Multi-Level Representation Learning with Semantic Alignment for Referring Video Object Segmentation

Dongming Wu[1,2∗‡], Xingping Dong[2∗], Ling Shao[3], Jianbing Shen[4†]
[1] Beijing Institute of Technology, [2] Inception Institute of Artificial Intelligence,
[3] Terminus Group, China, [4] SKL-IOTSC, University of Macau
wudongming@bit.edu.cn, {xingping.dong, shenjiangbingcg}@gmail.com, ling.shao@ieee.org

Abstract

Referring video object segmentation (RVOS) is a challenging language-guided video grounding task, which requires comprehensively understanding the semantic information of both video content and language queries for object prediction. However, existing methods adopt multi-modal fusion at a frame-based spatial granularity. The limitation of visual representation is prone to causing vision-language mismatching and producing poor segmentation results. To address this, we propose a novel multi-level representation learning approach, which explores the inherent structure of the video content to provide a set of discriminative visual embedding, enabling more effective vision-language semantic alignment. Specifically, we embed different visual cues in terms of visual granularity, including multi-frame long-temporal information at video level, intra-frame spatial semantics at frame level, and enhanced object-aware feature prior at object level. With the powerful multi-level visual embedding and carefully-designed dynamic alignment, our model can generate a robust representation for accurate video object segmentation. Extensive experiments on Refer-DAVIS$_{17}$ and Refer-YouTube-VOS demonstrate that our model achieves superior performance both in segmentation accuracy and inference speed.

1. Introduction

Given a natural language expression, referring video object segmentation (RVOS) aims to predict the most relevant visual target from a video. It has wide applications, including video editing, virtual reality and human-robot interaction [49]. Different from the regular unsupervised or semi-supervised video object segmentation (VOS) [12, 21, 33, 53, 54], which localizes objects with salience or annotations of key frames, RVOS requires cross-modal understanding between the language query and video content.

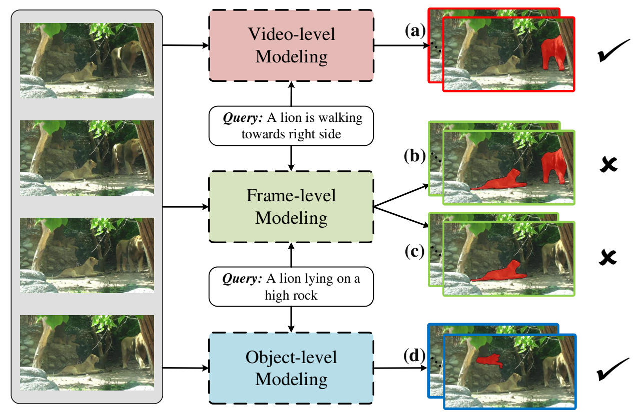

Figure 1. **Visual comparison among the different-level modelings.** The simple frame-level modeling has difficulty in recognizing (b) the moving object or (c) the occluded and small object. In contrast, our multi-level modeling offers a joint way to leverage the long-temporal and spatial salient cues for cross-modal alignment, thus providing more accurate results (a) (d).

As a human recognizes a referent object with the guidance of language, it is natural to rely on three steps: 1) observe its appearance (*i.e.*, frame-based), 2) check its movement based on multiple frames (*i.e.*, video-based), 3) shift more attention to the occluded or small objects (*i.e.*, object-based). Most current approaches [1, 25, 43] simply leverage successful referring image comprehension methods to the cross-model understanding. They either use referring image grounding [24, 31, 58, 60] to generate target object bounding boxes as proposals, or utilize referring image segmentation directly [6, 10, 18, 22, 27, 56]. However, these solutions build on the simple frame-level visual representation to perform frame-sentence interaction. These frame-level modeling methods suffer from two limitations compared to the human recognition system: ignoring long-temporal information and lacking attention to salient spatial objects.

The limitation of visual representations causes the mis-

∗Equal contribution. †Corresponding author: *Jianbing Shen*. ‡Work was done while Dongming Wu was an intern at IIAI.

alignment between two modalities, further producing inaccurate segmentation results. For example, as illustrated in Fig. 1, given an input video and its corresponding description, "a lion is walking towards right side", RVOS aims to segment the moving lion from the video. However, as there are multiple lions in the video, the frame-level modeling cannot recognize the correct one by employing only spatial appearance information as shown in Fig. 1(b). Since the referent object has a temporally moving status, it requires incorporating long-temporal information from multiple frames to identify the action. In addition, another expression, "a lion lying on a high rock" refers to an occluded and small-size lion. However, the frame-level modeling focuses only on the global semantics concerning each frame, and ignores these important and representative visual regions. It will lead to the referent object being missing, as shown in Fig. 1(c). To ease this difficulty, it is also necessary to capture the salient spatial objects from each frame as candidates to facilitate cross-modal understanding.

In this paper, we propose a novel multi-level learning framework for addressing RVOS. The model first presents a fine-grained analysis of video content for multi-granularity visual embedding:

- At the *video* granularity, we propose to model long-temporal dependencies of the entire video using a cross-frame pixel-wise calculator, which makes the feature representations to capture the object movement and dynamic scenes information.
- At the *frame* granularity, we encourage the frame representation to describe global content within a whole image, by learning to aggregate intra-frame information following the self-attention mechanism.
- At the *object* granularity, we leverage object-aware information generated from an *object detector* to enhance the foreground and background discriminability, benefiting from addressing the cases of occlusion and small object.

Once we obtain the multi-level visual embedding, we propose Dynamic Semantic Alignment (DSA) to interact them with the linguistic features. In particular, to effectively capture the granularity-specific information, we first separately incorporate global linguistic semantics according to the different visual cues. The generated vision-conditioned linguistic features are combined with the corresponding visual embedding to provide a granularity-specific representation for the referent object. Finally, we integrate the multi-level target-aware features and boundary information to guide the mask prediction of all frames using a Boundary-Aware Segmentation (BAS).

Overall, our contributions are summarized as three-fold: **First**, we propose a new framework for RVOS based on multi-level representation learning. It precludes the limitation of single frame-level visual modeling by a more structural video representation, promoting accurate vision-language semantic alignment. **Second**, we introduce a Dynamic Semantic Alignment (DSA), which dynamically learns and matches linguistic semantics with the different-granularity visual representation more compactly and effectively. **Third**, our approach achieves compelling performance on two challenging benchmarks, including Refer-DAVIS$_{17}$ [25] and Refer-YouTube-VOS [43]. Notably, we obtain a significant improvement of 6.6% than the best frame-grained method in terms of $\mathcal{J}$ on Refer-DAVIS$_{17}$. Meanwhile, it achieves a high inference speed at 53.2 FPS.

2. Related Work

2.1. Referring Video Object Segmentation

The goal of referring video object segmentation (RVOS) is to localize the entities in a video that are matched with the description of a natural language expression. Khoreva *et al*. [25] introduce a two-stage method, the first stage to generate bounding boxes in image [58, 60] and the second one to segment the referent object from video [20, 40]. Seo *et al*. [43] extend YouTube-VOS [54] into a new and large-scale benchmark, named Refer-YouTube-VOS. Meanwhile, they propose an end-to-end framework by unifying cross-modal attention module [56] and space-time memory network [38]. Recently, RefVOS [1] employs the fine-grained categorization of expressions to overcome the overfitting. However, their frame-sentence interaction mechanism lacks the long-temporal and fine-grained visual representations, further resulting in the cross-modal misalignment as discussed before. Although a large number of works on actor and action video segmentation [11, 19, 34, 45, 46, 57] also study the problem of language-queried video segmentation, their descriptions are limited into the format of 'actors' performing a salient 'action'. The newly appearing RVOS shows improved difficulties in both visual and linguistic modalities. Thus, our method can be regarded as a more generalized work to handle real-life situations.

2.2. Multi-Level Representation Learning

Multi-level representation learning is a common concept in feature embedding, including natural language processing [9, 14, 32] and computer vision [2, 7, 8, 13, 17, 52, 61]. The language processing usually cooperates with the word-phrase-sentence composition semantics to enrich word embedding, while the visual tasks focus on exploiting spatial or temporal granularity to learn a robust and powerful visual feature representation [29, 30, 47, 48]. For the video understanding task, the most popular granularity analysis is built on the temporal order [16, 17, 28, 49]. For instance, Hu *et al*. [17] associate different sub-networks to leverage the inherent temporal continuity of previous frames for fast video semantic segmentation. Lu *et al*. [30] summarizes the frame-term, short-term, long-term and global features of

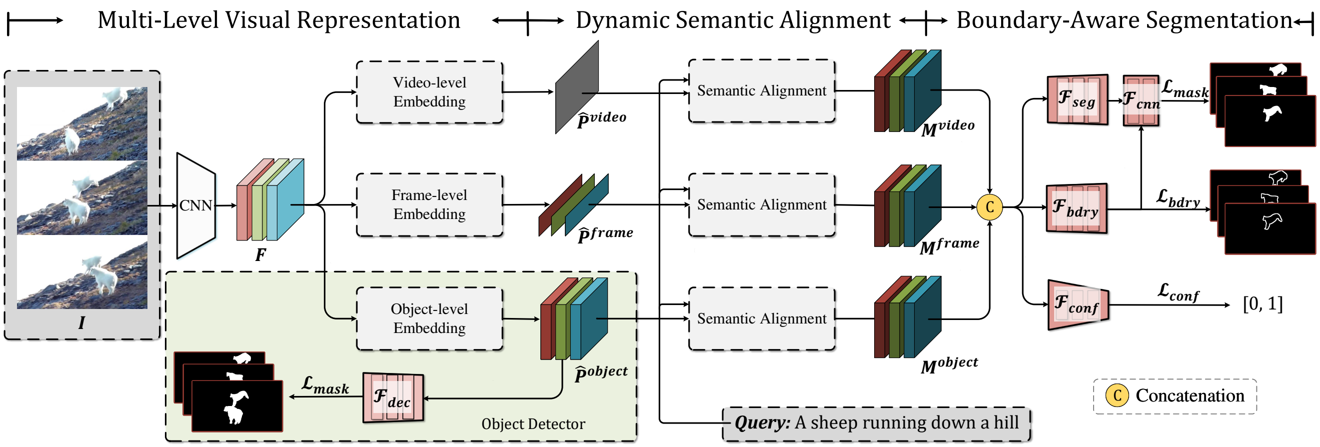

Figure 2. **Illustration of our multi-level representation learning with semantic alignment,** which consists of three major components, a multi-level visual representation for embedding different-level visual features (§3.1), a dynamic semantic alignment for matching vision-language modalities (§3.2), and a boundary-aware segmentation for outputting pixel-wise masks of the target (§3.3).

each video for robust unsupervised video object segmentation. However, these methods are limited in visual modeling and cannot deal with the crucial cross-modal understanding. Recently, several language-queried tasks [23,36,50,59] address these drawbacks and achieve promising object or moment localization via global-local video-language alignment, but they are not suitable for fine-grained object segmentation. This paper proposes a new perspective of exploring multi-level video information and cross-modal semantic alignment for precise mask prediction.

3. Methodology

Given a video clip and a natural language query, the goal of our approach is to automatically generate a set of referent object masks. We illustrate the overall pipeline in Fig. 2. The multi-level visual representation first separately embeds the CNN-encoded features at video, frame, and object level, which provides three enhanced visual representations (§3.1). The specific visual representation and the linguistic embedding are then fed into our dynamic semantic alignment to jointly highlight the visual features of interest (§3.2). Finally, the boundary-aware segmentation integrates the target-aware features and boundary information to guide the referent object prediction (§3.3). In the following, we will introduce them carefully.

3.1. Multi-Level Visual Representation

Before learning multi-level visual representations, we first extract frame-wise video features for a given video clip. The T-frame video $\boldsymbol{I} \in \mathbb{R}^{T\times 3\times H\times W}$ is fed into the ResNet-50 [15] to obtain the `res5` features $\boldsymbol{F} \in \mathbb{R}^{T\times c\times h\times w}$, where c, h, w represent the channel, height, weight number of the 3D tensors, respectively. Furthermore, a 1×1 convolution is used to reduce the channel dimension from c to a smaller d ($d << c$), as well as keep the same dimension with linguistic features in §3.2. The transformed video features are then fed into our multi-level visual representation module to embed different types of visual cues.

The multi-level visual representation consists of three independent embedding modules, 1) a video-level embedding to describe the global and long-temporal statistics of the entire video, 2) a frame-level embedding to learn the intra-frame long-distance semantic context, 3) an object-level embedding to highlight the object-aware features.

Video-level Embedding. Inspired by the recent success of visual transformer [3, 51], we take advantage of the core long-distance modeling ability of self-attention to formulate our video-level embedding module. It handles all video frames in a unified manner and models the pixel-level pairwise relation directly. Specifically, we flatten the entire video features into a 2D pixel-wise sequence $\boldsymbol{P} \in \mathbb{R}^{Thw\times d}$. Three different fully-connected layers $\boldsymbol{W}_Q, \boldsymbol{W}_K, \boldsymbol{W}_V$ are used to transform the sequence as $\boldsymbol{Q}^{video}, \boldsymbol{K}^{video}, \boldsymbol{V}^{video}$:

$$\begin{aligned} \boldsymbol{Q}^{video} &= \boldsymbol{W}_Q\boldsymbol{P} \in \mathbb{R}^{Thw\times d}, \\ \boldsymbol{K}^{video} &= \boldsymbol{W}_K\boldsymbol{P} \in \mathbb{R}^{Thw\times d}, \\ \boldsymbol{V}^{video} &= \boldsymbol{W}_V\boldsymbol{P} \in \mathbb{R}^{Thw\times d}. \end{aligned} \tag{1}$$

Then we calculate a similarity matrix $\boldsymbol{A}^{video}$ with pairwise dot product and normalize it with softmax,

$$\boldsymbol{A}^{video} = \mathrm{Softmax}(\frac{\boldsymbol{Q}^{video}\boldsymbol{K}^{video\top}}{\sqrt{d}}) \in \mathbb{R}^{Thw\times Thw}, \tag{2}$$

where $\boldsymbol{A}^{video}$ measures the relevance between each pixel in the video. These video sequence vectors are weighted according to the relevance and added with the original $\boldsymbol{P}$:

$$\hat{\boldsymbol{P}}^{video} = \boldsymbol{A}^{video}\boldsymbol{V}^{video} + \boldsymbol{P} \in \mathbb{R}^{Thw\times d}, \tag{3}$$

where $\hat{\boldsymbol{P}}^{video}$ is the video-level feature embedding. It models multi-frame information and represents holistic understanding of the video.

Frame-level Embedding. To learn frame-level feature embedding, following self-attention mechanism, we build the spatial pixel-wise relationship for each frame. Unlike prior work [43] that processes each pixel, our method is used on each frame independently. It maps each frame feature into 2D tensor $\boldsymbol{P}_t \in \mathbb{R}^{hw \times d}$ $(t = 1, ..., T)$, applies linear transformation to generate $\boldsymbol{Q}_t^{frame}$, $\boldsymbol{K}_t^{frame}$, $\boldsymbol{V}_t^{frame}$, and performs weighting operation using learnable attention:

$$\hat{\boldsymbol{P}}_t^{frame} = \text{Softmax}(\frac{\boldsymbol{Q}_t^{frame} \boldsymbol{K}_t^{frame\top}}{\sqrt{d}}) \boldsymbol{V}_t^{frame} + \boldsymbol{P}_t, \quad (4)$$

where $\hat{\boldsymbol{P}}_t^{frame} \in \mathbb{R}^{hw \times d}$ represents the feature embedding of the t^{th} frame.

Object-level Embedding. In addition to learning the global semantics for video and image, we also conduct object-level feature embedding to capture salient spatial information. This can be viewed an *object detection* process, which includes two sequential parts, an *object encoder* for object-aware feature extraction, and a *segmentation decoder* for salient object generation.

Let $\mathcal{F}_{enc}$ denote the object encoder, which accepts the original video features $\boldsymbol{F}$ as input, and directly outputs the object-level embedding $\hat{\boldsymbol{P}}^{object}$:

$$\hat{\boldsymbol{P}}^{object} = \mathcal{F}_{enc}(\boldsymbol{F}). \quad (5)$$

After that, we implement the segmentation decoder $\mathcal{F}_{dec}$ to generate all salient objects:

$$\boldsymbol{Y}^{object} = \mathcal{F}_{dec}(\hat{\boldsymbol{P}}^{object}) \in \mathbb{R}^{T \times 1 \times H \times W}, \quad (6)$$

where $\boldsymbol{Y}^{object}$ are one-channel feature maps for all frames, which are activated using a sigmoid function, and supervised by the object-level ground-truth $\hat{\boldsymbol{Y}}^{object}$:

$$\mathcal{L}_{object} = \mathcal{L}_{mask}(\boldsymbol{Y}^{object}, \hat{\boldsymbol{Y}}^{object}). \quad (7)$$

Here, with the encouragement of object-level loss $\mathcal{L}_{object}$, the object encoder can highlight the object-sensitive features to serve as the object-level embedding $\hat{\boldsymbol{P}}^{object}$. The mask loss $\mathcal{L}_{mask}$ is the summation of Dice loss $\mathcal{L}_{dice}$ [35] and focal loss $\mathcal{L}_{focal}$ [26], *i.e.*, $\mathcal{L}_{mask} = \mathcal{L}_{dice} + \mathcal{L}_{focal}$.

The object encoder $\mathcal{F}_{enc}$ can use various feature embedding models, such as fully-convolutional network (FCN), video-level encoder and frame-level encoder as aforementioned. Empirically, we choose the video-level encoder with a 3×3 convolution to be our object encoder according to the experiments in §4.4. The segmentation decoder $\mathcal{F}_{dec}$ is built on fully-convolutional network that is similar to the pyramid segmentation head in §3.3. To sum up, the joint multi-grained learning provides an enhanced and informative visual representation, which will facilitate the following vision-language semantic alignment.

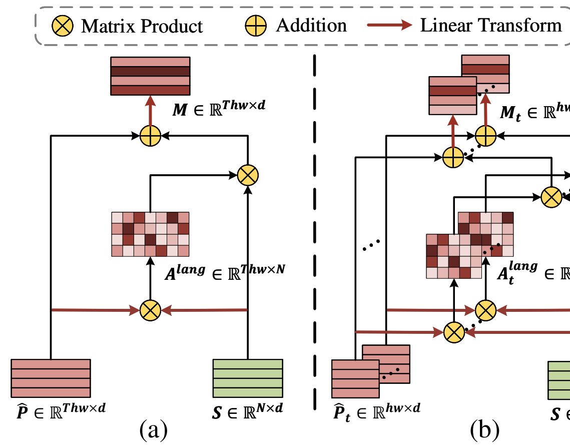

Figure 3. **Two solutions of semantic alignment:** (a) video-level alignment for global and long-temporal alignment, and (b) frame-level alignment for spatial alignment.

3.2. Dynamic Semantic Alignment

Given the different-level visual embedding representations $\hat{\boldsymbol{P}}^{video}$, $\{\hat{\boldsymbol{P}}_t^{frame}\}_{t=1}^T$, $\hat{\boldsymbol{P}}^{object}$ as well as its corresponding language description, the goal of DSA is to enable two modalities interact with each other for characterizing the referent object representation. To dynamically learn the global linguistic semantics that have the most relationship with each visual granularity, we individually embed three linguistic representations $\boldsymbol{S}^{video}$, $\boldsymbol{S}^{frame}$, $\boldsymbol{S}^{object}$. Taking the video branch as an example, for the input language query with N words, we follow the work [56] to encode each word into a feature vector. A transformer encoder [44] is trained to extract the specific linguistic features, which are denoted as $\boldsymbol{S}^{video} \in \mathbb{R}^{N \times d}$ where d is the feature dimension. The same operation is also applied on frame and object branch to obtain $\boldsymbol{S}^{frame}$ and $\boldsymbol{S}^{object}$.

DSA includes two kinds of solutions in terms of the interaction level, *i.e.*, video-level alignment and frame-level alignment, as depicted in Fig. 3. The video-level semantic alignment $\mathcal{F}_{video}$ (used at video granularity) takes the temporal information across two modalities to be aligned, while the frame-level alignment $\mathcal{F}_{frame}$ (used at frame and object granularity) is responsible for spatial alignment:

$$\begin{aligned} \boldsymbol{M}^{video} &= \mathcal{F}_{video}(\hat{\boldsymbol{P}}^{video}, \boldsymbol{S}^{video}), \\ \boldsymbol{M}_t^{frame} &= \mathcal{F}_{frame}(\hat{\boldsymbol{P}}_t^{frame}, \boldsymbol{S}^{frame}), \\ \boldsymbol{M}_t^{object} &= \mathcal{F}_{frame}(\hat{\boldsymbol{P}}_t^{object}, \boldsymbol{S}^{object}), \end{aligned} \quad (8)$$

where $\boldsymbol{M}^{video}$, $\boldsymbol{M}_t^{frame}$, and $\boldsymbol{M}_t^{object}$ are cross-modal features. Both $\mathcal{F}_{video}$ and $\mathcal{F}_{frame}$ have a standard semantic alignment architecture as presented in the following.

Semantic Alignment. For the convenience of description, we omit the granularity superscript and frame subscript of visual features and reshape them as $\hat{\boldsymbol{P}} \in \mathbb{R}^{(T)hw \times d}$. We add the position embedding, as proposed in [44, 51], into the visual and linguistic features to keep the coordinate alignment and employ linear layers to transform them:

$$\begin{aligned}\hat{\boldsymbol{P}}' &= \text{Linear}(\hat{\boldsymbol{P}} + \text{POS}^V) \in \mathbb{R}^{(T)hw \times d},\\ \boldsymbol{S}' &= \text{Linear}(\boldsymbol{S} + \text{POS}^L) \ \in \mathbb{R}^{N \times d},\end{aligned} \tag{9}$$

where POS^V and POS^L are the visual and linguistic position, respectively. The transformed visual and linguistic features are further calculated through matrix product and softmax normalization:

$$\boldsymbol{A}^{lang} = \text{Softmax}(\frac{\hat{\boldsymbol{P}}'\boldsymbol{S}'^{\top}}{\sqrt{d}}) \in \mathbb{R}^{(T)hw \times N}. \tag{10}$$

Here attention map $\boldsymbol{A}^{lang}$ represents the similarity between each word and each location of the visual representation. Next, the granularity-specific linguistic features are summarized as $\hat{\boldsymbol{S}} = \boldsymbol{A}^{lang}\boldsymbol{S} \in \mathbb{R}^{(T)hw \times d}$, and are added into original $\hat{\boldsymbol{P}}$ to automatically align two features:

$$\boldsymbol{M} = \text{Linear}(\boldsymbol{A}^{lang}\boldsymbol{S} + \hat{\boldsymbol{P}}) \in \mathbb{R}^{(T)hw \times d}, \tag{11}$$

where $\boldsymbol{M}$ represent the activated target-aware features after semantic alignment. We recover their level superscript and the format of video, *i.e.*, $\boldsymbol{M}^{video}$, $\boldsymbol{M}^{frame}$, $\boldsymbol{M}^{object} \in \mathbb{R}^{Thw \times d}$. Their size is reshaped as $T \times d \times h \times w$, and we concatenate them along channel dimension, *i.e.*, $\boldsymbol{M} = [\boldsymbol{M}^{video}, \boldsymbol{M}^{frame}, \boldsymbol{M}^{object}] \in \mathbb{R}^{T \times 3d \times h \times w}$ for following mask estimation.

3.3. Boundary-Aware Segmentation

The BAS aims to produce pixel-wise masks using rich target-aware and boundary-aware information. It first generates a one-channel boundary map $\boldsymbol{B}$ by accepting the modulated target-aware features $\boldsymbol{M}$ and original visual features $\boldsymbol{F}$ as input:

$$\boldsymbol{B} = \mathcal{F}_{bdry}\left(\boldsymbol{M}, \boldsymbol{F}^{[2,3,4,5]}\right) \in \mathbb{R}^{T \times 1 \times H \times W}, \tag{12}$$

where $\boldsymbol{F}^{[2,3,4,5]}$ is a simplified feature denotation from different backbone layers (`Res2`, `Res3`, `Res4`, `Res5`). The boundary head $\mathcal{F}_{bdry}$ and segmentation head $\mathcal{F}_{seg}$ have the same pyramid architecture by inserting the different-scale original features, as like the pyramid decoder of [43]. The outputs from two heads are concatenated together to estimate finer object masks $\boldsymbol{E}$:

$$\boldsymbol{E} = \mathcal{F}_{cnn}\left(\boldsymbol{B}, \mathcal{F}_{seg}\left(\boldsymbol{M}, \boldsymbol{F}^{[2,3,4,5]}\right)\right) \in \mathbb{R}^{T \times 1 \times H \times W}, \tag{13}$$

where $\mathcal{F}_{cnn}$ includes a superficial 3x3 convolutional layer. The adopted instance-level loss combines the mask and boundary supervision:

$$\mathcal{L}_{instance} = \mathcal{L}_{mask}(\boldsymbol{B}, \hat{\boldsymbol{Y}}) + \alpha\mathcal{L}_{bdry}(\boldsymbol{E}, \hat{\boldsymbol{Y}}^{bdry}), \tag{14}$$

where $\hat{\boldsymbol{Y}}$, $\hat{\boldsymbol{Y}}^{bdry}$ represent the ground-truth of $\boldsymbol{B}$ and $\boldsymbol{E}$. $\mathcal{L}_{mask}$ is the mask loss as mentioned in §3.1. $\mathcal{L}_{bdry}$ is the boundary loss following [42] and α is a hyper-parameter. The overall objective is the summation of object-level loss (Eq. 7) and instance-level loss (Eq. 14):

$$\mathcal{L} = \mathcal{L}_{object} + \mathcal{L}_{instance}. \tag{15}$$

3.4. Implementation Details

Network. The backbone model adopted in our approach is ResNet-50 [15], which is pretrained on ImageNet [5]. We only use the feature maps from the last layer for visual embedding and semantic alignment, while BAS accepts the feature pyramids of the backbone model for FPN-like coarse-to-fine segmentation. In BAS, the mapping block between two levels consists of a 3×3 convolution, a group normalization (8 groups) and a bilinear upsampling layer. The final one-channel feature maps of $\boldsymbol{B}$ and $\boldsymbol{E}$ are activated using sigmoid for training and inference.

Training. The input video has $T = 12$ frames with the size of 432×240. The language length is $N = 20$, and the feature dimension is set to $d = 384$. The object annotations (Eq. 7) can be obtained by combining all instance-level labels. We calculate the boundary annotations (Eq. 14) following the work of [42]. The hyper-parameter α is 0.2. Our model is implemented on *PyTorch* [39] and trained on four NVIDIA Tesla V100 GPUs with 32GB memory per card. We optimize the overall model with AdaW optimizer using learning rate $1e^{-4}$ for backbone, $1e^{-5}$ for the remaining part. The batch size is set to 2. Note that we predict confidence scores $\boldsymbol{C} = \mathcal{F}_{conf}(\boldsymbol{M}) \in \mathbb{R}^{T \times 1}$ for all frames with an extra confidence estimation head $\mathcal{F}_{conf}$, as shown in Fig. 2. Therefore, we build a new overall objective:

$$\mathcal{L} = \mathcal{L}_{object} + \mathcal{L}_{instance} + \beta\mathcal{L}_{conf}(\boldsymbol{C}, \text{IoU}(\boldsymbol{Y}, \hat{\boldsymbol{Y}}), \tag{16}$$

where IoU indicates the IoU calculation operation, and $\mathcal{L}_{conf}$ is L_2 loss. $\beta = 0.1$ serves as a hyperparameter. $\mathcal{F}_{conf}$ contains a global average pooling and three fully-convolution layers with the final layer outputting T scores.

Inference. During inference, we also exploit the recent VOS method, STCN [4] to improve the cross-frame object consistency as well as refine the segmentation results as a post-processing strategy. STCN propagates the highest-confidence mask in a bi-directional way to obtain the final segmentation masks for evaluation. We regard the output feature maps whose sigmoid activation value is higher than 0.5 as binary results.

4. Experiments

4.1. Experimental Setup

Datasets. We conduct experiments on two popular RVOS benchmarks, *i.e.*, Refer-DAVIS$_{17}$ [25] and Refer-YouTube-VOS [43]. Refer-DAVIS$_{17}$ expands DAVIS$_{17}$ [41] by annotating the objects of video with more than 1,500 refer-

Method	Pretrained	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$
Khoreava *et.al.* [25]	RefCOCO [37]	37.3	41.3	39.3
URVOS [43]	RefCOCO [37]	41.2	47.0	44.1
RefVOS [1]	RefCOCO [37]	–	–	45.1
Ours	RefCOCO [37]	45.1	51.2	48.2
Baseline (frame-based) [56]	Refer-YouTube-VOS	32.19	37.23	34.71
Baseline + RNN [56]	Refer-YouTube-VOS	36.94	43.45	40.20
URVOS (pretraining only) [43]	Refer-YouTube-VOS	44.29	49 41	46.85
URVOS [43]	Refer-YouTube-VOS	47.29	55.96	51.63
Ours (pretraining only)	Refer-YouTube-VOS	50.07	55.39	52.73
Ours	Refer-YouTube-VOS	**53.85**	**62.02**	**57.94**

Table 1. **The quantitative evaluation on Refer-DAVIS$_{17}$ *val* set**, with region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$, and average of $\mathcal{J}\&\mathcal{F}$.

Method	prec@0.5	prec@0.6	prec@0.7	prec@0.8	prec@0.9	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$
Baseline (frame-based) [56]	31.98	27.66	21.54	14.56	4.33	33.34	36.54	34.94
Baseline + RNN [56]	40.24	35.90	30.34	22.26	9.35	34.79	38.08	36.44
URVOS [43]	51.19	46.77	40.16	27.68	14.11	45.27	49.19	47.23
Ours	**54.18**	**48.99**	**42.20**	**33.62**	**18.94**	**48.43**	**50.96**	**49.70**

Table 2. **The quantitative evaluation on Refer-YouTube-VOS *val* set**, with region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$, average of $\mathcal{J}\&\mathcal{F}$. Success percentage (prec@X) is also reported.

ring expressions. It includes 90 videos, which are further split into two subsets: *training* set (60 videos), *val* set (30 videos). Refer-YouTube-VOS is a large-scale dataset, which includes 3,975 videos from YouTube-VOS [54] and 27,899 corresponding language descriptions. Similar to Refer-DAVIS$_{17}$, this dataset contains two subsets: *training* set and *val* set. Although both provide the full-video expression based on an entire video and the first-frame expression based on the first frame, we only use their full-video expression for training and validation.

Evaluation Metrics. Following the standard evaluation protocol [43], we adopt the region similarity $\mathcal{J}$ (%), contour accuracy $\mathcal{F}$ (%), and Precision@$\mathcal{X}$ (%) as our evaluation metrics. The region similarity $\mathcal{J}$ calculates the mean IoU between the prediction and ground truth, while the contour accuracy $\mathcal{F}$ measures the similarity between the boundary of the prediction and the ground truth. Precision@X (prec@$\mathcal{X}$) denotes the percentage of testing samples whose region similarity is higher than a predefined threshold $\mathcal{X}$, where $\mathcal{X}$ is sampled from the range $[0.5, 0.9]$.

4.2. Quantitative Results

We compare our approach with several previous models on the two aforementioned challenging benchmarks. Baseline is a frame-based method proposed in [43], which employs a cross-modal attention module [56] for vision-language understanding, and a feature pyramid decoder for mask prediction. Baseline+RNN [43] denotes a variant of baseline, which utilizes a GRU layer to visual features from multiple input frames for estimation of masks. URVOS [43] builds on frame-level interaction, which unifies a memory network to replay previous frames and masks for refining the mask prediction of the current frame. RefVOS [1] is a simple frame-based modeling method, which directly conducts element-wise multiplication between visual and linguistic features to obtain the cross-modal representation.

Refer-DAVIS$_{17}$ *val* set. Before training on Refer-DAVIS$_{17}$, we pre-train our model on the large-scale Refer-YouTube-VOS *training* set, and test its performance on the Refer-DAVIS$_{17}$ *val* set. As reported in Table 1, our approach has a remarkable performance improvement compared to the most recent model URVOS under the same 'pretraining only' case ($\mathcal{J}$: +5.8%, $\mathcal{F}$: +6.0%). After fine-tuning the pretrained model on the Refer-DAVIS$_{17}$ *training* set, our approach largely outperforms all the comparative methods across all metrics ($\mathcal{J}$:+6.6%, $\mathcal{F}$:+6.1% compared with URVOS). Besides, we also provide the results of our model pre-trained on RefCOCO [37], a referring image segmentation benchmark, which achieves higher scores than these frame-based methods, like URVOS [43] and RefVOS [1].

Refer-YouTube-VOS *val* set. We further examine the performance of the proposed approach on the Refer-YouTube-VOS *val*. We directly test the model trained on the Refer-YouTube-VOS *training* set. As seen in Table 2, our model significantly outperforms all the state-of-the-art methods in all metrics. Compared to URVOS [43], we improve the region similarity $\mathcal{J}$ by +3.1% and the contour accuracy $\mathcal{F}$ by +1.8%. Our method obtains a much higher score on precision@X (*e.g.*, prec@0.8:+5.0%, prec@0.9:+4.8%). All the results indicate the superiority of our multi-level representation learning with semantic alignment.

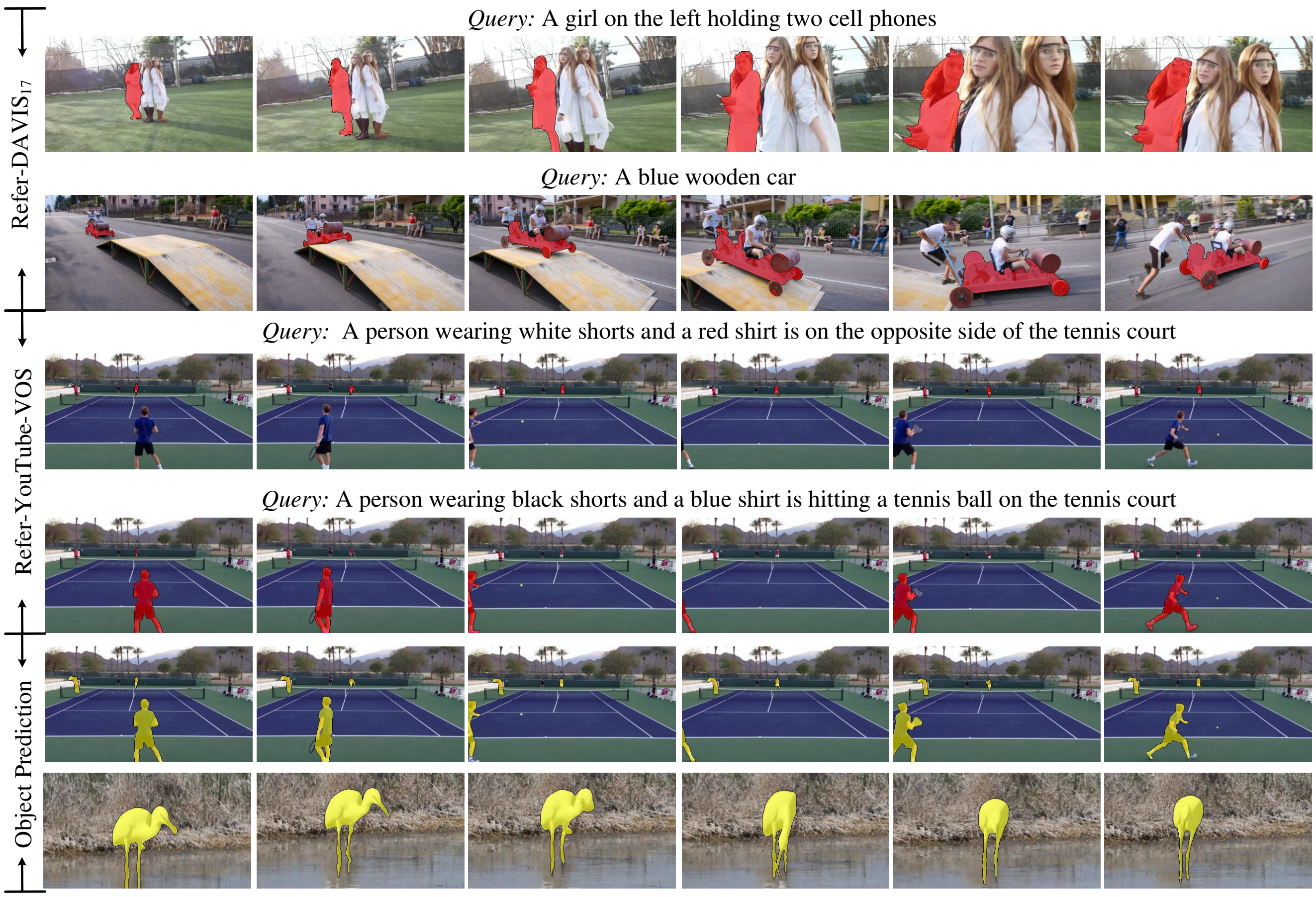

Figure 4. **Qualitative results on Refer-DAVIS$_{17}$ *val* and Refer-YouTube-VOS *val* set**. The first four sequences represent the referring video object segmentation results. The last two sequences are object-level results with respect to the salient object prediction (Eq. 6).

4.3. Qualitative Results

Fig. 4 shows some typical visual results of our approach. In the first sequence (*i.e.*, lab-coat), camera movement brings size deformation for the girl. In the second sequence (*i.e.*, soapbox), the blue wooden car moves forward, which has difficulty in boundary estimation due to the considerable appearance variation. The third and fourth sequences come from the same video (*i.e.*, 6031809500) but are more challenging due to local occlusion and visually similar objects in the background. Otherwise, our model succeeds in segmenting all the referent objects. Overall, benefiting from taking the multi-level embedding into account during the vision-language understanding, our model yields remarkable referring video object segmentation results.

In addition to the referring video object segmentation results, we also provide some object prediction results from object-level embedding in Fig. 4. It is well seen that all the objects are predicted with sharp boundaries, including the occluded and small ones, indicating that the object-aware feature maps can guide the generation of the salient object and provide object prior.

4.4. Ablation Studies

To analyze the effect of each component in our model, we conduct ablative studies on two benchmarks. Table 3 and Table 4 tabulate the results.

Multi-Level Analysis. To investigate our multi-level representation, we separately analyze video, frame, and object embedding in Table 3. As seen, by dropping the video embedding, the model encounters a performance drop ($\mathcal{J}$:-2.2%, $\mathcal{F}$:-1.9%). A similar trend is observed after discarding another two modules, thereby demonstrating the effectiveness of the multi-level representations. Moreover, we test two different variants of the object encoder, *i.e.*, FCN or frame-level encoder. But both two have lower scores than video-level encoder (*i.e.*, the full model).

Fig. 5 shows the ablative qualitative results by adding frame, video, object embedding one by one. The simple frame-level modeling cannot identify the moving and occluded objects accurately. Using video-level and object-level embedding can promote performance by learning the long-temporal information and shifting more attention.

Importance of Semantic Alignment. DSA is a key module in our method to achieve cross-modal understanding.

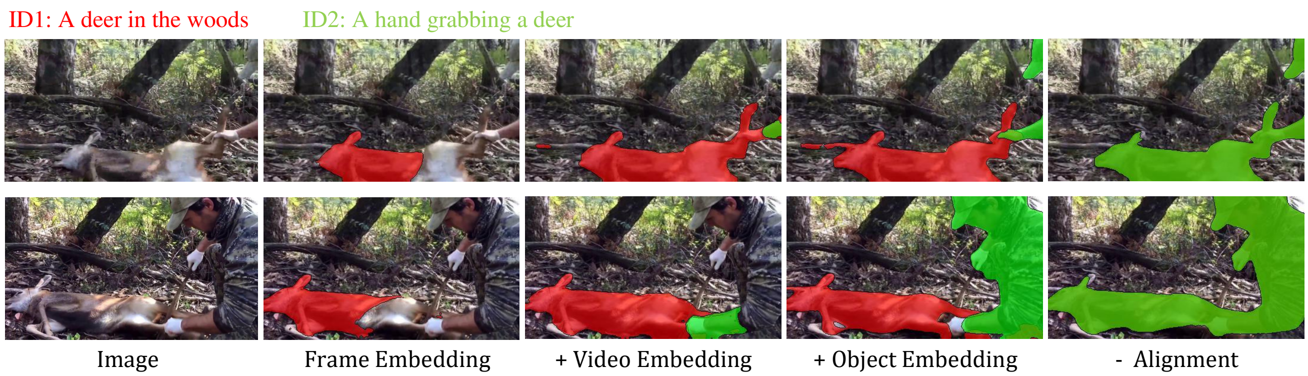

Figure 5. **Qualitative results about ablation studies on Refer-YouTube-VOS**. The video and object embedding is added into the frame embedding model one by one. Note that '-Alignment' is removed from the full model, with the same annotations for two objects.

Aspect	Variants	$\mathcal{J}$	$\mathcal{F}$
Full Model	-	**48.43**	**50.96**
Multi-Level Visual Representation	*w/o* video level	46.25	49.04
	w/o frame level	47.09	49.58
	w/o object level	46.95	49.10
Object Encoder	FCN	47.24	49.65
	frame encoder	47.56	50.04
Semantic Alignment	*w/o* alignment	36.23	40.40
Numbers of Frames	1	46.12	48.90
	2	46.80	49.24
	4	47.47	49.83
	8	48.11	50.41

Table 3. **Ablation studies on Refer-YouTube-VOS *val* set**, with region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$.

From Table 3, we can see that removing semantic alignment from our full model brings a considerable performance drop across all metrics ($\mathcal{J}$:-12.2%, $\mathcal{F}$:-10.6%). Fig. 5 clearly shows that semantic alignment plays an important role in identifying different objects.

Number of Frames. We also study the influence of different numbers of video frames on the final performance in Table 3. Better performance can be obtained with more input frames (*e.g.*,1 $\rightarrow$ 8). This observation indicates that the long-temporal modeling can mine cross-frame relationships to facilitate referring video segmentation. Due to the computation and memory limitation, we set the maximum number to 12 in our full model.

Mask Propagation Method. Next, we experiment several state-of-the-art mask propagation methods in Table 4, such as STM [38], CFBI+ [55], STCN [4], where STCN brings more refinement improvements. Further, we can observe that the performance gain is 3.9% and 3.5% in $\mathcal{J}$ and $\mathcal{F}$, respectively. It is worth noticing that our model without mask propagation still achieves better performance in comparison to the state-of-art URVOS ($\mathcal{J}$:+2.7%, $\mathcal{F}$:+0.6%).

Inference Speed. Finally, we calculate the inference speed on a NVIDIA Tesla V100 GPU using the entire Refer-DAVIS$_{17}$ *val* set. The input images are tested with size of 432×240, and Table 4 shows all FPS results. Our multi-grained model processes all input frames in a parallel way, which demonstrates the speed superiority at **53.2 FPS**. The speed of the full model with STCN [4] outperforms other methods with significant $3\times$ margins and achieves **17.2 FPS**. All the results indicate that our model is an efficient framework with a high inference speed.

Method	Propagation	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$&$\mathcal{F}$	FPS
URVOS	–	39.43	45.87	42.65	-
	STM [38] ICCV19	47.29	55.96	51.45	-
Ours	–	49.96	56.53	53.25	**53.2**
	STM [38] ICCV19	51.02	58.65	54.84	5.59
	CFBI+ [55] PAMI21	52.39	59.37	55.88	5.01
	STCN [4] NeurIPS21	**53.85**	**62.02**	**57.94**	17.2

Table 4. **Ablation studies about mask propagation on Refer-DAVIS$_{17}$ *val* set**, with region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$, average of $\mathcal{J}$&$\mathcal{F}$. Inference speed (FPS) is also reported.

5. Conclusion

In this paper, we proposed a novel multi-level representation learning framework to address RVOS task. We started with the observation that most RVOS methods rely heavily on frame-level modeling and omit the structural information of video content, leading to poor vision-language matching. Based on this motivation, we proposed to embed video-, frame-, and object-level semantics to provide a robust and informative visual representation. Afterward, to distinguish the referent object, we introduced dynamic semantic alignment for adaptively fusing two modalities. The boundary-aware segmentation integrated the generated target-aware feature and boundary information to predict the final results. Experiments show that our method outperforms previous methods by large margins on both Refer-DAVIS$_{17}$ and Refer-YouTube-VOS.

References

[1] Miriam Bellver, Carles Ventura, Carina Silberer, Ioannis Kazakos, Jordi Torres, and Xavier Giro-i Nieto. Refvos: A closer look at referring expressions for video object segmentation. *arXiv preprint arXiv:2010.00263*, 2020. 1, 2, 6

[2] Kaidi Cao, Maria Brbic, and Jure Leskovec. Concept learners for few-shot learning. In *ICLR*, 2020. 2

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 3

[4] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Rethinking space-time networks with improved memory coverage for efficient video object segmentation. In *NeurIPS*, 2021. 5, 8

[5] 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. 5

[6] Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. Vision-language transformer and query generation for referring segmentation. In *ICCV*, 2021. 1

[7] Xingping Dong, Jianbing Shen, Dongming Wu, Kan Guo, Xiaogang Jin, and Fatih Porikli. Quadruplet network with one-shot learning for fast visual object tracking. *IEEE TIP*, 2019. 2

[8] Ruoyi Du, Dongliang Chang, Ayan Kumar Bhunia, Jiyang Xie, Zhanyu Ma, Yi-Zhe Song, and Jun Guo. Fine-grained visual classification via progressive multi-granularity training of jigsaw patches. In *ECCV*, 2020. 2

[9] Feifan Fan, Yansong Feng, and Dongyan Zhao. Multi-grained attention network for aspect-level sentiment classification. In *EMNLP*, 2018. 2

[10] Guang Feng, Zhiwei Hu, Lihe Zhang, and Huchuan Lu. Encoder fusion network with co-attention embedding for referring image segmentation. In *CVPR*, 2021. 1

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

[12] Wenbin Ge, Xiankai Lu, and Jianbing Shen. Video object segmentation using global and instance embedding learning. In *CVPR*, 2021. 1

[13] Jianyuan Guo, Yuhui Yuan, Lang Huang, Chao Zhang, Jin-Ge Yao, and Kai Han. Beyond human parts: Dual part-aligned representations for person re-identification. In *ICCV*, 2019. 2

[14] Jie Hao, Xing Wang, Shuming Shi, Jinfeng Zhang, and Zhaopeng Tu. Multi-granularity self-attention for neural machine translation. In *EMNLP*, 2019. 2

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3, 5

[16] Yuk Heo, Yeong Jun Koh, and Chang-Su Kim. Interactive video object segmentation using global and local transfer modules. In *ECCV*, 2020. 2

[17] Ping Hu, Fabian Caba, Oliver Wang, Zhe Lin, Stan Sclaroff, and Federico Perazzi. Temporally distributed networks for fast video semantic segmentation. In *CVPR*, 2020. 2

[18] Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from natural language expressions. In *ECCV*, 2016. 1

[19] Tianrui Hui, Shaofei Huang, Si Liu, Zihan Ding, Guanbin Li, Wenguan Wang, Jizhong Han, and Fei Wang. Collaborative spatial-temporal modeling for language-queried video actor segmentation. In *CVPR*, 2021. 2

[20] Suyog Dutt Jain, Bo Xiong, and Kristen Grauman. Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. In *CVPR*, 2017. 2

[21] 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. 1

[22] Ya Jing, Tao Kong, Wei Wang, Liang Wang, Lei Li, and Tieniu Tan. Locate then segment: A strong pipeline for referring image segmentation. In *CVPR*, 2021. 1

[23] Ya Jing, Chenyang Si, Junbo Wang, Wei Wang, Liang Wang, and Tieniu Tan. Pose-guided multi-granularity attention network for text-based person search. In *AAAI*, 2020. 3

[24] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *ICCV*, 2021. 1

[25] Anna Khoreva, Anna Rohrbach, and Bernt Schiele. Video object segmentation with language referring expressions. In *ACCV*, 2018. 1, 2, 5, 6

[26] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 4

[27] Chenxi Liu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, and Alan Yuille. Recurrent multimodal interaction for referring image segmentation. In *ICCV*, 2017. 1

[28] Xiankai Lu, Wenguan Wang, Martin Danelljan, Tianfei Zhou, Jianbing Shen, and Luc Van Gool. Video object segmentation with episodic graph memory networks. In *ECCV*, 2020. 2

[29] Xiankai Lu, Wenguan Wang, Jianbing Shen, David Crandall, and Jiebo Luo. Zero-shot video object segmentation with co-attention siamese networks. *IEEE TPAMI*, 2020. 2

[30] Xiankai Lu, Wenguan Wang, Jianbing Shen, Yu-Wing Tai, David J Crandall, and Steven CH Hoi. Learning video object segmentation from unlabeled videos. In *CVPR*, 2020. 2

[31] Gen Luo, Yiyi Zhou, Xiaoshuai Sun, Liujuan Cao, Chenglin Wu, Cheng Deng, and Rongrong Ji. Multi-task collaborative network for joint referring expression comprehension and segmentation. In *CVPR*, 2020. 1

[32] Lei Ma, Felix Juefei-Xu, Fuyuan Zhang, Jiyuan Sun, Minhui Xue, Bo Li, Chunyang Chen, Ting Su, Li Li, Yang Liu, et al. Deepgauge: Multi-granularity testing criteria for deep learning systems. In *ASE*, 2018. 2

[33] Yunyao Mao, Ning Wang, Wengang Zhou, and Houqiang Li. Joint inductive and transductive learning for video object segmentation. In *ICCV*, 2021. 1

[34] Bruce McIntosh, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. Visual-textual capsule routing for text-based video segmentation. In *CVPR*, 2020. 2

[35] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *3DV*, 2016. 4

[36] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-global video-text interactions for temporal grounding. In *CVPR*, 2020. 3

[37] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *ECCV*, 2016. 6

[38] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *ICCV*, 2019. 2, 8

[39] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 5

[40] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *CVPR*, 2017. 2

[41] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 5

[42] Tao Ruan, Ting Liu, Zilong Huang, Yunchao Wei, Shikui Wei, and Yao Zhao. Devil in the details: Towards accurate single and multiple human parsing. In *AAAI*, 2019. 5

[43] Seonguk Seo, Joon-Young Lee, and Bohyung Han. Urvos: Unified referring video object segmentation network with a large-scale benchmark. In *ECCV*, 2020. 1, 2, 4, 5, 6

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 4

[45] Hao Wang, Cheng Deng, Fan Ma, and Yi Yang. Context modulated dynamic networks for actor and action video segmentation with language queries. In *AAAI*, 2020. 2

[46] Hao Wang, Cheng Deng, Junchi Yan, and Dacheng Tao. Asymmetric cross-guided attention network for actor and action video segmentation from natural language query. In *CVPR*, 2019. 2

[47] Wenguan Wang, Jianbing Shen, Xiankai Lu, Steven CH Hoi, and Haibin Ling. Paying attention to video object pattern understanding. *IEEE TPAMI*, 2020. 2

[48] Wenguan Wang, Jianbing Shen, Jianwen Xie, Ming-Ming Cheng, Haibin Ling, and Ali Borji. Revisiting video saliency prediction in the deep learning era. *IEEE TPAMI*, 2019. 2

[49] Wenguan Wang, Tianfei Zhou, Fatih Porikli, David Crandall, and Luc Van Gool. A survey on deep learning technique for video segmentation. *IEEE TPAMI*, 2021. 1, 2

[50] Xiaohan Wang, Linchao Zhu, and Yi Yang. T2vlad: global-local sequence alignment for text-video retrieval. In *CVPR*, 2021. 3

[51] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *CVPR*, 2021. 3, 4

[52] Dongming Wu, Mang Ye, Gaojie Lin, Xin Gao, and Jianbing Shen. Person re-identification by context-aware part attention and multi-head collaborative learning. *IEEE TIFS*, 2021. 2

[53] Haozhe Xie, Hongxun Yao, Shangchen Zhou, Shengping Zhang, and Wenxiu Sun. Efficient regional memory network for video object segmentation. In *CVPR*, 2021. 1

[54] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. Youtube-vos: Sequence-to-sequence video object segmentation. In *ECCV*, 2018. 1, 2, 6

[55] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by multi-scale foreground-background integration. *IEEE TPAMI*, 2021. 8

[56] Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. Cross-modal self-attention network for referring image segmentation. In *CVPR*, 2019. 1, 2, 4, 6

[57] Linwei Ye, Mrigank Rochan, Zhi Liu, Xiaoqin Zhang, and Yang Wang. Referring segmentation in images and videos with cross-modal self-attention network. *IEEE TPAMI*, 2021. 2

[58] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *CVPR*, 2018. 1, 2

[59] Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. Dense regression network for video grounding. In *CVPR*, 2020. 3

[60] Yuting Zhang, Luyao Yuan, Yijie Guo, Zhiyuan He, I-An Huang, and Honglak Lee. Discriminative bimodal networks for visual localization and detection with natural language queries. In *CVPR*, 2017. 1, 2

[61] Tianfei Zhou, Jianwu Li, Shunzhou Wang, Ran Tao, and Jianbing Shen. Matnet: Motion-attentive transition network for zero-shot video object segmentation. *IEEE TIP*, 2020. 2