervision for model training. Specifically, for low-level pixels, neighboring pixels with similar appearances are more likely to belong to the same semantic category. Conversely, pixels with distinct color variations need to be categorized separately. At a higher semantic level, the representation of an actor-action like *baby crawling* should be distinguishable from *dog running*. These inherent similarities and dissimilarities between frames naturally facilitate the establishment of diverse contrastive relations. This way, visually- and semantically similar representations should be attracted closer while their counterparts should be repelled away.

To tackle the VAAS task, we introduce the Visually and Semantically Contrastive Relations (VSCR) segmentation model. Our approach uniquely incorporates three contrastive relations: low-level edge-aware, class-activation map (CAM)-aware, and semantic tag-aware, supplementing each other for training. Notably, VSCR integrates unlabeled samples without CAM, enhancing network robustness and reducing the need for extensive pseudo-segmented data. This fusion of contrastive relations and unlabeled samples establishes a universal learning objective, enabling pseudo-segmentations to propagate to unlabeled frames. In addition to these contrastive relations, our strategy leverages motion cues to tackle the VAAS task. In contrast to methods like [8], which solely rely on actions during the initial pseudo-segmentation, we seamlessly integrate motion cues extracted from video clips during the model training. This infusion of motion cues augments the network's capacity to discern and classify actions effectively. Through the fusion of motion cues and the unified learning objective, our model acquires the ability to associate actions closely with the concurrency of video-level tags. For instance, when presented with a "dog-walking" video, our model identifies the associated actor-action even among videos containing diverse content. Our novel VSCR framework is illustrated in Figure 1, representing a reasonable solution trained with heterogeneous contrastive relations.

We summarize our contributions as follows:

- Our approach differs from previous methods [8, 56]. We introduce a new design that utilizes pixel- and frame-level contrastive relations for additional supervision, eliminating the need for further refinement.
- We present the VSCR segmentation network to tackle the weakly-supervised VAAS task. Additionally, we incorporate motion cues to enhance action discrimination and unlabeled samples to bolster model robustness, reducing the reliance on a large number of pseudo-segmentations.
- Leveraging our proposed techniques, the VSCR segmentation network outperforms state-of-the-art weakly-supervised methods in VAAS, delivering reasonable segmentations.

2. Related Work

Video Actor-Action Segmentation (VAAS). Video semantic segmentation aims to identify the object category of each pixel for every known object within a frame of the target video [28, 36, 41]. While action recognition tries to classify the action categories within a video [7, 47, 49]. VAAS highly relates to joint video semantic segmentation and action recognition [23, 60]. However, most of these segmentation methods assume a single dominant object in the video, and so does action recognition, where they assume only a single ongoing action. Despite the similarity, the joint semantic segmentation and action recognition are not directly comparable to VAAS. Another similar task is instance segmentation [20], but its segmentation remains consistent for all instances of the same object class, regardless of the actions associated with those instances.

Initially proposed by Xu *et al.* [54], VAAS has received significant attention in the computer vision community [18, 29, 31, 43, 53]. The works in the fully-supervised area fall into two categories: graph-based and two-stream-based. For graph-based methods [53], they first supervoxelize the video as their initial graphs with nodes being supervoxels, and then solve the problem as a graph cut problem. For the two-stream-based methods [29, 31, 43], the joint learning of two streams (frame- and video-level) is a common practice. For each stream, the methods are different from their pipelines, such as detection-based frameworks [29, 31], and convolutional LSTM [24] with FCN [43]. Note that all the above methods urge full pixel-level supervision to train the model.

The proposition of weakly-supervised VAAS by Yan *et al.* [56] addresses the high demand for full supervision by only using video-level tags. They initially generate pseudo-labels using supervoxelization and then train a set of ranking SVMs instead of classifiers for the final segmentation using CRF. The state-of-the-art method proposed by Chen *et al.* [8] follows a synthesize-and-refine scheme, where they mostly focus on selecting high-quality pseudo-segmentations and designing the stop criterion.

In contrast to conventional methods, we take a novel approach to address weakly-supervised VAAS tasks by harnessing the latent potential within frames and videos through a contrastive strategy. Unlike the prevalent approach of iteratively refining pseudo-segmentations [8, 46], we train our VSCR model with consistent pseudo-segmentations. The efficacy of our well-trained segmenter primarily stems from our introduced contrastive relations, which supply supplementary supervision, along with innovative techniques such as incorporating motion cues and unlabeled samples. This shift in methodology sets our approach apart from previous methods, positioning it as a novel and distinctive method within the field.

Contrastive Learning. Contrastive learning is widely used in weakly- and self-supervised representation learning methods [11, 19, 22, 35, 37]. To utilize the contrastive learning strategy, we have to define positive and negative samples. In general, the definitions of positive/negative pairs vary from application and application [12, 14, 15, 34, 62]. For a given anchor point in the representation space, contrastive learning aims to pull the anchor closer to its pos-

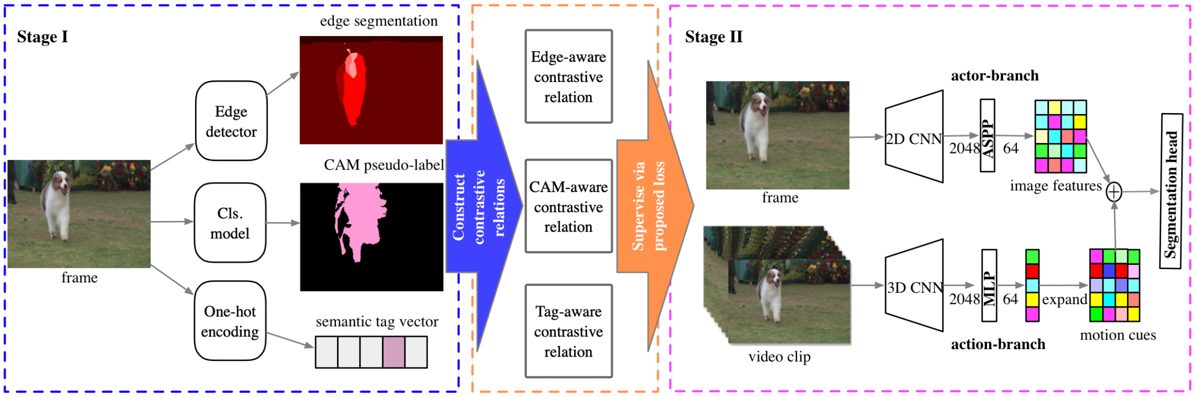


Figure 1. **Our VSCR segmentation architecture.** The edge segmentation and CAM pseudo-label are generated from a pretrained edge detector and classification model, while the semantic tag is encoded into a one-hot vector. The frame and video clip are input into respective CNN feature extractors to generate image-level representation and video-level motion cues. We use pretrained DeepLab [10] with ASPP convolution and I3D [5] as the corresponding backbone. Note that the I3D backbone is fixed where we focus on optimizing the MLP layer consisting of two linear layers. The expansion of motion cues is implemented by bi-linearly interpolating to the same size of image-level features. Our segmentation head is a two-layer MLP. It is worth noting that our model training is under two newly-proposed supervisions (*i.e.*, edge segmentation and semantic tag) plus CAM pseudo-annotation used in [8], along with a novel learning objective loss function to learn from both labeled and unlabeled samples. $\bigoplus$ denotes element-wise addition.

itive samples and, at the same time, push the anchor far away from its negative samples. This motivation of contrastive learning nominates itself as a well-suited tool in high-level tasks such as image recognition [11, 19, 22]. Recent works [26, 33, 58] introduce contrastive learning into semantic segmentation by maximizing the log-likelihood of extracted pixel features under a mixture of vMF distributions model. However, to the best of our knowledge, there is still no work to apply contrastive learning to VAAS. In our work, we define our positive/negative pairs based on our proposed heterogeneous contrastive relations. Moreover, our proposed method is different from semantic segmentation works [26, 33, 58] as we incorporate motion cues and unlabeled samples into training, and we formulate a novel objective to perform joint actor-action segmentation, different from most existing methods.

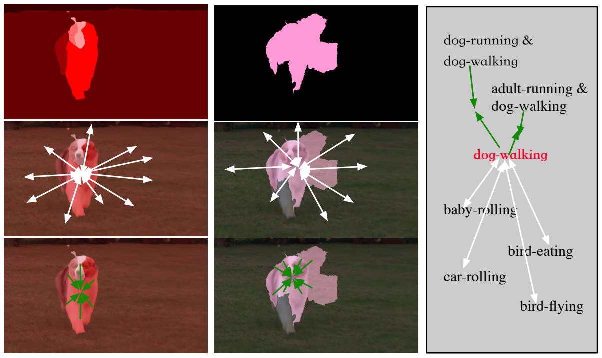
3. VSCR for Weakly-Supervised VAAS

In this section, we present an in-depth exploration of our proposed VSCR architecture (Figure 1), which serves as a solution for addressing the challenges posed by weakly-supervised VAAS. We commence by introducing the overarching structure of our framework, followed by an elaboration of how various contrastive relations are extracted. Then, we outline the formulation of a novel objective function that integrates these heterogeneous contrastive relations and unlabeled samples, thereby enhancing the capabilities of our segmentation model.

3.1. VSCR Segmentation Architecture

Our VSCR contains two stages: *Supervision construction* (Stage I) and Model training under constructed contrastive relations (Stage II). In Stage I, we extract edge segmentation, CAM pseduo–label, and semantic tag vector using edge detector, classification model, and one-hot encoding, respectively, as in Figure 1. The implementation details are in Sec. 4.2. After we obtain these pseudo-annotations, we construct the contrastive relations on the fly in Stage II using spherical kmeans clustering in the embedding space. We will introduce the mining and unifying of hetergeneous contrastive relation in the next subsection. Stage II fuses image feature and motion cues together to learn a robust segmentation model. Specifically, we adopt a simple yet effective two-stream architecture to jointly learn from a frame (*i.e.*, *actor-branch*) and a video sequence (*i.e.*, *action-branch*), as shown in Figure 1. The actor branch backbones a 2D DeepLab [10] with an Atrous Convolution layer (ASPP), while the action branch has an I3D [5] backbone. Different from the actor branch, the action branch backbone I3D encodes the video into a representation of a flatten 2048-dimensional vector. To match the feature dimension, we introduce a two-layer Multi-Layer Perceptron (MLP) to output the feature with the same dimension as the image representation. The final segmentation is obtained using Softmax. During training, the actor backbone (DeepLab) along with the ASPP layer, action-branch MLP, and segmentation head are trained using the proposed weakly-supervisions.

Learning with Motion Cues. Our target task is to generate pixel-wise semantic segmentation across the joint actor-action class space from input video data. The state-of-the-art method for weakly-supervised VAAS only considers the action during initial pseudo-labels generation. However, no motion/action is used during the actual segmentation model training. We propose to learn the segmentation model with



(a) edge-aware contrastive relation
(b) CAM-aware contrastive relation
(c) tag-aware contrastive relation

Figure 2. **Illustration of our contrastive relations.** Green lines denote that representations in the feature space should be pulled closer while white ones mean should be pushed away. Note that, for a given frame/video with single actor-action, *e.g.*, "*dog-walking*", the distance of its representation to its multi-action samples, *e.g.*, "dog-walking and adult-running" and "dog-walking and dog-running" should be closer compared to other irrelevant actor-actions, such as "baby-rolling".

motion cues. Our intuitions are: (**i**) Action awareness is the key to the VAAS problem. With motion cues included, our model can jointly learn actor and action representation, increasing the segmentation model's discriminative power, as suggested in [29,31]. (**ii**) Low-quality initial action pseudo-annotations require more runs to refine and may still result in suboptimal convergence. Compared with the well-trained image-level classification model $\geq 86\%$, the video-level classification model suffers from low prediction power $\leq 76\%$, which means that the predicted pixel-level action pseudo-annotation has more ambiguities, resulting inferior pseudo-annotations. Instead of using action in the initial generation of pseudo-annotations, we seamlessly integrate the motion cues into model training. With our novel loss function introduced later, we can jointly learn from frame and video, leading to a better segmentation model.

3.2. Mining and Unifying Heterogeneous Contrastive Relations

It is worth noting that VSCR provides the model with additional supervision other than pseudo-segmentations in [8] to learn a better segmentation model. The extra supervisions are products of our proposed contrastive relations. The contrastive relations are constructed from three different aspects – low-level edge-aware, CAM-aware, and semantic tag-aware. The high-level understanding of our designed contrastive relations is shown in Figure 2. Here, we first introduce the generation of pixel-level annotations, both edge and CAM annotations. Then, we present details of formulating contrastive relations for each aspect.

Pixel-Level Annotations. We generate two types of annotations: edge map and class activation map. We further utilize edge map to generate edge segmentation and SLIC algorithm [1] to refine class activation map as our pseudo-segmentations. Samples of annotations are shown in Figure 3. Here, CAM pseudo-annotation provides important pixel-level class information for learning the segmentation model, which we include in all our experiments. In Sec. 3.3, we propose a novel approach to learning from both pseudo-labeled and unlabeled (*i.e.*, no CAM annotation) with a new optimizable objective function. Moreover, edge segmentation focuses on the strong contour of the frame, which is most likely to be the boundary of different objects, leading the model to produce better clean-cut segmentation results.

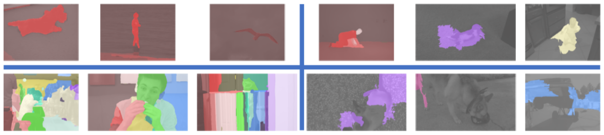


Figure 3. **Pixel-level annotations.** The **left** 3 columns are edge segmentations while the **right** 3 are CAM pseudo-labels. For all the images presented, we overlay masks as alpha channels on their corresponding raw RGB frames. High-quality masks are shown in the top row, whereas the bottom row is for low-quality annotations.

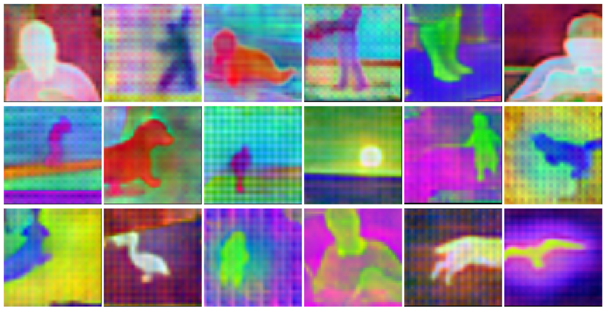


Figure 4. **Image embedding discriminates the actor regions.** Note that all embeddings are randomly pseudo-colored.

Spherical Kmeans Clustering. Since we have no ground-truth annotation for each pixel, it is intuitive that unsupervised approaches such as clustering can be utilized to mine the internal information among pixels as free supervisions. Previous works [26, 33, 39, 58] consider this classification problem as a clustering problem in the feature space, solved by spherical kmeans clustering [4]. This spherical kmeans clustering has a nice property to fit the general framework of contrastive learning, where it maximizes intra-class distances and minimizes inter-class distances. Analogically, suppose we assume negative pairs are in different clusters and positive pairs are in the same cluster. In that case, spherical kmeans can be used to guarantee the rationale of such positive and negative-pair construction. In this way, we can take the sample pairs within the same cluster as positive sample pairs and samples from different clusters as negative pairs. We present a sketch of the algorithm here for reference. Let $\boldsymbol{v}_i$ denotes the feature of pixel i, and h_i denotes the index of the class to which i belongs. Let R_h be the class set of pixels containing i, and $\boldsymbol{\mu}_h$

feature centroid. This spherical kmeans clustering algorithm uses Expectation-Maximization for finding the optimal partition, where E-step: $h_i\!=\arg\max_h \boldsymbol{v}_i^T\boldsymbol{\mu}_h$, and M-step: $\boldsymbol{\mu}\!=\!\Sigma_{i\in R_h}\boldsymbol{v}_i/||\Sigma_{i\in R_h}\boldsymbol{v}_i||$. Therefore, we utilize this spherical kmeans clustering as a tool to compose our heterogeneous contrastive relations into a universal optimization objective to optimize the segmentation model. This utilization of contrastive relations allows our model to learn discriminative representations, as shown in Figure 4.

Low-level Edge-aware Contrastive Relation. We generate edge segmentation using HED [38, 52] and gPb-owt-ucm [3], which is a common practice in weakly-supervised segmentations [26, 33, 58]. Edges are strong indicators of region homogeneity, *i.e.*, pixels of a coherent region, in general, have the same semantic label. This coherent prior of the semantic label plays an important role, especially in weakly-supervised segmentation tasks where the model should be capable of propagating labels throughout visually similar regions. To this end, we propose the contrastive relation based on contour-induced segmentation. Let i denote a pixel in the current frame, we denote $\mathcal{E}^+/\mathcal{E}^-$ (visually similar/different) as its positive/negative samples. Let c denote the resulting class that pixel i belongs to. Given the feature vector $\boldsymbol{v}_i$ of pixel i and centroids $\boldsymbol{\mu}$ of the partition, the posterior probability is:

$$p(h_i=c\,|\,\boldsymbol{v}_i,\boldsymbol{\mu})=\frac{\boldsymbol{e}^{k_e\boldsymbol{v}_i^T\boldsymbol{\mu}_c}}{\Sigma_{j\in R}\,\boldsymbol{e}^{k_e\boldsymbol{v}_i^T\boldsymbol{\mu}_j}}, \tag{1}$$

where k_e is a concentration hyper-parameter, set to 16 in all experiments, empirically. We minimizes the negative log-likelihood loss as in [26, 33, 58]:

$$\begin{aligned}\mathcal{L}_{\text{edge}}(i;\mathcal{E}^+,\mathcal{E}^-)&=-\sum_{j\in\mathcal{E}^+}\log p(h_i=j|\boldsymbol{v}_i,\boldsymbol{\mu})\\&=-\log\frac{\Sigma_{j\in\mathcal{E}^+}\,\boldsymbol{e}^{k_e\boldsymbol{v}_i^T\boldsymbol{\mu}_j}}{\Sigma_{j\in\mathcal{E}^+\cup\mathcal{E}^-}\,\boldsymbol{e}^{k_e\boldsymbol{v}_i^T\boldsymbol{\mu}_j}}.\end{aligned} \tag{2}$$

CAM-aware Contrastive Relation. Existing state-of-the-art weakly-supervised image/video segmentation methods follow a similar synthesize-and-refine scheme [2, 9, 16, 17, 45, 51, 55, 57]. However, different from the role CAM played in their work as it is the only driving force in the learning phase, CAM pseudo-annotation is just one of our three supervisions, as shown in this section. In our case, the pseudo-segmentations provide a rough pixel-wise estimation. When we create CAM pseudo-labels using the refined classification model, as described in Sec. 4.2, and utilize video-level labels from the same dataset, the CAM labels are inherently attuned to the specific semantic classes found within the dataset used for fine-tuning, making it less possible to classify unseen classes.

Let $\mathcal{C}^+/\mathcal{C}^-$ denote the positive/negative samples for pixel i, where positive/negative samples stand for pixels with the same/different labels, respectively. Let $\boldsymbol{v}_i,\boldsymbol{\mu}$ be the feature vector of pixel i and centroid of the clustering partition. We can calculate the CAM loss in the form of negative log-likelihood loss $\mathcal{L}_{\text{cam}}(i;\mathcal{C}^+,\mathcal{C}^-)\!=-\log\frac{\Sigma_{j\in\mathcal{C}^+}\,\boldsymbol{e}^{k_c\boldsymbol{v}_i^T\boldsymbol{\mu}_j}}{\Sigma_{j\in\mathcal{C}^+\cup\mathcal{C}^-}\,\boldsymbol{e}^{k_c\boldsymbol{v}_i^T\boldsymbol{\mu}_j}}$, where k_c is set empirically to 6.

Semantic Tag-aware Contrastive Relation. We build our contrastive relation in the context of semantic tags. For a given semantic tag, we encode it as a one-hot vector. Note that a video with multiple actor-actions has multiple tags. In this case, the corresponding location of each tag in the one-hot vector is set to 1. This one-hot encoding makes it straightforward to compare semantic distance by checking whether there is overlapping in the same location. From the perspective of representation learning, for example, a frame/video with 'running and walking' should be separated from 'climbing', but attracted to 'running' and 'walking'. This tag-aware can be considered as an auxiliary classification task seamlessly embedded into training. Compared to individually pretrain a classification model [8], our design is more robust, since the auxiliary classification task is included in the learning, we are free from fine-tuning the pretrained network, leading to a better convergence.

Specifically, we denote $\mathcal{T}^+/\mathcal{T}^-$ as the positive/negative samples with/without overlapping tags of current pixel i. Let $\boldsymbol{v}_i,\boldsymbol{\mu}$ be the feature vector of pixel i and partition centroids, we calculate the tag-aware loss as the form of $\mathcal{L}_{\text{tag}}(i;\mathcal{T}^+,\mathcal{T}^-)\!=\!-\log\frac{\Sigma_{j\in\mathcal{T}^+}\,\boldsymbol{e}^{k_t\boldsymbol{v}_i^T\boldsymbol{\mu}_j}}{\Sigma_{j\in\mathcal{T}^+\cup\mathcal{T}^-}\,\boldsymbol{e}^{k_t\boldsymbol{v}_i^T\boldsymbol{\mu}_j}}$, where k_t is set empirically to 8 in our experiments.

3.3. Learning with Unlabeled Samples

Unlabeled samples refer to frames/videos with no class activation maps (CAMs), while samples with CAMs as pseudo-labeled. This unlabeled sample is not avoidable since we have *none* category in the dataset, meaning no action in the video. Note that those unlabeled samples still have edge segmentation and video-level tags since the edge information has no semantic content and is only the reflection of low-level visual representation. There are many studies [30, 40, 50] for learning from both labeled and unlabeled samples, but few methods investigate the weakly-supervised setting where the so-called labeled samples are, in fact, pseudo-labeled. Our intuition for including these samples is to learn a model that can distinguish multiple actions within a single frame/video using our designed contrastive relations instead of multi-action pseudo-annotations. Let $\mathbf{D}_c$ denote samples with CAM pseudo-segmentations, and its complementary set $\mathbf{D}_u$ is the samples without CAMs. We implement an indication mechanism that assigns an identity to each sample as

$$\begin{aligned}\mathbb{1}_{\mathbf{D}_c}(x)=1 \quad &\text{s.t.,} \quad x\in\mathbf{D}_c,\\ \mathbb{1}_{\mathbf{D}_c\cup\mathbf{D}_u}(x)=1 \quad &\text{s.t.,} \quad x\in\mathbf{D}_c\cup\mathbf{D}_u,\end{aligned} \tag{3}$$

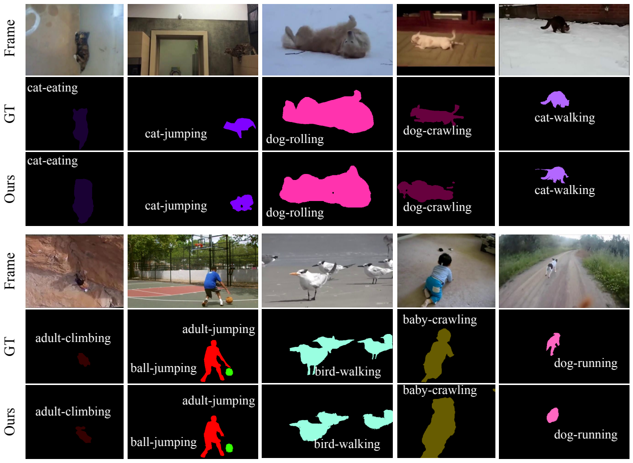


Figure 5. **Qualitative results on A2D dataset.** Every 3 rows show the frame, ground truth, and corresponding segmentation generated by our VSCR model.

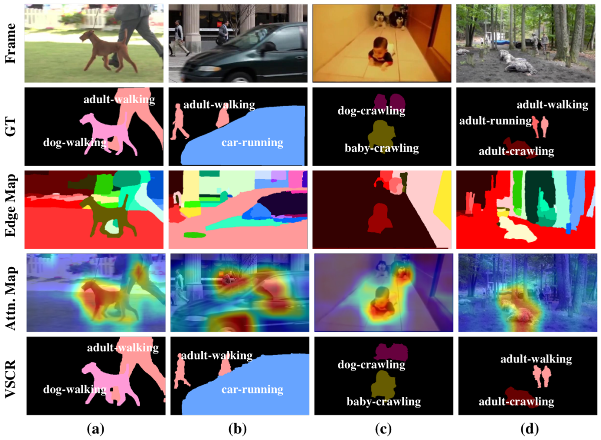


Figure 6. From top to bottom: input frame, ground-truth actor-action segmentation, edge segmentation, attention map, our result. We include different scenarios: (a) overlapping objects, (b) largely different sizes, (c) clear background, (d) complex background.

when the indicator returns 1, it means that the current index x is activated and participates in the loss backpropagation. Finally, we formulate our overall objective function as

$$\mathcal{L} = \mathbb{1}_{\mathbf{D}_c} \cdot \alpha_c \mathcal{L}_{\text{cam}} + \mathbb{1}_{\mathbf{D}_c \cup \mathbf{D}_u} \cdot (\alpha_e \mathcal{L}_{\text{edge}} + \alpha_t \mathcal{L}_{\text{tag}}), \quad (4)$$

where $\alpha_c/\alpha_e/\alpha_t$ are set to $0.3/0.3/0.1$, respectively. Our proposed learning objective loss function optimizes on both labeled and unlabeled samples. For the labeled part, all three supervisions are used where the CAM pseudo-annotation gives the basic pixel-wise information while the unlabeled part is not supervised by CAM annotation. We only utilize edge segmentations and semantic tags as the source of supervision.

Relation to Previous Contrastive Semantic Segmentation Work. We construct VSCR, akin to previous image-based weakly-/semi-supervised models [26, 33, 58], albeit with three differences. First, while they are targeting image-level segmentation, we are targeting segmentation in video with only access to the video-level tag. Second, motion cues are embedded into learning, whereas image-based models barely incorporate motion information. Third, different from heavily relying on pseudo-segmentations. In our case, unlabeled samples are also included during the training phase to address this annotation problem. Overall, our VSCR is different and specially designed for the weakly-supervised VAAS problem.

4. Experiments

We first present the details about the dataset and metrics. Then, we include the details for our pseudo-annotation generation, training, and inference. For experimental setup, we first compare our method – VSCR with other state-of-the-art fully- and weakly-supervised methods on the A2D dataset. Next, we decompose our model and comprehensively study the effectiveness of each component. Extensive experiments verify that VSCR outperforms video-level weakly-supervised methods and is even on par with several fully-supervised approaches.

4.1. Dataset and Metrics

Dataset. A2D dataset [54] is an actor-action segmentation dataset consisting of 3,782 videos with different resolutions and recording lengths, where the train/test split is 3,036/746. Unlike classic video object segmentation datasets [42], A2D is more challenging as it requires distinguishing actor and action simultaneously, *e.g.*, baby-crawling. It contains 7 actors and 9 actions in total, where there are multiple actors and multiple actions in some video clips. This actor-action ambiguity and unconstrained video quality contribute to the difficulties of A2D dataset.

Evaluation Metrics. Mean intersection over union (mIoU) is adopted to evaluate the model as a common practice in [8, 53]. Besides, we calculate the average per-class pixel accuracy (class_accuracy) and global pixel accuracy (global_accuracy) for quantitative evaluation under fully- and weakly-supervised settings on A2D dataset.

4.2. Implementation Details

Pseudo-Segmentation Generation. To identify the strong boundaries in the raw frames, we select HED contour detector [52] pretrained on the BSDS500 dataset [3]. We then perform hierarchical segmentation based on the edge map using gPb-owt-ucm [3]. The ucm threshold used in gPb-owt-ucm is 0.9. For pseudo-annotations (CAM [61]), we implement a Grad-CAM++ [6] with the backbone of ResNet-50 [21] pretrained on ImageNet [13] and then finetune the model on the A2D dataset. In total, 2,794

Method	mIou (actor-action/actor/action)
Xu *et al.* [53]	$19.9_{53.9\%}/33.4_{50.3\%}/32.0_{69.1\%}$
Kalogeiton *et al.* [31]	$29.7_{80.5\%}/49.5_{74.5\%}/42.2_{91.1\%}$
Qiu *et al.* [43]	$33.4_{90.5\%}/47.4_{71.4\%}/45.9_{99.1\%}$
Gavrilyuk *et al.* [18]	$34.8_{94.3\%}/53.7_{80.9\%}/\mathbf{49.4}_{106.7\%}$
Ji *et al.* [29]	$\mathbf{36.9}_{100\%}/\mathbf{66.4}_{100\%}/46.3_{100\%}$
Chen *et al.* [8]*	$26.7_{72.4\%}/49.2_{74.1\%}/38.7_{83.6\%}$
VSCR (Ours)	$\mathbf{29.6}_{80.2\%}/\mathbf{54.8}_{82.5\%}/\mathbf{40.2}_{86.8\%}$

Table 1. **Comparison to the state-of-the-art fully- and weakly-supervised methods on the A2D test set**. $\star$ is a weakly-supervised method while other methods are fully-supervised. The percentage denotes the performance compared to the best fully-supervised method [29].

Method	class_accuracy	global_accuracy
Trilayer *et al.* [54]	45.7/47.0/25.4	74.6/74.6/76.2
GPM+TSP [53]	58.3/60.5/43.3	85.2/85.3/84.2
GPM+GBH [53]	59.4/61.2/43.9	84.8/84.9/83.8
TSMT+GBH [31]	72.9/61.4/48.0	85.8/84.6/83.9
TSMT+SM [31]	73.7/60.5/47.5	90.6/89.3/88.7
Ji *et al.* [29]	**79.1/62.9/51.4**	**94.5/92.6/92.5**
Yan *et al.* [56]	41.7/ − /−	81.7/83.1/83.8
Chen *et al.* [8]	43.1/49.2/35.1	87.1/91.3/87.4
VSCR	**45.9/56.8/40.4**	**89.1/92.4/88.4**

Table 2. **Comparison to state-of-the-art fully-supervised and weakly-supervised methods on the A2D test set.**

video clips with single-action labels are used for finetuning. Our finetuned ResNet-50 achieves 86.21% accuracy on the single-action test set. We then apply the well-trained ResNet-50 with Grad-CAM++ to generate actor class activation maps. We use SLIC [1] to generate the initial pseudo-annotation and then apply the binarized class-activated map with a threshold of 0.5 to select appropriate regions with objects. It is worth noting that 3D action Grad-CAM++ is not used, which distinguishes our method from the previous method [8]. We achieve better performance using only the 2D image-level classification model, thanks to our proposed contrastive relations.

Training and Inference. We choose DeepLab [10] as the backbone of actor branch and I3D [5] pretrained on Kinetics-400 [32] as the backbone of action branch. While training, the inputs to the network are patches of 224 × 224 randomly cropped from images/videos. We set the learning rate to 8×10^{-4} with a poly learning rate policy of power 0.9 as suggested in [10] and train the model for 40,000 iterations. We adopt SGD as our optimizer with a momentum of 0.9, weight decay of 5×10^{-4}, and batch size of 10 frames. Corresponding to each input image to the actor branch, the video clip is input to the action branch to generate motion cues. The spherical Kmeans clustering iteration is 10 for each sample. During testing, we input the full-resolution frame to the model to generate full-size segmentation maps. The motion sequence is selected around the testing frames for accurate action representation. We also adopt a simple action alignment post-processing to unify the action label for the same actor where we assign the action label with the maximum votes to the actor of interest, as in [8, 48].

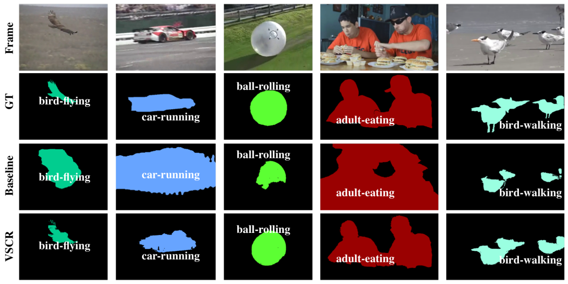


Figure 7. **Results of our VSCR and state-of-the-art method** [8].

4.3. Comparisons with State-of-the-Art Methods

We first show qualitative results under different scenarios in Figure 6. In addition to segmentation, we also show the edge segmentation and attention maps. It is clear that some edge segmentations preserve part of the interested objects (Figure 6(**a, c**)). This first verifies the effectiveness of using edge segmentation as our supervision and also proves that our trained model can correctly segment the objects under different scenarios. The attention maps show that our model can identify the interested regions to be segmented.

We then compare our weakly-supervised VSCR with the state-of-the-art fully- and weakly-supervised methods on the VAAS task, as shown in Table 1. It is worth noting that we achieve about 80% performance of the best fully-supervised model [29], compared to the best weakly-supervised method [8]. This also certifies the effectiveness of the motion cues, which the previous best weakly-supervised method fails to incorporate since they only use RGB frames as inputs, as well as our other proposed techniques, *i.e.*, contrastive relations and learning with unlabeled samples. In addition, we also compare with only two existing weakly-supervised methods [8, 56] as we are aware. For fair comparisons, following the evaluation metric used by them, we report our performance of the metrics in Table 2. It is clear that our model outperforms the current state-of-the-art weakly-supervised methods in terms of both class-accuracy and global-accuracy. We present qualitative comparisons in Figure 7. Compared to the baseline method, our generated segmentations exhibit better capture of the content with some fine-grained details as in the first sample. Our model accurately captures the overall shapes of different actors even when the actors are small in the context of the image size, *e.g.*, the adult climbing case. Overall, the actor-action segmentations generated by our VSCR model verify its effectiveness.

4.4. Ablation Study

Here, we conduct comprehensive experiments on those elements and analyze the results for each building block. In total, we derive five variants of our VSCR, *i.e.*, **(i) w/o edge segmentation** denotes the variant model trained without edge segmentation generated by [3, 52]; **(ii) w/o seman-**

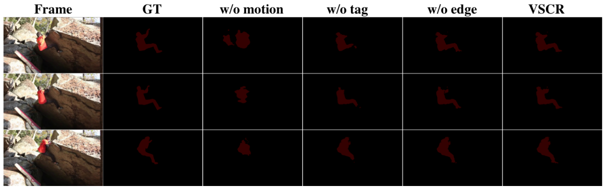


Figure 8. **Qualitative results for model ablations in frames.**

Model	mIou (Actor-Action)
w/o Edge Segmentation	28.2
w/o Semantic Tag	27.4
w/o Motion Cue	27.2
w/o Unlabeled Sample	28.1
VSCR (Ours)	29.6

Table 3. **Ablation study on the usage of contrastive relations, motion cues, and unlabeled samples.** Pseudo-labels provide the basic classification information which is required for the least functionality of VSCR so that they are included in all ablations.

tic tag as a variant without the supervision of semantic tags; **(iii) w/o motion cue** denotes that we only use RGB frames to train our model instead of using both frames and video sequence; **(iv) w/o unlabeled samples** represents the variant model that excludes unlabeled samples from training. **(v) VSCR** is the model with all our proposed modules.

Table 3 shows the ablation results in the model variants. Overall, the drop of motion cues mainly degrades the most performance of the proposed model. This performance falloff in turn verifies the necessity of including motion cues to learn a segmenter for joint actor-action segmentation. Besides, embedding motion cues into the model can be viewed as training an auxiliary classification subnetwork under the supervision of semantic action tags. This is also reflected by the performance when the semantic tags are removed from training, assuring the significance of high-level semantic guidance for training the model. Despite the performance degeneration, we can still observe the variants of our model outperform the current state-of-the-art method, which certifies the efficacy of our proposed model. We also show qualitative comparisons in Figure 8 of segmentation in a framed manner, confirming the contribution of each module. The model w/o *Unlabeled sample* is not included due to its similar performance to the model w/o edge segmenation. To sum up, each designed module contributes to the significant improvement of the proposed model, proving their effectiveness on the task of VAAS. To further study the robustness of the model, we conduct an empirical study on the concentration and loss weight hyper-parameters used in our proposed unified learning objective when training the proposed network. The empirical results are shown in Table 4.

In addition to the ablation study on model variants and hyper-parameters, we also design an ablative experiment on the influence of the number of pseudo-labeled samples used during the training phase. The results are shown in Table 5. It is worth noting that we turn the rest of the unused pseudo-labeled samples into unlabeled samples, *i.e.*, the pseudo-annotations of those unused samples are not used for training while their edge segmentations and semantic tags supervise them. The flexibility of this strategy is enabled by our proposed novel objective loss function Eq. (4). The results of this ablative study show that we can still train a reasonable segmentation model with fewer labeled samples, which is important to address the problem of annotation scarcity.

k_c	k_e	k_t	mIou
4	8	4	28.7
4	8	8	28.6
6	16	4	29.0
6	16	8	**29.6**
8	20	4	29.2
8	20	8	28.8

(a) Concentration ablation.

α_c	α_e	α_t	mIou
0.1	0.1	0.1	29.0
0.1	0.3	0.3	29.1
0.3	0.1	0.1	29.2
0.3	0.3	0.1	**29.6**
0.3	0.3	0.3	29.2
0.5	0.3	0.1	29.0

(b) Loss weights ablation.

Table 4. **Ablation study on concentration and loss weight parameters used in our unified objective for training.**

# Frames	Ratio	mIou (Actor-Action)
7,968	25%	27.4
15,936	50%	28.2
23,904	75%	29.6
31,872	100%	29.5

Table 5. **Ablation study on the portion of the pseudo-labeled samples (CAM pseudo-annotations) used in training.** When the ratio equals 100%, it means that we use all pseudo-labeled samples (single-action videos) for training purposes. Note that the samples are uniformly drawn for each video sequence to maintain the actor-action diversity.

5. Conclusion

This paper introduces a novel weakly-supervised framework, termed VSCR, designed to address the challenging problem of video actor-action segmentation. Our primary objective is to develop a segmentation model that achieves both visual plausibility and semantic consistency. The crux of our approach lies in extracting richer supervision signals from raw frames, breaking away from the conventional synthesize-refine pipeline seen in prior methods. We introduce a variety of contrastive relations that can be formulated into an optimized objective, thereby guiding the model towards generating enhanced segmentations within the context of VAAS. Moreover, we introduce two pivotal techniques: learning with motion cues to improve the model's ability to distinguish actions and incorporating unlabeled samples to transfer supervisory signals from pseudo-labeled samples to unlabeled ones. Our extensive evaluations demonstrate the effectiveness of our proposed approach, both qualitatively and quantitatively. Compared to existing methods, our framework significantly enhances the segmentation model's performance, establishing a new state-of-the-art for the weakly-supervised VAAS task.

Acknowledgements. This work is supported by NSF IIS-2309073 and NSF SCH-2123521. This article solely reflects the opinions of its authors and not the funding agency.

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *TPAMI*, 34(11):2274–2282, 2012. 4, 7

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, 2018. 5

[3] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *TPAMI*, 33(5):898–916, 2010. 5, 6, 7

[4] Arindam Banerjee, Inderjit S Dhillon, Joydeep Ghosh, Suvrit Sra, and Greg Ridgeway. Clustering on the unit hypersphere using von mises-fisher distributions. *JMLR*, 6(9), 2005. 4

[5] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 3, 7

[6] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *WACV*, 2018. 6

[7] Chun-Fu Richard Chen, Rameswar Panda, Kandan Ramakrishnan, Rogerio Feris, John Cohn, Aude Oliva, and Quanfu Fan. Deep analysis of cnn-based spatio-temporal representations for action recognition. In *CVPR*, 2021. 2

[8] Jie Chen, Zhiheng Li, Jiebo Luo, and Chenliang Xu. Learning a weakly-supervised video actor-action segmentation model with a wise selection. In *CVPR*, 2020. 1, 2, 3, 4, 5, 6, 7

[9] Liyi Chen, Weiwei Wu, Chenchen Fu, Xiao Han, and Yuntao Zhang. Weakly supervised semantic segmentation with boundary exploration. In *ECCV*, 2020. 5

[10] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 40(4):834–848, 2017. 3, 7

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2, 3

[12] Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. In *NeurIPS*, 2020. 2

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 6

[14] Ali Diba, Vivek Sharma, Reza Safdari, Dariush Lotfi, Saquib Sarfraz, Rainer Stiefelhagen, and Luc Van Gool. Vi2clr: Video and image for visual contrastive learning of representation. In *ICCV*, 2021. 2

[15] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *ICCV*, 2021. 2

[16] Junsong Fan, Zhaoxiang Zhang, Chunfeng Song, and Tieniu Tan. Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation. In *CVPR*, 2020. 5

[17] Junsong Fan, Zhaoxiang Zhang, and Tieniu Tan. Employing multi-estimations for weakly-supervised semantic segmentation. In *ECCV*, 2020. 5

[18] Kirill Gavrilyuk, Amir Ghodrati, Zhenyang Li, and Cees GM Snoek. Actor and action video segmentation from a sentence. In *CVPR*, 2018. 1, 2, 7

[19] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 2, 3

[20] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 2

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6

[22] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *ICML*, 2020. 2, 3

[23] Minh Hoai, Zhen-Zhong Lan, and Fernando De la Torre. Joint segmentation and classification of human actions in video. In *CVPR*, 2011. 2

[24] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. 2

[25] Seunghoon Hong, Donghun Yeo, Suha Kwak, Honglak Lee, and Bohyung Han. Weakly supervised semantic segmentation using web-crawled videos. In *CVPR*, 2017. 1

[26] Jyh-Jing Hwang, Stella X Yu, Jianbo Shi, Maxwell D Collins, Tien-Ju Yang, Xiao Zhang, and Liang-Chieh Chen. Segsort: Segmentation by discriminative sorting of segments. In *ICCV*, 2019. 3, 4, 5, 6

[27] Suyog Dutt Jain and Kristen Grauman. Supervoxel-consistent foreground propagation in video. In *ECCV*, 2014. 1

[28] Ge-Peng Ji, Keren Fu, Zhe Wu, Deng-Ping Fan, Jianbing Shen, and Ling Shao. Full-duplex strategy for video object segmentation. In *ICCV*, 2021. 2

[29] Jingwei Ji, Shyamal Buch, Alvaro Soto, and Juan Carlos Niebles. End-to-end joint semantic segmentation of actors and actions in video. In *ECCV*, 2018. 1, 2, 4, 7

[30] Tarun Kalluri, Girish Varma, Manmohan Chandraker, and CV Jawahar. Universal semi-supervised semantic segmentation. In *ICCV*, 2019. 5

[31] Vicky Kalogeiton, Philippe Weinzaepfel, Vittorio Ferrari, and Cordelia Schmid. Joint learning of object and action detectors. In *ICCV*, 2017. 1, 2, 4, 7

[32] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint*, 2017. 7

[33] Tsung-Wei Ke, Jyh-Jing Hwang, and Stella X Yu. Universal weakly supervised segmentation by pixel-to-segment contrastive learning. In *ICLR*, 2021. 3, 4, 5, 6

[34] Haofei Kuang, Yi Zhu, Zhi Zhang, Xinyu Li, Joseph Tighe, Sören Schwertfeger, Cyrill Stachniss, and Mu Li. Video contrastive learning with global context. In *ICCV*, 2021. 2

[35] Xin Lai, Zhuotao Tian, Li Jiang, Shu Liu, Hengshuang Zhao, Liwei Wang, and Jiaya Jia. Semi-supervised semantic segmentation with directional context-aware consistency. In *CVPR*, 2021. 2

[36] Yong Jae Lee, Jaechul Kim, and Kristen Grauman. Key-segments for video object segmentation. In *ICCV*, 2011. 2

[37] Shuang Li, Yilun Du, Antonio Torralba, Josef Sivic, and Bryan Russell. Weakly supervised human-object interaction detection in video via contrastive spatiotemporal regions. In *ICCV*, 2021. 2

[38] Simon Niklaus. A reimplementation of HED using PyTorch. `https://github.com/sniklaus/pytorch-hed`, 2018. 5

[39] Yassine Ouali, Céline Hudelot, and Myriam Tami. Autoregressive unsupervised image segmentation. In *ECCV*, 2020. 4

[40] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *CVPR*, 2020. 5

[41] Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *ICCV*, 2013. 2

[42] Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012. 6

[43] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning deep spatio-temporal dependence for semantic video segmentation. *TMM*, 20(4):939–949, 2017. 1, 2, 7

[44] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017. 1

[45] Kunyang Sun, Haoqing Shi, Zhengming Zhang, and Yongming Huang. Ecs-net: Improving weakly supervised semantic segmentation by using connections between class activation maps. In *ICCV*, 2021. 5

[46] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *CVPR*, 2017. 1, 2

[47] Limin Wang, Zhan Tong, Bin Ji, and Gangshan Wu. Tdn: Temporal difference networks for efficient action recognition. In *CVPR*, 2021. 2

[48] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 7

[49] Xiaohan Wang, Linchao Zhu, Heng Wang, and Yi Yang. Interactive prototype learning for egocentric action recognition. In *ICCV*, 2021. 2

[50] Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S Huang. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *CVPR*, 2018. 5

[51] Tong Wu, Junshi Huang, Guangyu Gao, Xiaoming Wei, Xiaolin Wei, Xuan Luo, and Chi Harold Liu. Embedded discriminative attention mechanism for weakly supervised semantic segmentation. In *CVPR*, 2021. 5

[52] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *ICCV*, 2015. 5, 6, 7

[53] Chenliang Xu and Jason J Corso. Actor-action semantic segmentation with grouping process models. In *CVPR*, 2016. 1, 2, 6, 7

[54] Chenliang Xu, Shao-Hang Hsieh, Caiming Xiong, and Jason J Corso. Can humans fly? action understanding with multiple classes of actors. In *CVPR*, 2015. 2, 6, 7

[55] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, Ferdous Sohel, and Dan Xu. Leveraging auxiliary tasks with affinity learning for weakly supervised semantic segmentation. In *ICCV*, 2021. 5

[56] Yan Yan, Chenliang Xu, Dawen Cai, and Jason J Corso. Weakly supervised actor-action segmentation via robust multi-task ranking. In *CVPR*, 2017. 1, 2, 7

[57] Fei Zhang, Chaochen Gu, Chenyue Zhang, and Yuchao Dai. Complementary patch for weakly supervised semantic segmentation. In *ICCV*, 2021. 5

[58] Xiao Zhang and Michael Maire. Self-supervised visual representation learning from hierarchical grouping. In *NeurIPS*, 2020. 3, 4, 5, 6

[59] Yongqiang Zhang, Yancheng Bai, Mingli Ding, Yongqiang Li, and Bernard Ghanem. W2f: A weakly-supervised to fully-supervised framework for object detection. In *CVPR*, 2018. 1

[60] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *ICCV*, 2017. 2

[61] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 6

[62] Roland S Zimmermann, Yash Sharma, Steffen Schneider, Matthias Bethge, and Wieland Brendel. Contrastive learning inverts the data generating process. In *ICML*, 2021. 2