This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Decoupling Static and Hierarchical Motion Perception for Referring Video Segmentation

Shuting He[1] Henghui Ding[1,2] ✉
[1]Nanyang Technological University [2]Institute of Big Data, Fudan University

Abstract

Referring video segmentation relies on natural language expressions to identify and segment objects, often emphasizing motion clues. Previous works treat a sentence as a whole and directly perform identification at the video-level, mixing up static image-level cues with temporal motion cues. However, image-level features cannot well comprehend motion cues in sentences, and static cues are not crucial for temporal perception. In fact, static cues can sometimes interfere with temporal perception by overshadowing motion cues. In this work, we propose to decouple video-level referring expression understanding into static and motion perception, with a specific emphasis on enhancing temporal comprehension. Firstly, we introduce an expression-decoupling module to make static cues and motion cues perform their distinct role, alleviating the issue of sentence embeddings overlooking motion cues. Secondly, we propose a hierarchical motion perception module to capture temporal information effectively across varying timescales. Furthermore, we employ contrastive learning to distinguish the motions of visually similar objects. These contributions yield state-of-the-art performance across five datasets, including a remarkable ***9.2%*** *$\mathcal{J}\&\mathcal{F}$ improvement on the challenging MeViS dataset. Code is available at https://github.com/heshuting555/DsHmp.*

1. Introduction

Referring video segmentation [8, 14, 23, 47] is a continually evolving area that lies at the crossroads of computer vision and natural language processing. This emerging realm of study is concentrated on segmenting and tracking specific objects of interest within video content, guided by natural language expressions. While it shares a historical connection with video object segmentation, it distinguishes itself by leveraging natural language expressions as guidance, which can offer motion-related cues like "*walking*" and "*jumping*". Notably, recent datasets like MeViS [8] have emphasized the role of motion expressions in this evolving field, underscoring the significance of comprehending multi-modal motion information in this context.

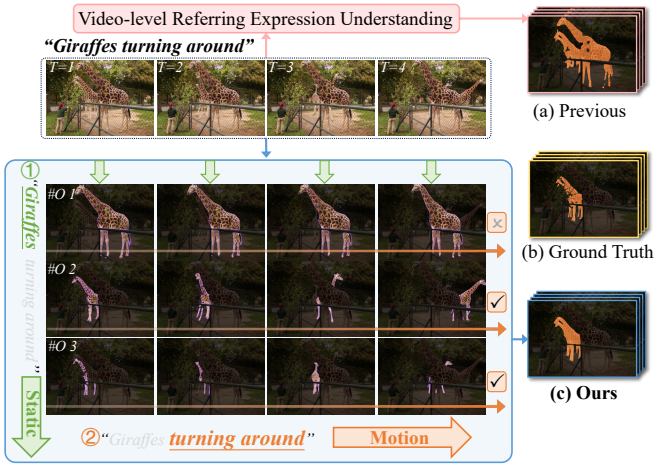


Figure 1. Previous works treat a sentence as a whole and perform referring understanding at the video-level. However, image-level features struggle to understand motion cues, and static cues can sometimes disrupt temporal perception by overshadowing motion cues. We introduce a decoupling of static and motion perception, with a particular focus on enhancing temporal understanding.

Current approaches [8, 15, 29, 43, 49, 55, 56] in referring video segmentation typically oversimplify the complex nature of language by reducing it to a single sentence embedding. For example, ReferFormer [56] and LMPM [8] both employ the strategy of duplicating a single sentence embedding for multiple query embeddings within a Transformer architecture. This approach tends to overshadow the distinct importance and unique static or motion cues offered by the sentence. For instance, consider the sentences: "*The little girl in red standing near the chair and drinking*" and "*The little girl in red moving near the chair and drinking*". These two sentences share 10 out of 11 words, leading to highly similar sentence embeddings, despite potentially referring to different targets. To address this issue, we propose to decouple image-level segmentation and temporal-level motion understanding. As shown in Fig. 1, we let the given sentence focus on two distinct components: static and

✉ Corresponding author (henghui.ding@gmail.com).

motion. The static cues are leveraged to identify potential candidates based on the static visual features present in each individual frame, where motion cues are not necessary. Then, the motion cues are used to pinpoint the target objects among the identified candidates by aligning them with temporal features observed throughout the video. In this way, static cues and motion cues perform their distinct and complementary roles, enhancing the comprehensive understanding of the referring expressions and videos.

A significant challenge in referring video segmentation is the precise capture and alignment of motions across the temporal domain. The motion cues provided by expressions may span a variable number of frames. For example, there are brief actions that occur over a few frames, like "*flying away*", as well as long-term actions that persist throughout the entire video, such as "*walking from leftmost to rightmost*". The unpredictability of the number of frames in which these actions occur greatly intensifies the challenge and complexity of capturing and comprehending motion. Recently, LMPM [8] has introduced a method to capture motions using object tokens, offering computational efficiency and increasing the number of frames considered in temporal learning. However, LMPM tends to treat all frames uniformly and overlooks the distinctions between fleeting motions and long-term motions. In this work, we propose a hierarchical motion perception module to gradually comprehend temporal information based on object tokens, starting from short-term actions and progressing toward long-term actions. This module mimics the way humans understand videos by processing short clips and building an understanding of long-term concepts based on the recollection of short-term clips.

Furthermore, another challenge arises in differentiating between objects that exhibit nearly identical static appearances, such as two sheep that look very similar but have distinct motions. In such cases, the visual features extracted by the image encoder are highly alike, necessitating a significant reliance on the nuanced temporal differences to distinguish similar-looking objects. To address this challenge and enhance object discrimination using motion cues, we employ an object-wise contrastive learning. For a robust learning process, a memory bank is built to generate feature centroids for different objects, which greatly enhance the quality and quantity of positive and negative samples. We prioritize negative samples from the same category, like the three giraffes in Fig. 1, as they are more likely to share a similar appearance and serve as challenging samples for distinguishing objects with distinct motions. This training objective effectively helps to emphasize the disparities in motion features for the similar-looking objects.

In summary, our main contributions are as follows:

- We propose to decouple referring video segmentation into static perception and motion perception. Static perception focuses on grounding candidate objects on image-level based on static cues, while motion perception aims at understanding temporal context and identifying the target objects on temporal-level using motion cues.
- We propose a Hierarchical Motion Perception that effectively processes temporal motions, enabling the capture of motion patterns spanning various frame intervals.
- We leverage contrastive learning to acquire discriminative motion representations and enhance the model's ability to distinguish visually similar objects using motion cues.
- We achieve new state-of-the-art performance on five referring video segmentation datasets, with a particularly significant **9.2%** $\mathcal{J}\&\mathcal{F}$ improvement on the challenging MeViS dataset, showing the effectiveness of our method.

2. Related Work

Referring Image Segmentation. Referring image segmentation aims to segment the target object within the image according to the given sentence [6, 13, 18, 27, 31, 32, 34, 42]. Its prevailing methods fall into two main categories: one-stage methods that perform end-to-end prediction and two-stage methods that involve instance segmentation followed by language-instance matching. For example, Hu et al. [18] fuse visual and linguistic features and then conduct pixel-wise classification for mask prediction, representing a one-stage method. In contrast, Yu et al. [63] use a instance segmentation model to detect all instances in the image and subsequently select the one that best aligns with the sentence, characterizing a two-stage approach. More recently, the success of Transformer [28, 51] has inspired a wave of research in referring image segmentation. Ding et al. [7] first introduce Transformer into this domain and propose the Vision-Language Transformer (VLT). Subsequently, many Transformer-based methods have emerged [19, 24, 33, 35, 45, 48, 53, 57, 58, 60, 64, 66].

Referring Video Segmentation. Referring video segmentation aims to segment the target object within a given video according to a natural language expression [4, 11, 15, 26, 29, 30, 36, 43, 49, 52, 54, 55, 61, 65]. This field is continually evolving with the introduction of A2D-Sentences [14], Ref-DAVIS17 [23], Ref-YouTube-VOS [47], and MeViS [8]. Many previous methods in referring video segmentation have primarily adapted referring image segmentation approaches to perform frame-by-frame target object segmentation, often overlooking the temporal dimension. For example, Khoreva *et al*. [23] ues the referring image segmentation method MAttNet [63] for frame-level segmentation and then applied post-processing techniques to ensure temporal consistency. URVOS [47] and RefVOS [1] utilize cross-modal attention for per-frame segmentation but do not leverage the temporal dimension. Despite their performance, these methods

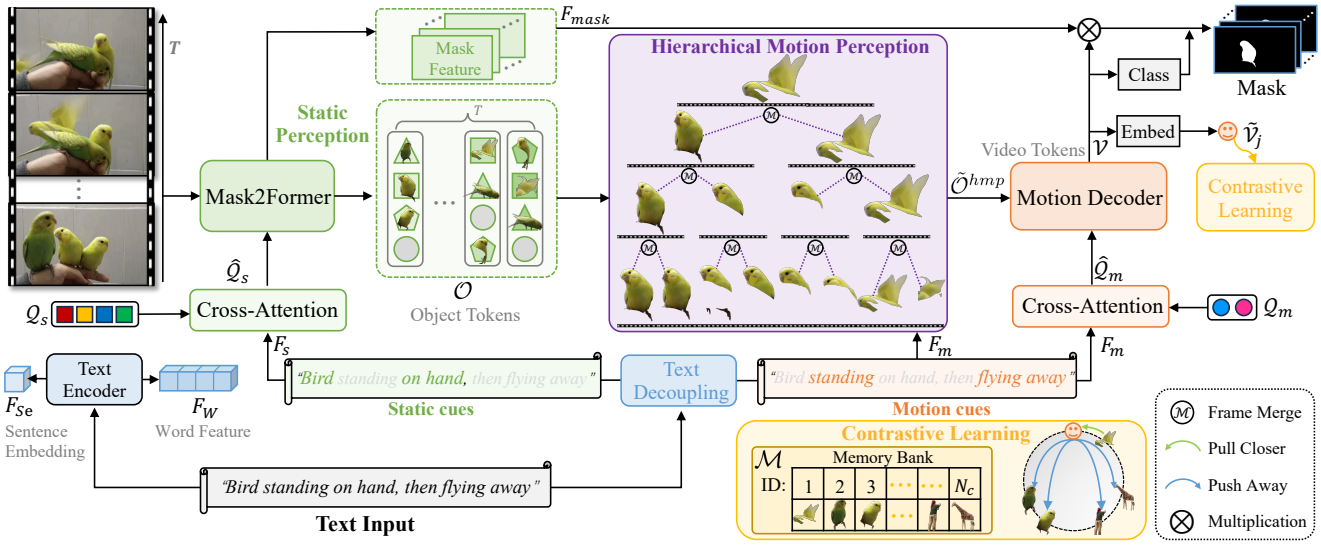


Figure 2. Overview of the proposed approach, named as DsHmp. We decouple the referring video segmentation to image-level static perception and temporal-level motion perception. We first employ Mask2Former to segment the possible objects according to static cues F_s. Then based on the object tokens $\mathcal{O}$ generated by Mask2Former, a hierarchical motion perception is employed to gradually comprehend temporal motions from short-term to long-term. Next, we employ a Motion Decoder to identify the target object according to motion cues F_m and produce video tokens $\mathcal{V}$, which are used for mask predictions. Contrastive learning is applied on video tokens to help the model differentiate visually similar objects with distinct motion patterns.

largely overlook the motion information inherent to videos. Some other works, like ReferFormer [56] and MTTR [2], employ the DETR-like structure in the RVOS field which simplifies the referring pipeline and achieves impressive performance. More recently, based on complex video object segmentation dataset MOSE [9], MeViS [8] dataset has been constructed to emphasize the significance of motion expressions and highlight the inadequacies of existing methods in comprehending the motion information present in languages and videos. In our work, the primary focus is on enhancing the understanding of motion cues in both visual and linguistic features.

3. Approach

The overview of the proposed approach, named DsHmp, is shown in Fig. 2. We first extract word feature F_W and sentence embedding F_{Se}, and decouple the given sentence to static cues F_s and motion cues F_m. With the static cues F_s as queries, we employ a Mask2Former [5] to extract object tokens $\mathcal{O}$ of potential candidate objects and mask feature F_{mask} at image-level. Then the proposed Hierarchical Motion Perception (HMP) is conducted on the object tokens to progressively and hierarchically collect temporal information, generating motion-aware object tokens $\tilde{\mathcal{O}}^{hmp}$ with the guidance of motion cues F_m. Next, we employ the motion cues F_m to identify the target object with a Motion Decoder and produce video tokens $\mathcal{V}$. Finally, the predicted masks are obtained by multiplying the video tokens $\mathcal{V}$ and mask features F_{mask}, and these with class scores higher than a threshold are selected as output. Contrastive learning is applied on the video tokens to enhance the model's ability to distinguish objects using motion cues. To facilitate contrastive learning, a memory bank $\mathcal{M}$ is established to store video tokens from various objects, ensuring a supply of high-quality negative samples.

3.1. Decoupling Motion and Static Perception

Existing approaches [8, 15, 29, 43, 49, 55, 56] in referring video segmentation often oversimplify the complex nature of language by reducing it to a single sentence embedding. Meanwhile, commonly used visual backbones, such as Mask2Former [5] and Video-Swin [38], primarily function as image-level or short-video-level (*e.g.*, 5 frames) segmentation models. They face a challenge in comprehending motion cues within a single sentence embedding.

To address these challenges and effectively leverage the cues provided by expressions, we propose to decouple the static perception and motion perception. Concerning language, we introduce a decoupling of the given expression into static and motion cues, which serve as cues for static perception and motion perception, respectively. In terms of visual processing, we employ Mask2Former [5] to concentrate on extracting potential objects relevant to image-level static cues. Subsequently, the Hierarchical Motion Perception (see Sec. 3.2) and Motion Decoder are responsible for capturing motion based on motion cues. This decoupling allows both sub-tasks, *i.e.*, static perception and motion perception, to learn more comprehensively.

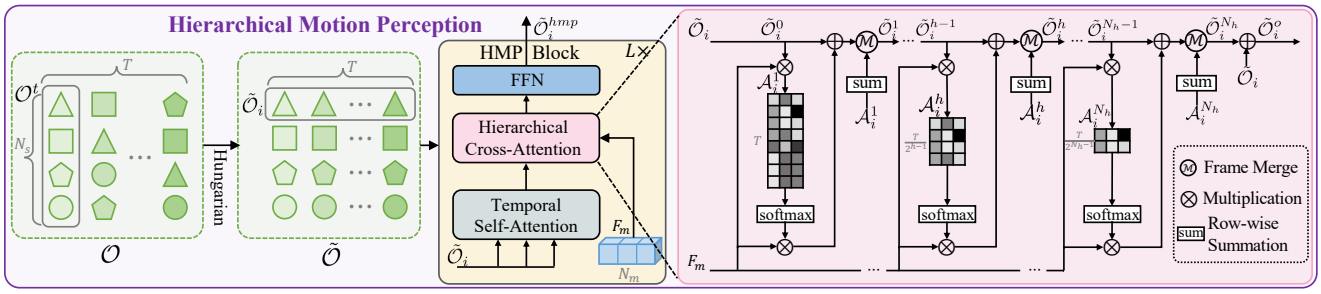


Figure 3. Architecture of the proposed Hierarchical Motion Perception (HMP). Hierarchical Motion Perception module effectively processes short-term and long-term motions, enabling the capture of motion patterns spanning various frame intervals.

As shown in Fig. 2, given the sentence "*Bird standing on hand, then flying away*," we employ an external tool [46] to identify nouns, adjectives, and prepositions in the sentence, yielding static cues such as "*bird, on, hand*." Meanwhile, we extract verbs and adverbs, obtaining motion cues like "*standing, flying away*." Consequently, we extract static words feature as $F_s \in \mathbb{R}^{K_s \times C}$ and motion words feature as $F_m \in \mathbb{R}^{K_m \times C}$, where K_s/K_m represents the length of static/motion cues and C is number of channels. It is worth noting that we add the sentence embedding F_{Se} in both the motion and static features to provide a contextual understanding of the given expression.

Furthermore, unlike the previous methods [8, 56], which directly use language features as object query, we use cross-attention to inject static cues into the learnable static query:

$$\hat{\mathcal{Q}}_s = \mathcal{Q}_s + \text{softmax}\left(\frac{\mathcal{Q}_s F_s^T}{\sqrt{C}}\right) F_s, \tag{1}$$

where $\mathcal{Q}_s \in \mathbb{R}^{N_s \times C}$ is the N_s initialized learnable static queries. After the cross-attention, $\hat{\mathcal{Q}}_s$ not only acquires a grasp of the objects data distributions but also captures specific static cues of the target. It enables the image-level segmentation of candidate objects via Mask2Former.

The motion cues F_m are incorporated into the learnable motion query for Motion Decoder:

$$\hat{\mathcal{Q}}_m = \mathcal{Q}_m + \text{softmax}\left(\frac{\mathcal{Q}_m F_m^T}{\sqrt{C}}\right) F_m, \tag{2}$$

where $\mathcal{Q}_m \in \mathbb{R}^{N_m \times C}$ is the N_m initialized learnable motion queries. Following the cross-attention, $\hat{\mathcal{Q}}_m$ gains specific motion cues related to the target. This facilitates the identification of target objects through the Motion Decoder. In this way, static cues and motion cues play distinct and complementary roles, enhancing the comprehensive understanding of the referring expressions and videos.

3.2. Hierarchical Motion Perception

A significant challenge in referring video segmentation is precisely capturing and aligning motions across different timeframes. Expressions can provide motion cues that vary in the number of frames they span, *e.g.*, fleeting or long-term motions. Inspired by LMPM [8] that captures motions via object tokens, we propose a Hierarchical Motion Perception (HMP) module based on object tokens. This module progressively analyzes short-term and long-term motions, mirroring human video comprehension by forming an understanding of long-term concepts through the recollection of short-term clips.

Fig. 3 illustrates the architecture of the proposed HMP module. It takes in the object tokens $\{\mathcal{O}^t | t \in [1, T], \mathcal{O}^t \in \mathbb{R}^{N_s \times C}\}$, which are the N_s candidate objects' tokens generated by the Mask2Former for each of the T frames, and it outputs the motion-aware tokens $\tilde{\mathcal{O}}^{hmp}$. In the HMP module, the Hungarian matching algorithm [25] is first employed to match $\mathcal{O}$ of adjacent frames, as is done in [20]:

$$\begin{cases} \tilde{\mathcal{O}}^t = \text{Hungarian}(\tilde{\mathcal{O}}^{t-1}, \mathcal{O}^t), & t \in [2, T] \\ \tilde{\mathcal{O}}^t = \mathcal{O}^t, & t = 1 \end{cases}, \tag{3}$$

where $\tilde{\mathcal{O}}$ is the matched object tokens and can be considered as the tracking result with noise. As such, we can obtain N_s object trajectories, *i.e.*, $\{\tilde{\mathcal{O}}_i | i \in [1, N_s], \tilde{\mathcal{O}}_i \in \mathbb{R}^{T \times C}\}$, where $\tilde{\mathcal{O}}_i$ is a single object trajectory. As shown in the middle part of Fig. 3, the proposed Hierarchical Motion Perception (HMP) module is composed of L HMP blocks that are cascaded together. Each block consists of three main components: temporal self-attention, hierarchical cross-attention, and FFN layer. Temporal self-attention is utilized to grasp long-term motion, while hierarchical cross-attention is used to progressively and hierarchically gather temporal information from the short-term to the long-term. To perform the hierarchical cross-attention, we first highlight frames containing the target motions by calculating the similarity between the motion feature F_m and each object trajectory $\tilde{\mathcal{O}}_i^{h-1}$ from last hierarchical stage using the following equation:

$$\mathcal{A}_i^h = \text{softmax}\left(\frac{\tilde{\mathcal{O}}_i^{h-1} {F_m}^T}{\sqrt{C}}\right), \tag{4}$$

where $h \in [1, N_h]$ denotes the stage number of hierarchical operation and $\tilde{\mathcal{O}}_i^0 = \tilde{\mathcal{O}}_i$. $\mathcal{A}_i^h \in \mathbb{R}^{T_h \times K_m}$ represents the attention map for the T_h frames of object trajectory $\tilde{\mathcal{O}}_i^{h-1}$ and the K_m motion cues, where $T_h = \frac{T}{2^{h-1}}$. The softmax operation is performed on the T_h axis. Next, we obtain the

motion feature-enriched object feature $\hat{\tilde{\mathcal{O}}}_i^h$ by incorporating the related motion cues, *i.e.*,

$$\hat{\tilde{\mathcal{O}}}_i^h = \tilde{\mathcal{O}}_i^{h-1} + \frac{\mathcal{A}_i^h}{\hat{\mathcal{A}}_i^h} F_m, \tag{5}$$

where $\hat{\mathcal{A}}_i^h = \sum_k \mathcal{A}_i^{h,(t,k)} \in \mathbb{R}^{T_h}$ can be regarded as frame weight by summing the effect of K_m motion cues on each of the T_h frames. This weight signifies the importance of each individual frame's token within the trajectory spanning over T_h frames concerning the motion cues.

Subsequently, we engage in token merging to accumulate short-term motion information and create merged tokens for higher-level understanding within the hierarchical framework. With the motion feature-enriched object feature $\hat{\tilde{\mathcal{O}}}_i^h$ and their corresponding importance weight $\hat{\mathcal{A}}_i^h$, we apply a frame merging operation to combine adjacent two tokens into one token using the following equation:

$$\tilde{\mathcal{O}}_i^h = \textcircled{\scriptsize M}(\hat{\tilde{\mathcal{O}}}_i^h, \hat{\mathcal{A}}_i^h), \tag{6}$$

where $\tilde{\mathcal{O}}_i^h \in \mathbb{R}^{\frac{T_h}{2} \times C}$. $\textcircled{\scriptsize M}$ is a token merging operation that blends two tokens of neighboring frames into a single one using a weighted average based on $\hat{\mathcal{A}}_i^h$. The merging operation reduces redundant tokens and enhances those associated with motion cues. The merged trajectory $\tilde{\mathcal{O}}_i^h$ is used as input for the next stage to generate $\tilde{\mathcal{O}}_i^{h+1} \in \mathbb{R}^{\frac{T_h}{4} \times C}$ with a larger temporal context view.

The operations from Eq. (4) to Eq. (6) are iteratively performed for a total of N_h times, as shown in Fig. 3. This gradual merging of tokens and expansion of the temporal scope forms a hierarchical motion perception, transitioning from short-term to long-term motion understanding. The output of the hierarchical cross-attention is $\tilde{\mathcal{O}}_i^o$, and it is obtained by:

$$\tilde{\mathcal{O}}_i^o = \tilde{\mathcal{O}}_i + \tilde{\mathcal{O}}_i^{N_h}, \tag{7}$$

where $\tilde{\mathcal{O}}_i^{N_h} \in \mathbb{R}^{\frac{T}{2^{N_h}} \times C}$ and is expanded to match the dimensions of $\tilde{\mathcal{O}}_i$ for summation. $\tilde{\mathcal{O}}_i^o \in \mathbb{R}^{T \times C}$ is then fed to the FFN to generate this block's output, which serves as the input for the next HMP block.

The final output of the Hierarchical Motion Perception module, denoted as $\tilde{\mathcal{O}}^{hmp}$, is used as the key and value inputs to the Motion Decoder for the identification of target objects, along with the query $\hat{\mathcal{Q}}_m$ generated by Eq. (2). The Motion Decoder produces video tokens for target objects, denoted as $\mathcal{V} \in \mathbb{R}^{N_m \times C}$. With the motion-aware object tokens $\tilde{\mathcal{O}}^{hmp}$ generated by Hierarchical Motion Perception, Motion Decoder is able to more effectively understand the motion information conveyed by the language.

3.3. Contrastive Learning

Although the proposed hierarchical motion perception simplifies the motion identification for the Motion Decoder, the presence of objects with highly similar appearances can still pose challenges and lead to confusion in the identification process. To address this challenge, we apply contrastive learning on the output of Motion Decoder, video token $\mathcal{V}$. This approach enhances the model's capacity to differentiate similar-looking objects via motion cues.

• **Vanilla Samples Selection.** In contrastive learning, the choice of positive and negative samples is crucial [16, 59]. We select the video token* with the lowest cost to the ground truth as the anchor, while video tokens of other objects in the mini-batch serve as negative samples. However, this straightforward approach faces two issues: 1) lack of corresponding positive samples, and 2) insufficient negative samples limited by mini-batch size, which significantly impact the final outcome [22]. To address these issues, we introduce a memory bank to store more video tokens.

• **Memory Bank.** Since there are vast numbers of video tokens in our training process, directly storing all the video tokens, like a traditional memory bank [3], severely slows down the learning process. Therefore, we choose to maintain a video token centroid for per target. We introduce a memory bank $\mathcal{M} \in \mathbb{R}^{N_c \times C}$ to gather representative video token centroid for each target, with each element denoting the feature centroid of projected video token $\in \mathbb{R}^C$ from the contrastive head. N_c is the number of target objects in dataset. Specifically, given a video, we utilize the projected feature of anchor video token $\tilde{\mathcal{V}}_j$ to update the corresponding target object centroid feature of $\mathcal{M}$:

$$\mathcal{M}_{[\tilde{\mathcal{V}}_j]} = \beta \mathcal{M}_{[\tilde{\mathcal{V}}_j]} + (1-\beta)\tilde{\mathcal{V}}_j, \tag{8}$$

where $[\tilde{\mathcal{V}}_j]$ is an index mapping from $\tilde{\mathcal{V}}_j$ to its corresponding target index in the memory bank. The hyperparameter β controls the update speed. This way brings two advantages: 1) Storing the centroid feature of target objects instead of each target object feature greatly saves memory consumption. 2) The centroid feature is more representative and robust to encompass the static and motion cues, contributing to the construction of a well-structured feature space.

We apply an object-wise contrastive learning as follows:

$$\mathcal{L}_{con} = -\log \frac{\exp(\tilde{\mathcal{V}}_j \cdot m^+/\tau)}{\exp(\tilde{\mathcal{V}}_j \cdot m^+/\tau) + \sum_{m^- \in \mathbf{N}} \exp(\tilde{\mathcal{V}}_j \cdot m^-/\tau)}, \tag{9}$$

where τ is a temperature hyperparameter. Positive sample m^+ is the feature centroid of the target object that $\tilde{\mathcal{V}}_j$ belongs to, *i.e.* $\mathcal{M}_{[\tilde{\mathcal{V}}_j]}$. $\mathbf{N}$ is the collection of N_n negative samples which come from different objects in $\mathcal{M}$. We prioritize negatives belonging to the same category in the same video, as they are more likely to have a similar appearance and serve as the challenging samples in contrastive learning of distinct motions. $\mathcal{L}_{con}$ is computed after N_i iterations to ensure a stable training process.

*For MeViS [8] dataset with multiple target objects, we average the matched video tokens to represent their collective characteristics.

3.4. Training Objective

Following [8, 17], we employ the match loss $\mathcal{L}_f$ between per-frame outputs and frame-wise ground truth, along with $\mathcal{L}v$ as the video-level loss with video-level ground truth. The total training objective to optimize the model is: $\mathcal{L}_{train} = \mathcal{L}_f + \mathcal{L}_v + \lambda_{con}\mathcal{L}_{con}$, where λ_{con} is used for balancing the contrastive loss $\mathcal{L}_{con}$.

4. Experiments

4.1. Datasets and Evaluation Metrics

Dataset. The proposed approach, named as **DsHmp**, is evaluated on five video datasets: MeViS [8], Ref-YouTube-VOS [47], Ref-DAVIS17 [23], A2D-Sentences [14], and JHMDB-Sentences [21]. MeViS is a newly established dataset that is targeted at motion information analysis and consists of 2,006 videos and 28K annotations. The Ref-YouTube-VOS stands out as the most extensive R-VOS dataset, comprising 3,978 videos and approximately 13K annotations. Ref-DAVIS17 builds upon DAVIS17 [44], enriched with linguistic annotations for diverse objects, and offers 90 videos. A2D-Sentences, designed for actor and action segmentation, encompasses over 3.7K videos paired with 6.6K action annotations. Meanwhile, JHMDB-Sentences provides 928 videos, each with a description, spread across 21 unique action categories.

Evaluation Metrics. Unless otherwise specific, the evaluation metrics we used are: region similarity $\mathcal{J}$ (average IoU), contour accuracy $\mathcal{F}$ (mean boundary similarity), and their average $\mathcal{J}\&\mathcal{F}$. The evaluations are conducted using the official code or online platforms. For A2D-Sentences and JHMDB-Sentences, we employ mAP, overall IoU (oIoU), and mean IoU (mIoU) as the evaluation metrics.

4.2. Implementation Details

For experiments on MeViS dataset, we follow the default setting of [8]. Specifically, we train the models directly on MeViS without any pre-training on RefCOCO/+/g [41, 62]. The training spans 50,000 iterations using the AdamW optimizer [39] with a learning rate set at 0.00005. For experiments on YouTube-VOS/A2D-Sentences, following [2, 56], the experiments begin with pre-training on RefCOCO/+/g [41, 62] and then undergo main training. Besides, models trained on the Ref-YouTube-VOS/A2D-Sentences training set are evaluated directly on the val set of Ref-DAVIS17/JHMDB-Sentences without the use of additional post-processing techniques. During the pre-training phase, we train the model with 300,000 iterations. In the main training phase, we train with 50,000 iterations. All experiments use RoBERTa [37] as the text encoder. All frames are cropped to have the longest side of 640 pixels and the shortest side of 360 pixels during training and evaluation. For hyperparameters, we set values for N_s, N_m, N_h, N_n, N_i, β, τ, λ_{con} at 20, 10, 3, 100, 10,000, 0.2, 0.07, 0.5, respectively.

Index	Components			Results		
	DS	HMP	CL	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
0	✗	✗	✗	39.7	36.6	42.8
1	✓	✗	✗	42.5	39.4	45.6
2	✗	✓	✗	43.8	40.7	46.9
3	✗	✗	✓	42.1	39.0	45.2
4	✓	✓	✗	45.1	41.8	48.4
5	✓	✗	✓	43.9	40.8	47.0
6	✗	✓	✓	44.9	41.7	48.1
7	✓	✓	✓	**46.4**	**43.0**	**49.8**

Table 1. Ablation study of our method on MeViS dataset. DS, HMP, and CL denote components of decoupling sentence, hierarchical motion perception, and contrastive learning, respectively.

4.3. Ablation Study

Since the main focus of this paper is exploiting motion information, we conduct ablation study on MeViS [8].

Module Effectiveness. We conduct ablation experiments to evaluate the effectiveness of different components. As shown in Tab. 1, the inclusion of Decoupling Sentence (DS, index 1) leads to a performance improvement of **2.8%** $\mathcal{J}\&\mathcal{F}$ compared to vanilla baseline (index 0), which is adopted from LMPM [8] with our reproduction. The introduction of DS enhances the model's ability to comprehensively learn static and motion cues for both image-level and temporal-level understanding. Then, HMP (index 2) is used to capture motion information at multiple temporal granularities, encompassing both short-term and long-term motions. HMP improves the performance by **4.1%** $\mathcal{J}\&\mathcal{F}$, highlighting the significance of motion understanding for referring video segmentation. Next, we present Contrastive Learning (CL, index 3) to construct discriminative motion representations enhancing the model's ability to distinguish similar-looking objects. Utilizing CL improves the $\mathcal{J}\&\mathcal{F}$ by **2.4%**. When integrating all the components together (index 7), referred to as DsHmp, we observe a substantial improvement and achieve new state-of-the-art performance of 46.4% $\mathcal{J}\&\mathcal{F}$ on the challenging MeViS dataset, demonstrating the effectiveness of the proposed method.

Importance of sentence decoupling. In Tab. 2 (a), we study the impact of sentence decoupling on the input queries to MaskFormer and Motion Decoder. Utilizing only the basic sentence embedding F_{Se} like ReferFormer [56] results in a 1.5% $\mathcal{J}\&\mathcal{F}$ decrease, demonstrating that relying solely on sentence embedding limits discriminative capability and risks overlooking key cues. Meanwhile, solely using the sentence decoupling query without the sentence embedding F_{Se} leads to a 0.8% $\mathcal{J}\&\mathcal{F}$ decrease, which is due to the lack of sentence context. DS w/o $\mathcal{Q}_s$/$\mathcal{Q}_m$ uses F_s and F_m directly, bypassing the integration of static or motion cues into the learnable query, leading to a 0.5% drop in $\mathcal{J}\&\mathcal{F}$. This underscores the importance of incorporating

Input Query	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
F_{Se}	44.9	41.7	48.1
DS w/o F_{Se}	45.6	42.1	49.1
DS w/o $\mathcal{Q}_s/\mathcal{Q}_m$	45.9	42.3	49.5
DS	**46.4**	**43.0**	**49.8**

(a) Different input query variations.

N_h	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
0	43.9	40.8	47.0
1	45.0	41.8	48.2
2	45.8	42.3	49.3
3	**46.4**	**43.0**	**49.8**

(b) Different hierarchical stages.

N_n	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
0	45.1	41.8	48.4
10	45.4	41.9	48.9
100	**46.4**	**43.0**	**49.8**
200	46.5	43.1	49.8

(c) Different negative samples.

Tokens	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
$\mathcal{O}$	43.8	40.7	46.9
$\tilde{\mathcal{O}}$	**46.4**	**43.0**	**49.8**

(d) Effect of Hungarian match.

Table 2. **Ablation studies of different architecture designs** on MeViS.

a learnable query to grasp the global dataset distribution. These findings show that while sentence embeddings are vital for maintaining language comprehension, relying solely on them does not facilitate a thorough learning process. Besides, decoupling sentence into static and motion cues is helpful for enhancing referring video segmentation.

Number of hierarchical stages N_h in HMP. Tab. 2 (b) shows results obtained with varying numbers of hierarchical stages. For $N_h = 0$, only vanilla temporal self-attention is applied. For $N_h = 1$, the hierarchical cross-attention mechanism is introduced to capture motion cues at per-frame level. With increasing N_h, the attention mechanism is refined to capture a larger temporal context. This hierarchical merging of tokens and extension of the temporal scope facilitates a transition from short-term to long-term motion understanding. To effectively capture motion information across multiple levels of granularity, we have configured the hierarchical stages at 3 to achieve the best result.

Hungarian match in HMP. Tab. 2 (d) shows the necessity of adding Hungarian in HMP. Without Hungarian match, the calculation of motion information and token merge may not align for the same object, adversely impacting performance and resulting in a 2.6% drop in $\mathcal{J}\&\mathcal{F}$.

Number of negative samples N_n. In Tab. 2 (c), we report results with different numbers of negative samples used in contrastive loss in Eq. (9). When only using 10 negative samples, which can be achieved within a mini-batch without memory bank, the performance is improved by 0.3% $\mathcal{J}\&\mathcal{F}$ only. Increasing N_n to 100 brings 1.3% $\mathcal{J}\&\mathcal{F}$ performance gain, demonstrating the necessity of using a memory bank to store more samples. Employing more negative samples in Eq. (9) contributes to the establishment of a discriminative and comprehensive motion representation space yet consumes more computing resources. Striking a balance between accuracy and efficiency, we set N_n to 100.

t-SNE visualization. In Fig. 4, we employ t-SNE [50] to visualize video token distribution with and without our contrastive learning across 25 different objects, each described by multiple language expressions. Without contrastive learning, tokens for the same object diverge due to language diversity, leading to overlap between tokens of similar-looking objects. Contrastive learning, on the other hand, brings tokens of the same target object closer and separates them from other objects, enhancing the model's ability to distinguish motions of visually similar objects.

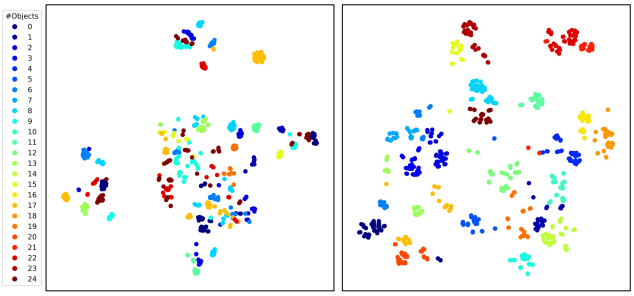


Figure 4. Visualization of features learned w/o CL (left) and w/ CL (right). Features are colored according to class labels. As seen, the proposed CL brings a well-structured video token feature space.

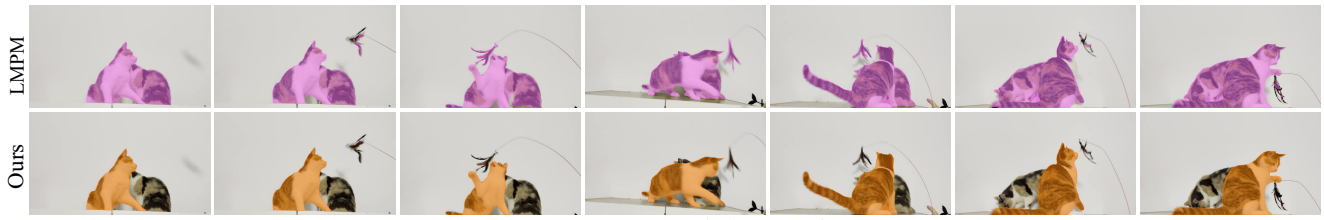
Methods	Reference	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
URVOS [47]	[ECCV'20]	27.8	25.7	29.9
LBDT [12]	[CVPR'22]	29.3	27.8	30.8
MTTR [2]	[CVPR'22]	30.0	28.8	31.2
ReferFormer [56]	[CVPR'22]	31.0	29.8	32.2
VLT+TC [10]	[TPAMI'22]	35.5	33.6	37.3
LMPM [8]	[ICCV'23]	37.2	34.2	40.2
DsHmp (ours)	[CVPR'24]	**46.4**	**43.0**	**49.8**

Table 3. Comparison on MeViS.

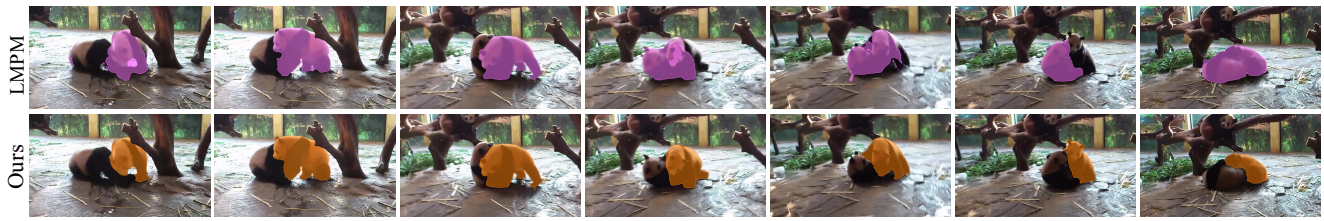
4.4. Comparison with State-of-the-Art Methods

MeViS [8]. In Tab. 3, we evaluate the proposed approach DsHmp on the newly released motion expression video segmentation dataset MeViS. Following [8], we use Swin-Tiny as the backbone. DsHmp achieves superior performance compared to other state-of-the-art methods and surpasses the previous state-of-the-art LMPM [8] by a remarkable **9.2%** $\mathcal{J}\&\mathcal{F}$. These results demonstrate the effectiveness of our approach DsHmp in capturing motion information.

Ref-YouTube-VOS [47] & **Ref-DAVIS17** [23]. In Tab. 4, we report results on Ref-YouTube-VOS and Ref-DAVIS17. Our approach exceeds existing methods on the two datasets across all metrics. On Ref-YouTube-VOS, DsHmp with the Video-Swin-Tiny backbone achieves 63.6% $\mathcal{J}\&\mathcal{F}$, which is 1.2% higher than the previous state-of-the-art SOC [40]. When a larger backbone is used, *i.e.*, Video-Swin-Base, the performance of DsHmp further improves to 67.1% $\mathcal{J}\&\mathcal{F}$, consistently outperforming all other methods by more than 1.1%. On Ref-DAVIS17, our approach achieves 64.9% $\mathcal{J}\&\mathcal{F}$ and surpasses SOC [40] by 0.7%. The performance gains on these two datasets are relatively modest compared to MeViS, mainly because the datasets may include sentences with image-level descriptions for the first frame and not strictly require motion expressions. Despite



(a) *"Cat turning around and playing with toy"*



(b) *"Panda pushing another panda and falling over"*

Figure 5. Visualization results of complex and motion language descriptions on MeViS. Orange masks represent positive segmentation results and pink masks denote the negatives. Our DsHmp can capture temporal information effectively across varying timescales.

Method	Reference	Ref-YouTube-VOS $\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	Ref-DAVIS17 $\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
Video-Swin-Tiny							
ReferFormer [56]	[CVPR'22]	59.4	58.0	60.9	59.6	56.5	62.7
HTML [15]	[ICCV'23]	61.2	59.5	63.0	-	-	-
R^2-VOS [29]	[ICCV'23]	61.3	59.6	63.1	-	-	-
SgMg [43]	[ICCV'23]	62.0	60.4	63.5	61.9	59.0	64.8
TempCD [49]	[ICCV'23]	62.3	60.5	64.0	62.2	59.3	65.0
SOC [40]	[NIPS'23]	62.4	61.1	63.7	63.5	60.2	66.7
DsHmp (ours)	[CVPR'24]	**63.6**	**61.8**	**65.4**	**64.0**	**60.8**	**67.2**
Video-Swin-Base							
ReferFormer [56]	[CVPR'22]	62.9	61.3	64.6	61.1	58.1	64.1
OnlineRefer [55]	[ICCV'23]	62.9	61.0	64.7	62.4	59.1	65.6
HTML [15]	[ICCV'23]	63.4	61.5	65.2	62.1	59.2	65.1
SgMg [43]	[ICCV'23]	65.7	63.9	67.4	63.3	60.6	66.0
TempCD [49]	[ICCV'23]	65.8	63.6	68.0	64.6	61.6	67.6
SOC [40]	[NIPS'23]	66.0	64.1	67.9	64.2	61.0	67.4
DsHmp (ours)	[CVPR'24]	**67.1**	**65.0**	**69.1**	**64.9**	**61.7**	**68.1**

Table 4. Comparison on Ref-YouTube-VOS and Ref-DAVIS17.

this, our method maintains state-of-the-art performance, underscoring its generalizability and effectiveness.

A2D-Sentences & JHMDB-Sentences [14]. We further evaluate the proposed approach DsHmp on A2D-Sentences and JHMDB-Sentences in Tab. 5. Following [56], the models are first pre-trained on RefCOCO/+/g and then fine-tuned on A2D-Sentences. JHMDB-Sentences is used for evaluation only. The proposed DsHmp achieves new state-of-the-art performance and outperforms the nearest competitor SgMg [43] by 1.3% and 0.8% mAP on A2D-Sentences and JHMDB-Sentences, respectively.

4.5. Qualitative Visualization

As shown in Fig. 5, DsHmp is able to understand both fleeting motions "*turning around, falling over*" and long-term motion "*playing, pushing*" and segment the target object precisely. In contrast, LMPM [8] tends to identify all the objects in the video and fails to comprehend the

Method	Reference	A2D-Sentences mAP	oIoU	mIoU	JHMDB-Sentences mAP	oIoU	mIoU
Video-Swin-Tiny							
MTTR [2]	[CVPR'22]	46.1	72.0	64.0	39.2	70.1	69.8
ReferFormer [56]	[CVPR'22]	52.8	77.6	69.6	42.2	71.9	71.0
HTML [15]	[ICCV'23]	53.4	77.6	69.2	42.7	-	-
SOC [40]	[NIPS'23]	54.8	78.3	70.6	42.7	72.7	71.6
SgMg [43]	[ICCV'23]	56.1	78.0	70.4	44.4	72.8	71.7
DsHmp (ours)	[CVPR'24]	**57.2**	**79.0**	**71.3**	**44.9**	**73.1**	**72.1**
Video-Swin-Base							
ReferFormer [56]	[CVPR'22]	55.0	78.6	70.3	43.7	73.0	71.8
OnlineRefer [55]	[ICCV'23]	-	79.6	70.5	-	73.5	71.9
HTML [15]	[ICCV'23]	56.7	79.5	71.2	44.2	-	-
SOC [40]	[NIPS'23]	57.3	80.7	72.5	44.6	73.6	72.3
SgMg [43]	[ICCV'23]	58.5	79.9	72.0	45.0	73.7	72.5
DsHmp (ours)	[CVPR'24]	**59.8**	**81.1**	**72.9**	**45.8**	**73.9**	**73.0**

Table 5. Comparison on A2D-Sentences and JHMDB-Sentences.

motion information. The qualitative results further show the effectiveness of our DsHmp that can capture temporal information effectively across varying timescales.

5. Conclusion

We propose a decoupled static and hierarchical motion perception approach to enhance temporal comprehension for referring video segmentation. The static cues and motion cues provided by the given language expressions are employed to image-level referring segmentation and temporal-level motion identification, respectively. Additionally, our hierarchical motion perception module effectively captures temporal information across varying timescales and can well handle both the fleeting and long-term motions in the language and video. Furthermore, the use of contrastive learning enables the model to distinguish the motions of visually similar objects. The proposed approach consistently achieves new state-of-the-art performance across 5 datasets.

References

[1] Miriam Bellver, Carles Ventura, Carina Silberer, Ioannis Kazakos, Jordi Torres, and Xavier Giro-i Nieto. A closer look at referring expressions for video object segmentation. *Multimedia Tools and Applications*, 2022. 2

[2] Adam Botach, Evgenii Zheltonozhskii, and Chaim Baskin. End-to-end referring video object segmentation with multimodal transformers. In *CVPR*, 2022. 3, 6, 7, 8

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 5

[4] Weidong Chen, Dexiang Hong, Yuankai Qi, Zhenjun Han, Shuhui Wang, Laiyun Qing, Qingming Huang, and Guorong Li. Multi-attention network for compressed video referring object segmentation. In *ACM MM*, 2022. 2

[5] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 3

[6] Henghui Ding, Scott Cohen, Brian Price, and Xudong Jiang. PhraseClick: toward achieving flexible interactive segmentation by phrase and click. In *ECCV*, 2020. 2

[7] Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. Vision-language transformer and query generation for referring segmentation. In *ICCV*, 2021. 2

[8] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, and Chen Change Loy. MeViS: A large-scale benchmark for video segmentation with motion expressions. In *ICCV*, 2023. 1, 2, 3, 4, 5, 6, 7, 8

[9] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, Philip HS Torr, and Song Bai. MOSE: A new dataset for video object segmentation in complex scenes. In *ICCV*, 2023. 3

[10] Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. VLT: Vision-language transformer and query generation for referring segmentation. *IEEE TPAMI*, 2023. 7

[11] Zihan Ding, Tianrui Hui, Shaofei Huang, Si Liu, Xuan Luo, Junshi Huang, and Xiaoming Wei. Progressive multimodal interaction network for referring video object segmentation. *The 3rd Large-scale Video Object Segmentation Challenge*, 2021. 2

[12] Zihan Ding, Tianrui Hui, Junshi Huang, Xiaoming Wei, Jizhong Han, and Si Liu. Language-bridged spatial-temporal interaction for referring video object segmentation. In *CVPR*, 2022. 7

[13] Guang Feng, Zhiwei Hu, Lihe Zhang, and Huchuan Lu. Encoder fusion network with co-attention embedding for referring image segmentation. In *CVPR*, 2021. 2

[14] Kirill Gavrilyuk, Amir Ghodrati, Zhenyang Li, and Cees GM Snoek. Actor and action video segmentation from a sentence. In *CVPR*, 2018. 1, 2, 6, 8

[15] Mingfei Han, Yali Wang, Zhihui Li, Lina Yao, Xiaojun Chang, and Yu Qiao. Html: Hybrid temporal-scale multimodal learning framework for referring video object segmentation. In *ICCV*, 2023. 1, 2, 3, 8

[16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 5

[17] Miran Heo, Sukjun Hwang, Seoung Wug Oh, Joon-Young Lee, and Seon Joo Kim. Vita: Video instance segmentation via object token association. In *NeurIPS*, 2022. 6

[18] Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from natural language expressions. In *ECCV*, 2016. 2

[19] Yutao Hu, Qixiong Wang, Wenqi Shao, Enze Xie, Zhenguo Li, Jungong Han, and Ping Luo. Beyond one-to-one: Rethinking the referring image segmentation. In *ICCV*, 2023. 2

[20] De-An Huang, Zhiding Yu, and Anima Anandkumar. Minvis: A minimal video instance segmentation framework without video-based training. In *NeurIPS*, 2022. 4

[21] Hueihan Jhuang, Juergen Gall, Silvia Zuffi, Cordelia Schmid, and Michael J Black. Towards understanding action recognition. In *ICCV*, 2013. 6

[22] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *NeurIPS*, 2020. 5

[23] Anna Khoreva, Anna Rohrbach, and Bernt Schiele. Video object segmentation with language referring expressions. In *ACCV*, 2018. 1, 2, 6, 7

[24] Namyup Kim, Dongwon Kim, Cuiling Lan, Wenjun Zeng, and Suha Kwak. Restr: Convolution-free referring image segmentation using transformers. In *CVPR*, 2022. 2

[25] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 1955. 4

[26] Dezhuang Li, Ruoqi Li, Lijun Wang, Yifan Wang, Jinqing Qi, Lu Zhang, Ting Liu, Qingquan Xu, and Huchuan Lu. You only infer once: Cross-modal meta-transfer for referring video object segmentation. In *AAAI*, 2022. 2

[27] Ruiyu Li, Kaican Li, Yi-Chun Kuo, Michelle Shu, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia. Referring image segmentation via recurrent refinement networks. In *CVPR*, 2018. 2

[28] Xiangtai Li, Henghui Ding, Wenwei Zhang, Haobo Yuan, Jiangmiao Pang, Guangliang Cheng, Kai Chen, Ziwei Liu, and Chen Change Loy. Transformer-based visual segmentation: A survey. *arXiv preprint arXiv:2304.09854*, 2023. 2

[29] Xiang Li, Jinglu Wang, Xiaohao Xu, Xiao Li, Bhiksha Raj, and Yan Lu. Robust referring video object segmentation with cyclic structural consensus. In *ICCV*, 2023. 1, 2, 3, 8

[30] Chen Liang, Yu Wu, Yawei Luo, and Yi Yang. Clawcranenet: Leveraging object-level relation for text-based video segmentation. *arXiv preprint arXiv:2103.10702*, 2021. 2

[31] Chenxi Liu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, and Alan Yuille. Recurrent multimodal interaction for referring image segmentation. In *ICCV*, 2017. 2

[32] Chang Liu, Xudong Jiang, and Henghui Ding. Instance-specific feature propagation for referring segmentation. *IEEE TMM*, 2022. 2

[33] Chang Liu, Henghui Ding, and Xudong Jiang. GRES: Generalized referring expression segmentation. In *CVPR*, 2023. 2

[34] Chang Liu, Henghui Ding, Yulun Zhang, and Xudong Jiang. Multi-modal mutual attention and iterative interaction for referring image segmentation. *IEEE TIP*, 32, 2023. 2

[35] Jiang Liu, Hui Ding, Zhaowei Cai, Yuting Zhang, Ravi Kumar Satzoda, Vijay Mahadevan, and R Manmatha. Polyformer: Referring image segmentation as sequential polygon generation. In *CVPR*, 2023. 2

[36] Si Liu, Tianrui Hui, Shaofei Huang, Yunchao Wei, Bo Li, and Guanbin Li. Cross-modal progressive comprehension for referring segmentation. *IEEE TPAMI*, 2022. 2

[37] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 6

[38] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *CVPR*, 2022. 3

[39] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[40] Zhuoyan Luo, Yicheng Xiao, Yong Liu, Shuyan Li, Yitong Wang, Yansong Tang, Xiu Li, and Yujiu Yang. Soc: Semantic-assisted object cluster for referring video object segmentation. In *NeurIPS*, 2023. 7, 8

[41] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016. 6

[42] Edgar Margffoy-Tuay, Juan C Pérez, Emilio Botero, and Pablo Arbeláez. Dynamic multimodal instance segmentation guided by natural language queries. In *ECCV*, 2018. 2

[43] Bo Miao, Mohammed Bennamoun, Yongsheng Gao, and Ajmal Mian. Spectrum-guided multi-granularity referring video object segmentation. In *ICCV*, 2023. 1, 2, 3, 8

[44] 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. 6

[45] Mengxue Qu, Yu Wu, Yunchao Wei, Wu Liu, Xiaodan Liang, and Yao Zhao. Learning to segment every referring object point by point. In *CVPR*, 2023. 2

[46] Sebastian Schuster, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D Manning. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the fourth workshop on vision and language*, pages 70–80, 2015. 4

[47] 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, 6, 7

[48] Jiajin Tang, Ge Zheng, Cheng Shi, and Sibei Yang. Contrastive grouping with transformer for referring image segmentation. In *CVPR*, 2023. 2

[49] Jiajin Tang, Ge Zheng, and Sibei Yang. Temporal collection and distribution for referring video object segmentation. In *ICCV*, 2023. 1, 2, 3, 8

[50] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 2008. 7

[51] 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. 2

[52] 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 *ICCV*, 2019. 2

[53] Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. Cris: Clip-driven referring image segmentation. In *CVPR*, 2022. 2

[54] Dongming Wu, Xingping Dong, Ling Shao, and Jianbing Shen. Multi-level representation learning with semantic alignment for referring video object segmentation. In *CVPR*, 2022. 2

[55] Dongming Wu, Tiancai Wang, Yuang Zhang, Xiangyu Zhang, and Jianbing Shen. Onlinerefer: A simple online baseline for referring video object segmentation. In *ICCV*, 2023. 1, 2, 3, 8

[56] Jiannan Wu, Yi Jiang, Peize Sun, Zehuan Yuan, and Ping Luo. Language as queries for referring video object segmentation. In *CVPR*, 2022. 1, 3, 4, 6, 7, 8

[57] Jianzong Wu, Xiangtai Li, Shilin Xu, Haobo Yuan, Henghui Ding, Yibo Yang, Xia Li, Jiangning Zhang, Yunhai Tong, Xudong Jiang, Bernard Ghanem, and Dacheng Tao. Towards open vocabulary learning: A survey. *IEEE TPAMI*, 2024. 2

[58] Yixuan Wu, Zhao Zhang, Chi Xie, Feng Zhu, and Rui Zhao. Advancing referring expression segmentation beyond single image. In *CVPR*, 2023. 2

[59] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018. 5

[60] Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, Hengshuang Zhao, and Philip HS Torr. Lavt: Language-aware vision transformer for referring image segmentation. In *CVPR*, 2022. 2

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

[62] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *ECCV*, 2016. 6

[63] 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. 2

[64] Seonghoon Yu, Paul Hongsuck Seo, and Jeany Son. Zero-shot referring image segmentation with global-local context features. In *CVPR*, 2023. 2

[65] Pengfei Zhu, Longyin Wen, Dawei Du, Xiao Bian, Heng Fan, Qinghua Hu, and Haibin Ling. Detection and tracking meet drones challenge. *IEEE TPAMI*, 2022. 2

[66] Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, et al. Generalized decoding for pixel, image, and language. In *CVPR*, 2023. 2