Exploiting Textual Semantics for Robust Cross-View Object Correspondence

Bing Fan **Yunhe Feng** **Yan Huang** **Heng Fan***
Department of Computer Science and Engineering, University of North Texas

Abstract

Cross-view object correspondence (CVOC) aims to establish associations of the same target between query and search videos captured from different viewpoints. Existing methods often rely on visual cues for CVOC. Despite recent progress, these methods struggle in complicated scenarios with substantial viewpoint changes and appearance variations, due to lacking sufficient target information. Addressing this, we introduce ***TeSCo***, a novel framework that exploits rich **Te**xtual **S**emantics of an object, in addition to its visual cues, for cross-view object **Co**rrespondence. Our *key insight* is, textual description of a target, by capturing diverse attributes such as category, texture, and color, provides *viewpoint-* and *appearance-invariant* semantics, which are complementary to visual cues and can hence enhance correspondence robustness in complicated scenes. Inspired by this, TeSCo first generates a textual expression of the target from the given query view using a vision-language model, and then extracts textual feature to enhance visual features for localization in the search view. To realize this, we introduce a simple yet effective *text-conditioned cross-modal fusion* (TCF) module that incorporates the textual feature into visual features with text-guided modulation, producing more robust multimodal target representation for object correspondence. Since not all textual cues are equally compatible with the target in current search-view frame, we propose an *iterative textual feature refinement* (ITFR) module that progressively adjusts textual feature using search-view information before the TCF module, enabling search-view-aware textual semantics to be fused into visual features and leading to better performance. In extensive experiments on Ego-Exo4D and HANDAL-X, TeSCo demonstrates state-of-the-art results and largely outperforms other models, validating its efficacy. Our code is released at https://github.com/VAIL-Fan/TeSCo.

1 Introduction

Cross-view object correspondence (CVOC) aims to establish associations of the same target across different viewpoints, *e.g.*, ego-centric and exo-centric views [10]. In specific, given a query-view video with frame-wise target mask annotations, the goal is to segment the same object in the corresponding frames of a search-view video. As an important problem in multi-view video understanding, CVOC enables many crucial applications, such as robot learning, augmented reality, human-object interaction understanding, thereby attracting growing interest in recent years (*e.g.*, [1, 8, 15, 23, 9, 40, 31]).

Current models predominantly rely on visual cues for CVOC. One line of work (*e.g.*, [1, 23, 9, 40, 31]) focuses on directly learning frame-level cross-view visual correspondence through extracting and matching visual features using image pairs across query and search views, while another line of work (*e.g.*, [15]) leverages both cross-view image pairs and temporal modeling of visual features within each viewpoint to enhance correspondence. Despite their progress, these approaches often struggle in challenging cross-view scenarios with substantial viewpoint and appearance changes, because visual

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

representations learned by existing methods are *inherently sensitive* to such visual variations, making it difficult to maintain consistent target representations across views when relying on only visual cues and resulting in degraded performance. This limitation motivates the search for complementary target information that is inherently more robust to viewpoint and appearance changes for CVOC.

In this paper, we leverage textual cues for CVOC. Our *key insight* is that, the textual description of an object (*e.g.*, "*a red basketball with black line, white stripe, and leather surface*") often captures rich high-level attributes, such as *category*, *color*, *material*, and *texture*. While the visual appearance of a target may vary drastically across views and over time in complicated scenarios, its attributes remain *consistent*, providing partially more stable semantic cues that are complementary to visual information and can therefore enhance the robustness of cross-view object correspondence.

Motivated by the above, we introduce ***TeSCo***, a novel framework that exploits rich **Te**xtual **S**emantics of the target, in addition to its visual cues, for cross-view object **Co**rrespondence. Specifically, TeSCo, is built on top of the popular SAM 2 model [34]. Its core idea is to leverage the textual information of the object to enhance spatial view and temporal memory features, and employ the resulted multimodal representation, which is more robust against the view and appearance variations, for cross-view object correspondence. Since the textual description is not provided in the standard CVOC task, we first generate a detailed textual description of the target from the given query-view frame using an off-the-shelf vision-language model, and then extract its textual feature to enhance visual features for correspondence. To realize this, we introduce a simple yet effective *text-conditioned cross-modal fusion* (TCF) module, which injects the textual information into visual features with text-guided modulation. TCF enables selective enhancement of target-relevant cues in fusion using text, resulting in more robust multimodal target representation for cross-view object correspondence. In addition, considering that not all textual semantics are equally compatible with the current search frame, we propose an *iterative textual feature refinement* (ITFR) module in TeSCo, which progressively adjusts the textual feature using information in search view before cross-modal fusion in the TCF module, producing search-view-aware textual semantics for more effective feature fusion. Through the ITFR and TCF modules, our TeSCo is able to exploit rich textual information of the target to enhance visual features for more robust CVOC. Fig. 1 illustrates the framework of TeSCo.

We note that an early approach [8] has also explored the use of text for cross-view correspondence. Although TeSCo shares the general idea of applying text, it substantially differs from [8]. *First*, the text in [8], containing only object name, is primarily utilized as part of prompt to instruct the large language model (LLM) for segmentation, whereas TeSCo adopts richer textual semantics, including category, color, texture, *etc.*, as feature-level guidance to explicitly enhance spatial and temporal visual features for correspondence. *Second*, different from [8] in applying cross-attention to merge text and visual embeddings, TeSCo introduces a dedicated TCF module that further refines feature fusion using text, enabling further enhancement of target-relevant cues for more robust multimodal representation. *Third*, unlike [8] directly using the text feature, we introduce an ITFR module to adjust the textual feature using search-view information, leading to search-view-aware textual semantics for better performance. Compared with [8], our TeSCo more effectively exploits textual semantics from a different perspective and achieves more robust performance, as demonstrated in later experiments.

To validate the effectiveness of TeSCo, we conduct extensive experiments on two benchmarks, including Ego-Exo4D [10] and HANDAL-X [13], and compare it with existing models. Our evaluation shows that TeSCo achieves new state-of-the-art results, with Intersection over Union (IoU) scores of **60.4%** in the ego→exo setting and **70.3%** in the exo→ego setting on the Ego-Exo4D and **90.0%** on the HANDAL-X, which largely surpass current methods and its baseline SAM 2, showing its efficacy.

In summary, our ***contributions*** are as follows: ♠ we propose a framework, named TeSCo, that exploits textual cues as semantic guidance to enhance CVOC and achieves state-of-the-art performance in extensive experiments on Ego-Exo4D and HANDAL-X; ♥ we present the TCF module, which injects textual semantics into visual features with text-guided modulation to enhance target-relevant cues; ♣ we further design an ITFR module, which adapts textual features with search-view information before fusion, producing search-view-aware textual semantics for more effective feature fusion.

2 Related Work

Cross-View Object Correspondence. Cross-view object correspondence (CVOC) aims to associate the same target between different viewpoints. Owing to its important applications, CVOC has recently

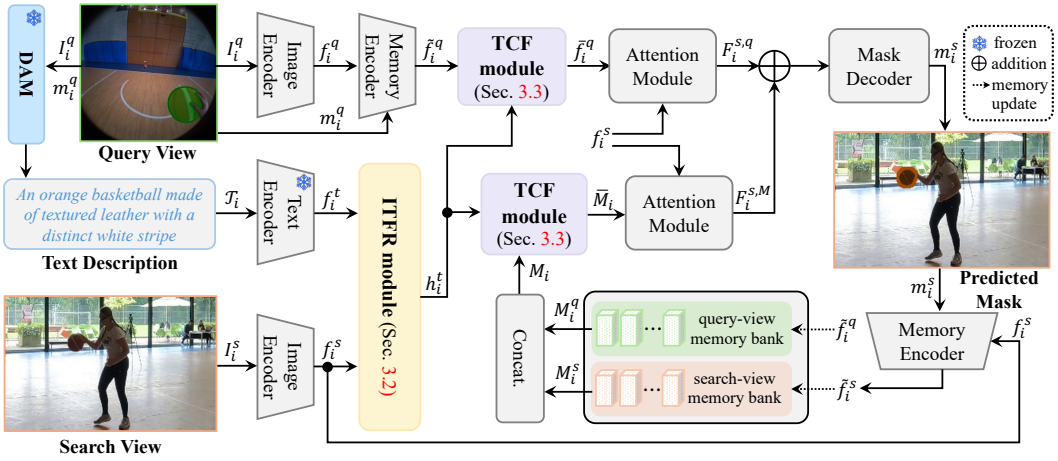


Figure 1: Overview of the proposed TeSCo that exploits rich textual information for robust cross-view correspondence through iterative textual feature refinement (ITFR) and text-conditioned cross-modality fusion (TCF) modules. *Best viewed in pdf for all figures throughout the paper.*

received extensive attentions with many methods proposed. The work of [1] designs a self-supervised framework for cross-view correspondence with predictive cycle consistency. The approaches in [23, 30] propose matching masks between different views to achieve object correspondence. The work in [40] presents cycle-consistency training to learn view-invariant visual representations for cross-view correspondence. The work of [8] utilizes a large language model with visual and text prompts for CVOC. The methods in [15, 31] explores SAM 2 [34] for cross-view segmentation. The model of [9] applies geometry and appearance features in [37] to learn robust representation for CVOC.

In this work, we introduce TeSCo that exploits rich textual semantics for cross-view correspondence. TeSCo is built upon SAM 2 by leveraging its excellent segmentation capacity. However, different from existing methods [15, 31] that explores visual cues only within the SAM 2 framework, TeSCo leverages both visual and textual cues for cross-view correspondence, leading to more robust performance. In addition, despite sharing a similar spirit, TeSCo is significantly different from [8] in how text is used and adapted to enhance CVOC as discussed earlier.

Visual Correspondence. Visual correspondence aims to establish associations of the same visual element across different frames or viewpoints, and is a crucial computer vision problem. Depending on the granularity of the association, visual correspondence can be divided into two groups, including point-level correspondence such as optical flow [29, 17] and point tracking [18, 19], which focuses on fine-grained matching of pixels, and object-level correspondence such as pedestrian search [39], video segmentation [34, 3] and tracking [24], and visual query localization [20, 6], which aims to associate the entire target. TeSCo falls in the object-level correspondence category. However, different from existing object correspondence methods, TeSCo focuses on associating the target in videos captured by different viewpoints, which is challenging due to extreme variations in appearance and viewpoint.

Text in Visual Localization. Textual description provides rich semantics of an object, and has been widely applied in various visual localization tasks, such as visual grounding [44, 42, 26], spatial-temporal video grounding [41, 11, 12], referring multi-object tracking [38, 21], referring expression segmentation [25, 16, 4], and vision-language tracking [14, 7], in which the text is offered to localize the target of interest in the image or video. Different from the above tasks, in this paper we focus on exploiting the textual information for improving cross-view object correspondence.

3 The Proposed Method

Task Definition. Cross-view object correspondence aims to associate the same target across different viewpoints. Formally, given a query-view (*e.g.*, ego-centric) sequence $\{I_i^q\}_{i=1}^N$ with N frames and masks $\{m_i^q\}_{i=1}^N$ of a target of interest in each frame, the goal is to segment the object masks $\{m_i^s\}_{i=1}^N$ of the same target in each frame of a synchronized search-view (*e.g.*, exo-centric) video $\{I_i^s\}_{i=1}^N$.

Overview. In this paper, we introduce TeSCo, which leverages textual information to enhance target representation for improving CVOC. TeSCo is built upon SAM 2 [34] for its excellent segmentation capacity. Specifically, as displayed in Fig. 1, TeSCo first generates a detailed textual description of the target from the given query-view frame, and then applies it to enhance the visual features in SAM 2 for target segmentation (§3.1). To realize this, two simple yet effective modules, including ITFR (§3.2) and TCF (§3.3), are introduced, with ITFR to refine the textual feature using information in current search-view frame and TCF to inject the refined textual information into visual features with text-guided modulation for learning multimodal features for segmentation.

3.1 Our TeSCo Framework

In the following, we detail TeSCo by describing how textual information is leveraged to segment the target object from the ith frame I_i^s in the search-view video.

Textual Description Generation. Since the text is *not* provided in the standard CVOC task, we first utilize a pre-trained vision-language model DAM [22] to generate a detailed textual description of the target from the given query-view frame. The reason for employing DAM is its excellent capability for detailed image region captioning. Specifically, given the ith frame I_i^q in the query-view video with the target mask m_i^q, the textual description of the target is generated as follows,

$$\mathcal{T}_i = \texttt{DAM}(I_i^q, m_i^q) \tag{1}$$

where $\mathcal{T}_i$ is the generated text in frame I_i^q, which covers category, color, texture, *etc*. Due to limited space, please refer to the ***appendix*** for examples of $\mathcal{T}_i$. It is worth noting that, although $\mathcal{T}_i$ is generated from the query view, its encoded textual semantics capture attributes that are relatively stable across views, and can thus assist target segmentation in the search view.

In this work, we generate the textual description $\mathcal{T}_i$ from each query-view frame I_i^q. This is because the target may reveal different visible attributes at different moments, and the frame-specific description can provide more suitable semantics for localization in the corresponding search-view frame I_i^s.

Exploiting Textual Semantics for CVOC. After generating $\mathcal{T}_i$, TeSCo exploits its textual semantics for improving CVOC. Specifically, TeSCo is built upon SAM 2 [34], and applies text to enhance the query-view and temporal memory features for target segmentation in I_i^s.

Given a pair of I_i^q and I_i^s, we first extract their features $f_i^q = \Phi(I_i^q)$ and $f_i^s = \Phi(I_i^s)$, where $\Phi(\cdot)$ is the image encoder [35] in SAM 2. After that, f_i^q, together with the object mask m_i^q, are applied to generate the object-aware query-view feature $\tilde{f}_i^q = \texttt{MemEnc}(f_i^q, m_i^q)$, where $\texttt{MemEnc}(\cdot,\cdot)$ denotes the memory encoder of SAM 2. Besides the query-view feature $\tilde{f}_i^q$, the temporal memory feature, formed by memory banks from both query and search views, is exploited in TeSCo. Let M_i^q and M_i^s denote query-view and search-view memory banks in frame i. For $i = 1$, both M_i^q and M_i^s are empty. Then, the temporal memory feature $M_i = \texttt{Concat}(M_i^q, M_i^s)$ is obtained by concatenating M_i^q and M_i^s.

Different from [15] directly using query-view and memory features for CVOC, we apply textual cues to enhance visual features and then use the resulted more robust multimodal features for segmentation. Specifically, given the textual description $\mathcal{T}_i$, we first extract its feature $f_i^t = \Psi(\mathcal{T}_i)$, where the text encoder $\Psi(\cdot)$ adopts RoBERTa [27]. Considering that not all textual cues in $\mathcal{T}_i$ are equally important in distinguishing the target in I_i^s due to the gap between different views, we apply ITFR (as described later in Sec. 3.2) to refine f_i^t using information in the search-view frame, as follows,

$$h_i^t = \texttt{ITFR}(f_i^t, f_i^s) \tag{2}$$

where h_i^t denotes the refined text feature, which focuses more on textual semantics that are compatible with the search-view frame I_i^s for segmentation. Afterwards, we inject the textual cues into $\tilde{f}_i^q$ and M_i through the TCF module (as explained later in Sec. 3.3), as follows,

$$\bar{f}_i^q = \texttt{TCF}(\tilde{f}_i^q, h_i^t) \qquad \bar{M}_i = \texttt{TCF}(M_i, h_i^t) \tag{3}$$

where $\bar{f}_i^q$ and $\bar{M}_i$ are enhanced multimodal query-view and memory features. Compared to $\tilde{f}_i^q$ and M_i, they are more robust against view and appearance changes, benefiting cross-view correspondence.

After this, we fuse $\bar{f}_i^q$ and $\bar{M}_i$ into the search-view feature f_i^s via attention module, as follows,

$$F_i^{s,q} = \texttt{MemAtt}(f_i^s, \bar{f}_i^q) \qquad F_i^{s,M} = \texttt{MemAtt}(f_i^s, \bar{M}_i) \tag{4}$$

where $\texttt{MemAtt}(\cdot,\cdot)$ denotes the memory attention module in SAM 2. The two outputs are combined and fed into the mask decoder to predict the target mask in the search-view frame, as follows,

$$m_i^s = \texttt{MaskDec}(F_i^{s,q} + F_i^{s,M}) \tag{5}$$

where m_i^s denotes the predicted target mask in I_i^s. After obtaining m_i^s, we generate the object-aware search-view feature $\tilde{f}_i^s$ via $\tilde{f}_i^s = \texttt{MemEnc}(f_i^s, m_i^s)$, and apply $\tilde{f}_i^s$ and $\tilde{f}_i^q$ to update the memory banks M_i^s and M_i^q to M_{i+1}^s and M_{i+1}^q with the first-in-first-out strategy for the $(i+1)^{\text{th}}$ frame.

3.2 Iterative Textual Feature Refinement (ITFR)

Accurate textual cues are important for robust target segmentation in the search-view frame. Although the text captures semantic attributes that are relatively stable across different viewpoints, some cues may become *less* applicable to the search view because of the cross-view appearance variations. For example, certain attributes such as texture in the query-view description may become ambiguous or even invisible in the search view. To deal with this issue, we propose an iterative textual feature refinement (ITFR) module, which progressively adjusts the textual feature using search-view information, enabling better adaption to current search-view context before being fused with visual features.

Specifically, ITFR comprises K iterations of textual feature refinement. In each iteration, the refined textual feature from last stage is interacted with search-view feature for further adjustment. Concretely, let t_i^k represent the input textual feature in iteration k ($1 \le k \le K$). In the first iteration ($k = 1$), t_i^1 is initialized via $t_i^1 = \texttt{Linear}(f_i^t)$, where $\texttt{Linear}(\cdot)$ is a linear projection layer that maps the channel dimension of the textual feature to match that of f_i^s.

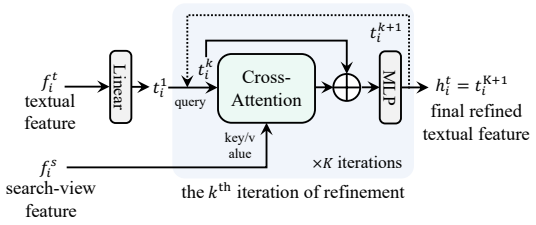


Figure 2: Illustration of the ITFR module.

To adapt the textual representation to the current search-view context, in each iteration, we first adopt cross-attention [36] to inject search-view information into the textual feature, and then employ a residual connection to preserve the original textual semantics while incorporating the search-view-guided refinement, followed by a multilayer perceptron (MLP) module. Mathematically, this process can be expressed as follows,

$$t_i^{k+1} = \texttt{MLP}(\texttt{CA}(t_i^k, f_i^s) + t_i^k) \qquad k = 1, 2, \cdots, K \tag{6}$$

where $\texttt{CA}(z, u)$ denotes cross-attention with z and u the query and key/value, respectively. After K iterations, the final refined textual feature h_i^t is obtained as $h_i^t = t_i^{K+1}$, which will be fed into the TCF module for multimodal fusion. Fig. 2 illustrates the ITFR for textual feature refinement.

It is worth ***noting*** that, while ITFR is implemented with a simple cross-attention-based structure, it effectively adapts the query-view textual representation to the current search-view context. Through ITFR, the textual feature progressively incorporates search-view visual information, suppressing less irrelevant attributes caused by cross-view variations and enhancing textual semantics that are more compatible with the current search-view frame.

3.3 Text-Conditioned Cross-Modality Fusion (TCF)

To improve cross-view object segmentation, we use textual semantics to enhance target representation. Specifically, after obtaining the refined textual feature, we fuse it into the query-view feature and the temporal memory feature, and employ the resulting multimodal features for target segmentation. To realize this, we introduce a simple yet effective text-conditioned cross-modality fusion (TCF) module, which injects the textual feature into the visual representation with text-guided modulation, enabling better enhancement of target-relevant cues in fusion. Fig. 3 illustrates the architecture of TCF.

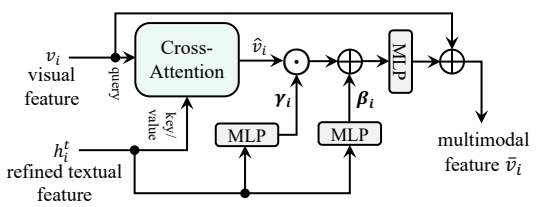
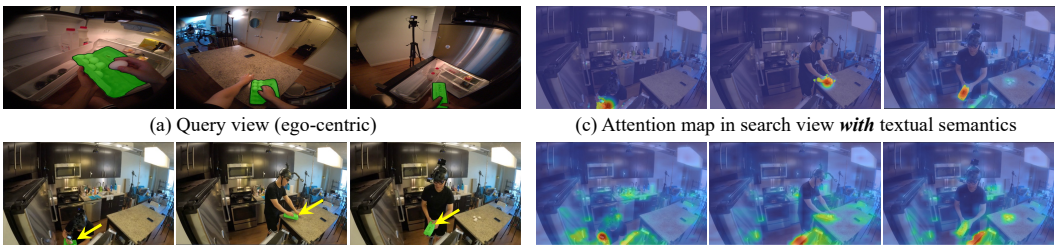


Figure 3: Illustration of the TCF module.



(a) Query view (ego-centric)

(c) Attention map in search view ***with*** textual semantics

(b) Search view (exo-centric)

(d) Attention map in search view ***without*** textual semantics

Figure 4: Image (a): query-view frames; image (b): search-view frames; image (c): attention maps in search view with textual semantics; and (d) attention maps in search view without textual semantics. The visualized attention maps is obtained by aggregating the attention maps from the two attention modules in TeSCo. We can observe that, the attention map with textual semantics can more accurately focus on the target region, evidencing the effectiveness of textual information.

Concretely, given the refined textual feature h_i^t and the visual feature v_i, where v_i represents either the query-view feature $\tilde{f}_i^q$ or the memory feature M_i, we first perform cross-attention to integrate the textual information into the visual feature as follows,

$$\widehat{v}_i = \texttt{CA}(v_i, h_i^t) \tag{7}$$

where $\widehat{v}_i$ denotes the multimodal target representation that contains both visual and textual information. To further enhance target-relevant cues in $\widehat{v}_i$, we leverage the textual feature h_i^t as a guidance to modulate the multimodal target feature. Specifically, inspired by conditional normalization [33, 5], we generate two text-conditioned vector parameters, $\boldsymbol{\gamma}_i$ and $\boldsymbol{\beta}_i$, to modulate $\widehat{v}_i$, with MLPs via

$$\boldsymbol{\gamma}_i = \texttt{MLP}(h_i^t) \qquad \boldsymbol{\beta}_i = \texttt{MLP}(h_i^t) \tag{8}$$

Here, $\boldsymbol{\gamma}_i$ is a channel-wise reweighting parameter that selectively emphasizes target-relevant cues in $\bar{v}_i$, while $\boldsymbol{\beta}_i$ acts as a shift parameter that further adapts the feature distribution according to textual semantics. With $\boldsymbol{\gamma}_i$ and $\boldsymbol{\beta}_i$, the final multimodal representation is then obtained via modulation with a residual connection, as follows,

$$\bar{v}_i = \texttt{MLP}(\widehat{v}_i \odot \boldsymbol{\gamma}_i + \boldsymbol{\beta}_i) + v_i \tag{9}$$

where $\odot$ denotes element-wise multiplication, and $\bar{v}_i$ the final multimodal feature, which corresponds $\bar{f}_i^q$ when TCF is applied to $\tilde{f}_i^q$, or $\bar{M}_i$ when TCF is applied to M_i. It is worth ***noting*** that, in practice, the vector parameters $\boldsymbol{\gamma}_i$ and $\boldsymbol{\beta}_i$ are broadcast along the spatial dimension in Eq. (9) to match the shape of $\widehat{v}_i$. Through this text-guided modulation, the resulting multimodal representation can better focus on target-relevant cues for cross-view segmentation.

Compared with the visual features $\tilde{f}_i^q$ and M_i, the multimodal features $\bar{f}_i^q$ and $\bar{M}_i$, by integrating textual semantics, which are view- and appearance invariant, are able to better preserve discriminative target cues under cross-view appearance variations, thereby improving the robustness of cross-view object correspondence. As demonstrated in Fig. 4, with textual semantics incorporated through TCF, the resulting attention map in the subsequent attention modules focuses more accurately on the target region, validating the effectiveness of textual semantics.

3.4 Optimization

Given a query-view video $\{I_i^q\}_{i=1}^N$ with object masks $\{m_i^q\}_{i=1}^N$ and a search-view sequence $\{I_i^s\}_{i=1}^N$, our TeSCo predicts the target mask m_i^s in each search-view frame I_i^s. During training, we follow SAM 2 [34], and employ multiple losses, including dice coefficient loss, focal mask loss, IoU loss, and binary cross-entropy loss to supervise the predicted mask m_i^s in each search-view frame. The final training objective is obtained by summing the losses over all predicted search-view masks. Because of the limitation of space, please kindly refer to SAM 2 [34] for more details and implementations of the loss function for training.

4 Experiments

Implementation. TeSCo is built upon SAM 2 [34], and introduces two dedicated modules, TCF and ITFR, to exploit the textual cues for cross-view correspondence. It is implemented using PyTorch [32]

Table 1: Results of Ego→Exo and Exo→Ego on the Ego-Exo4D test set. The best and the second-best results are highlighted in **bold** and underlined in all state-of-the-art comparison tables.

Method	Ego→Exo			Exo→Ego			mIoU (↑)
	IoU (↑)	Cont.A (↑)	Loc.E (↓)	IoU (↑)	Cont.A (↑)	Loc.E (↓)	
XSegTx[CVPR'24] [10]	18.90	0.386	0.070	27.10	0.358	0.104	23.00
XMem[CVPR'24] [10]	19.30	0.262	0.151	16.60	0.240	0.160	17.95
XMem+XSegTx[CVPR'24] [10]	34.90	0.559	0.038	25.00	0.237	0.117	29.95
PCC[CVPR'25] [1]	39.01	0.600	0.033	47.06	0.590	0.054	43.04
ObjectRelator[ICCV'25] [8]	35.30	0.540	0.036	40.30	0.500	0.068	37.80
O-MaMa (k-NN)[ICCV'25] [30]	31.90	0.414	0.195	30.90	0.373	0.127	31.40
O-MaMa[ICCV'25] [30]	42.60	0.590	0.033	44.10	0.524	0.082	43.35
CCMP[CVPR'26] [40]	41.95	0.669	0.038	47.18	0.591	0.081	44.57
V^2-SAM[CVPR'26] [31]	46.30	0.616	0.056	49.60	0.575	0.078	47.95
LM-EEC[NeurIPS'25] [15]	54.98	0.778	0.017	65.77	0.774	0.031	60.38
Baseline	52.70	0.719	0.027	58.30	0.664	0.043	55.50
TeSCo (ours)	**60.40**	**0.847**	**0.011**	**70.30**	**0.808**	**0.023**	**65.35**

and trained with 4 Nvidia A6000 GPUs. Following [34], we utilize the HieraB+ [35] as the visual encoder to extract visual features. For the text, we apply RoBERTa [27] to extract its feature. Our TeSCo is end-to-end trained for 60 epoches with a batchsize of 12 (3 per GPU) using the AdamW optimizer [28]. The learning rates for the image encoder and our new modules/decoder are 6×10^{-6} and 10^{-5}, respectively. During training and inference, all video frames are resized to 480×480 because of GPU memory constraints. The number of iteration K in ITFR is empirically set to 2. The loss function for training is kept the same as in [34]. Code and model will be released.

Datasets. We use Ego-Exo4D [10] and HANDAL-X [13] for experiments. Ego-Exo4D [10] contains 1,335 videos takes and around 1.8M annotated object masks (742K in the ego-centric view and 1.1M in the exo-centric view), covering diverse real-world scenarios such as basketball, bike Repair, cooking, health, and music. Following previous works [15, 40, 31], we employ the official training/test split in Ego-Exo4D and adopt two settings, Ego→Exo (Ego-centric view as query) and Exo→Ego (Exo-centric view as query), for experiments. HANDAL-X [13], introduced in [8], consists of real-world manipulable objects with 6-DoF pose labels and is used to further assess the generalization ability of different models. It contains 44,102 training pairs and 14,074 test pairs for experiments.

Evaluation Metrics. Following [31], we use *Intersection over Union* (IoU), *Location Error* (Loc.E), and *Contour Accuracy* (Cont.A), provided in [10], as the evaluation metrics on Ego-Exo4D. According to Ego-Exo4D Correspondence Challenge [10], the primary evaluation metric is the mean Intersection-over-Union (mIoU) of both Ego→Exo and Exo→Ego tasks. Please refer to [10] for the details of these metrics. On HANDAL-X, following prior works [8, 40], we utilize IoU for evaluation.

4.1 State-of-the-art Comparison

Results on Ego-Exo4D. To verify the effectiveness of TeSCo, we evaluate it on Ego-Exo4D with comparison against existing state-of-the-art methods and its baseline. The baseline is implemented by removing the ITFR and TCF modules form TeSCo. Due to limited space, we display the architecture of the baseline in the ***appendix***.

Tab. 1 presents the results for Ego→Exo and Exo→Ego. As shown, TeSCo consistently achieves the best performance across all metrics in both settings. Specifically, it achieves IoU scores of 60.40% for Ego→Exo and 70.30% for Exo→Ego, with an overall mIoU of 65.35%, significantly outperforming the second best method, LM-EEC [15], which achieves 54.98% and 65.77% IoU scores and a mean IoU of 60.38%. Notably, although LM-EEC is also built upon SAM 2 [34], it primarily relies on long-term visual cues. The superior performance of TeSCo over LM-EEC highlights the effectiveness of incorporating textual semantics for CVOC. Compared to our baseline obtaining 52.70% and 58.30% IoU for Ego→Exo and Exo→Ego, respectively, with an mIoU of 55.50%, TeSCo achieves absolute gains of 7.70%, 12.0%, and 9.85%, further demonstrating the benefit of text. In addition, compared to ObjectRelator [8], which also uses textual information and achieves 35.30% and 40.30% IoU with an mIoU of 37.8%, TeSCo yields substantial improvements of 25.10%, 23.24%, and 22.31%, evidencing the advantage of our design in exploiting text for object correspondence.

Results on HANDAL-X. We further evaluate TeSCo on HANDAL-X. Following [40, 8], we conduct experiments in zero-shot learning and retraining. In the zero-shot learning setting, the methods are trained on Ego-Exo4D and tested on HANDAL-X; in the retraining setting, the methods are both trained and evaluated on HANDAL-X. Tab. 2 reports the results and comparison.

Table 2: Results on the HANDAL-X test set.

Method	Dataset	IoU (↑)
(a) zero-shot learning		
PSALM[ECCV'24] [43]	Ego-Exo4D	39.90
ObjectRelator[ICCV'25] [8]	Ego-Exo4D	42.80
CCMP[CVPR'26] [40]	Ego-Exo4D	<u>78.80</u>
V^2-SAM[CVPR'26] [31]	Ego-Exo4D	77.20
Baseline	Ego-Exo4D	75.26
TeSCo (ours)	Ego-Exo4D	**85.10**
(b) retraining		
PSALM[ECCV'24] [43]	HANDAL-X	83.40
ObjectRelator[ICCV'25] [8]	HANDAL-X	84.70
CCMP[CVPR'26] [40]	HANDAL-X	<u>85.00</u>
Baseline	HANDAL-X	83.94
TeSCo (ours)	HANDAL-X	**90.00**

As shown in Tab. 2 (a), although TeSCo is not trained on HANDAL-X, it achieves the best performance with an IoU of 85.10%, surpassing the second best CCMP [40] with 78.80% IoU by 6.30% and the baseline with 75.26% IoU by 9.84%, demonstrating strong cross-view generalization ability. When training with HANDAL-X, as shown in Tab. 2 (b), the IoU score of TeSCo is further improved to 90.0%, outperforming all other methods and its baseline. These results consistently demonstrate the effectiveness of TeSCo and the importance of textual information for robust cross-view correspondence.

4.2 Ablation Study

To better understand TeSCo, we conduct extensive ablation studies in Ego→Exo on the test set of Ego-Exo4D. Our final configuration in each ablation is highlighted in light blue.

Impact of TCF and ITFR. TCF and ITFR are key components in TeSCo. To verify their effectiveness, we conduct ablation studies in Tab. 3. Please note that ITFR cannot function alone in TeSCo. Since it is designed to adapt textual features for fusion, it must be used together with TCF. As shown in Tab. 3, without TCF and ITFR, our baseline achieves an IoU score of 52.70% (❶). When TCF is applied to incorporate textual semantics, the IoU score is significantly improved to 58.74% with 6.04% absolute gains (❶ *v.s.* ❷). When using our ITFR to adjust the textual feature, the performance of TeSCo is further improved to 60.40% (❶ *v.s.* ❸), indicating that better textual semantics could benefit more cross-view correspondence and the necessity of ITFR. These results clearly validate the efficacy of TCF and ITFR in exploiting textual semantics for improving CVOC.

Comparison of TCF and other fusion strategies. TCF is presented to incorporate textual semantics by fusing textual features into visual representations. We compare TCF with two alternative fusion methods, including addition-based fusion and cross-attention-based fusion. The former simply adds the textual and visual features, while the latter injects textual information into visual features using standard cross-attention [36]. The results are reported in Tab. 4. As shown, addition-based fusion achieves an IoU score of 56.43% (❶). When utilizing cross-attention for fusion, the IoU score is improved to 58.26% (❷). When applying TCF in TeSCo, we achieve the best performance with the IoU score of 60.40% (❸), demonstrating the superiority of our TCF.

Impact of design choice in TCF. In TeSCo, TCF applies the text as a condition to generate two key parameters, including reweight parameter γ_i and shift parameter β_i, to further enhance the multimodal feature (see Eq. (9)). We conduct ablation studies on γ_i and β_i in Tab. 5. As shown, when removing γ_i and β_i, TCF degenerates into the standard cross-attention and achieves an IoU score of 58.26% (❶). When adopting γ_i alone to reweight the feature, the IoU score is improved to 59.73% (❶ *v.s.* ❷). When using β_i alone to further inject textual guidance, the IoU score is enhanced to 58.80% (❶ *v.s.* ❸). When applying both γ_i and β_i, we achieve the best IoU score of 60.40% (❶ *v.s.* ❹). All these results evidence the effectiveness of our design choice using γ_i and β_i in TCF.

Impact of number iterations in ITFR. ITFR aims to iteratively refine the text feature using search view information in a progressive way. We conduct an ablation on the number of iterations K in ITFR. As shown in Tab. 6, when setting $K = 1$, which means adjusting the textual once, the IoU score is 59.90% (❶). When further refining the textual feature with another iteration, *i.e.*, $K = 2$, the IoU score is improved to 60.40% (❷), indicating further interaction enables better search-view-aware textual semantics. However, when setting $K = 3$, the IoU score is slightly decreased to 59.29% (❸), probably caused by over-refinement of the textual feature. Therefore, we set $K = 2$ in this work.

Table 3: Ablation on TCF and ITFR in TeSCo.

	TCF	ITFR	IoU (↑)	Cont.A (↑)	Loc.E (↓)
❶	-	-	52.70	0.719	0.027
❷	✓	-	58.74	0.798	0.016
❸	✓	✓	60.40	0.847	0.011

Table 4: Ablation on different fusion strategies.

		IoU (↑)	Cont.A (↑)	Loc.E (↓)
❶	Addition	56.43	0.781	0.018
❷	Cross-Attention	58.26	0.792	0.016
❸	Our TCF	60.40	0.847	0.011

Table 5: Ablation on design choice in TCF.

	γ_i	β_i	IoU (↑)	Cont.A (↑)	Loc.E (↓)
❶	-	-	58.26	0.792	0.016
❷	✓	-	59.73	0.827	0.013
❸	-	✓	58.80	0.800	0.015
❹	✓	✓	60.40	0.847	0.011

Table 6: Ablation on the number of iterations for text adaptation in ITFR.

	Num. of Iterations	IoU (↑)	Cont.A (↑)	Loc.E (↓)
❶	$K = 1$	59.90	0.833	0.012
❷	$K = 2$	60.40	0.847	0.011
❸	$K = 3$	59.29	0.794	0.017

Impact on Text Injection Location. In TeSCo, we inject the textual cues to both query view and temporal memory. We conduct an ablation on injecting text to different locations in TeSCo, and report the results in Tab. 7. As shown in Tab. 7, injecting text into query view only yields 55.28% IoU score (❶), while injecting text into memory alone improves to 58.94% score (❷). When injecting text to both query-view feature and temporal memory, we achieve the best IoU score of 60.40% (❸). Therefore, in this work, we inject the textual cues into both query view and temporal memory features.

Table 7: Ablation on text injection location.

	Inj. to Query View	Inj. to Mem.	IoU (↑)	Cont.A (↑)	Loc.E (↓)
❶	✓	-	55.28	0.753	0.022
❷	-	✓	58.94	0.804	0.015
❸	✓	✓	60.40	0.847	0.011

Impact of Text Generation Methods. We adopt DAM [22] for text generation. To show its effectiveness, we experiment TeSCo with Qwen2.5-VL-7B [2] using respectively box and mask for text generation. As in Tab. 8, using DAM for text generation achieves better IoU score of 60.40% (❸). This is probably because DAM is trained for multi-granular region captioning, making it better suited for generating text. We further compare DAM with manual text provided by human. For each video, a global textual description is labeled. As shown, TeSCo with DAM achieves performance very close to that with manual text (❸ *v.s.* ❹), evidencing again the effectiveness of DAM for TeSCo.

Table 8: Ablation on TeSCo with different text generation methods. "b": box; "m": mask.

	Text Generation	IoU (↑)	Cont.A (↑)	Loc.E (↓)
❶	Qwen2.5-VL + b	56.28	0.766	0.020
❷	Qwen2.5-VL + m	59.75	0.828	0.013
❸	DAM + m	60.40	0.847	0.011
❹	Manual Text	61.20	0.830	0.010

Due to limited space, please refer to our ***appendix*** for more experiments and analysis.

5 Conclusion

This paper introduces TeSCo, that exploits rich textual semantics of the target to improve CVOC. Specifically, TeSCo first generates a textual description of the target from the given query view, and then integrates the textual information into visual features for segmentation. To realize this, we design the iterative textual feature refinement (ITFR) and text-conditioned cross-modal fusion (TCF) modules in TeSCo. The former aims to adjust the textual feature using search-view information, while the latter works to incorporate the textual cues into visual features with text-guided modulation. In experiments on two datasets, TeSCo achieves state-of-the-art results, validating its efficacy.

Limitation. Despite achieving promising results, TeSCo has two main limitations. First, it relies on frame-level descriptions, which may be degraded under occlusion or motion blur. In the future, we will explore temporal information in videos to generate more stable target descriptions. Second, TeSCo introduces additional computational resource and cost from text generation and textual feature extraction. In the future, we will optimize our method for more resource-friendly and efficient CVOC.

Acknowledgments and Disclosure of Funding

This work receives no external funding. The authors declare no competing interests.

References

[1] Alan Baade and Changan Chen. Self-supervised cross-view correspondence with predictive cycle consistency. In *CVPR*, 2025.

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv*, 2025.

[3] Ho Kei Cheng and Alexander G Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *ECCV*, 2022.

[4] Yong Xien Chng, Henry Zheng, Yizeng Han, Xuchong Qiu, and Gao Huang. Mask grounding for referring image segmentation. In *CVPR*, 2024.

[5] Harm De Vries, Florian Strub, Jérémie Mary, Hugo Larochelle, Olivier Pietquin, and Aaron C Courville. Modulating early visual processing by language. *NIPS*, 2017.

[6] Bing Fan, Yunhe Feng, Yapeng Tian, James Chenhao Liang, Yuewei Lin, Yan Huang, and Heng Fan. Prvql: Progressive knowledge-guided refinement for robust egocentric visual query localization. In *ICCV*, 2025.

[7] Xiaokun Feng, Xuchen Li, Shiyu Hu, Dailing Zhang, Meiqi Wu, Jing Zhang, Xiaotang Chen, and Kaiqi Huang. Memvlt: Vision-language tracking with adaptive memory-based prompts. In *NeurIPS*, 2024.

[8] Yuqian Fu, Runze Wang, Bin Ren, Guolei Sun, Biao Gong, Yanwei Fu, Danda Pani Paudel, Xuanjing Huang, and Luc Van Gool. Objectrelator: Enabling cross-view object relation understanding across ego-centric and exo-centric perspectives. In *ICCV*, 2025.

[9] Yulu Gao, Bohao Zhang, Zongheng Tang, Jitong Liao, Wenjun Wu, and Si Liu. Vggt-segmentor: Geometry-enhanced cross-view segmentation. In *CVPR*, 2026.

[10] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *CVPR*, 2024.

[11] Xin Gu, Heng Fan, Yan Huang, Tiejian Luo, and Libo Zhang. Context-guided spatio-temporal video grounding. In *CVPR*, 2024.

[12] Xin Gu, Yaojie Shen, Chenxi Luo, Tiejian Luo, Yan Huang, Yuewei Lin, Heng Fan, and Libo Zhang. Knowing your target: Target-aware transformer makes better spatio-temporal video grounding. In *ICLR*, 2025.

[13] Andrew Guo, Bowen Wen, Jianhe Yuan, Jonathan Tremblay, Stephen Tyree, Jeffrey Smith, and Stan Birchfield. Handal: A dataset of real-world manipulable object categories with pose annotations, affordances, and reconstructions. In *IROS*, 2023.

[14] Mingzhe Guo, Zhipeng Zhang, Heng Fan, and Liping Jing. Divert more attention to vision-language tracking. In *NeurIPS*, 2022.

[15] Yijun Hu, Bing Fan, Xin Gu, Haiqing Ren, Dongfang Liu, Heng Fan, and Libo Zhang. Robust ego-exo correspondence with long-term memory. In *NeurIPS*, 2025.

[16] 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.

[17] Wonjoon Jin, Qi Dai, Chong Luo, Seung-Hwan Baek, and Sunghyun Cho. Flovd: Optical flow meets video diffusion model for enhanced camera-controlled video synthesis. In *CVPR*, 2025.

[18] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *ECCV*, 2024.

[19] Nikita Karaev, Yuri Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker3: Simpler and better point tracking by pseudo-labelling real videos. In *ICCV*, 2025.

[20] Savya Khosla, Sethuraman TV, Alexander Schwing, and Derek Hoiem. Relocate: A simple training-free baseline for visual query localization using region-based representations. In *CVPR*, 2025.

[21] Guangyao Li, Siping Zhuang, Yajun Jian, Yan Yan, and Hanzi Wang. Language decoupling with fine-grained knowledge guidance for referring multi-object tracking. In *ICCV*, 2025.

[22] Long Lian, Yifan Ding, Yunhao Ge, Sifei Liu, Hanzi Mao, Boyi Li, Marco Pavone, Ming-Yu Liu, Trevor Darrell, Adam Yala, et al. Describe anything: Detailed localized image and video captioning. In *ICCV*, 2025.

[23] Jitong Liao, Yulu Gao, Shaofei Huang, Jialin Gao, Jie Lei, Ronghua Liang, and Si Liu. Domr: Establishing cross-view segmentation via dense object matching. In *ACM MM*, 2025.

[24] Liting Lin, Heng Fan, Zhipeng Zhang, Yaowei Wang, Yong Xu, and Haibin Ling. Tracking meets lora: Faster training, larger model, stronger performance. In *ECCV*, 2024.

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

[26] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *ECCV*, 2024.

[27] 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.

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

[29] Yawen Lu, Dongfang Liu, Qifan Wang, Cheng Han, Yiming Cui, Zhiwen Cao, Xueling Zhang, Yingjie Victor Chen, and Heng Fan. Promotion: Prototypes as motion learners. In *CVPR*, 2024.

[30] Lorenzo Mur-Labadia, Maria Santos-Villafranca, Jesus Bermudez-Cameo, Alejandro Perez-Yus, Ruben Martinez-Cantin, and Jose J Guerrero. O-mama: Learning object mask matching between egocentric and exocentric views. In *ICCV*, 2025.

[31] Jiancheng Pan, Runze Wang, Tianwen Qian, Mohammad Mahdi, Yanwei Fu, Xiangyang Xue, Xiaomeng Huang, Luc Van Gool, Danda Pani Paudel, and Yuqian Fu. V^2-sam: Marrying sam2 with multi-prompt experts for cross-view object correspondence. In *CVPR*, 2026.

[32] 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.

[33] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

[34] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloé Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross B. Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. In *ICLR*, 2025.

[35] Chaitanya Ryali, Yuan-Ting Hu, Daniel Bolya, Chen Wei, Haoqi Fan, Po-Yao Huang, Vaibhav Aggarwal, Arkabandhu Chowdhury, Omid Poursaeed, Judy Hoffman, et al. Hiera: A hierarchical vision transformer without the bells-and-whistles. In *ICML*, 2023.

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

[37] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *CVPR*, 2025.

[38] Dongming Wu, Wencheng Han, Tiancai Wang, Xingping Dong, Xiangyu Zhang, and Jianbing Shen. Referring multi-object tracking. In *CVPR*, 2023.

[39] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaogang Wang. Joint detection and identification feature learning for person search. In *CVPR*, 2017.

[40] Shannan Yan, Leqi Zheng, Keyu Lv, Jingchen Ni, Hongyang Wei, Jiajun Zhang, Guangting Wang, Jing Lyu, Chun Yuan, and Fengyun Rao. Learning cross-view object correspondence via cycle-consistent mask prediction. In *CVPR*, 2026.

[41] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Tubedetr: Spatio-temporal video grounding with transformers. In *CVPR*, 2022.

[42] Li Yang, Yan Xu, Chunfeng Yuan, Wei Liu, Bing Li, and Weiming Hu. Improving visual grounding with visual-linguistic verification and iterative reasoning. In *CVPR*, 2022.

[43] Zheng Zhang, Yeyao Ma, Enming Zhang, and Xiang Bai. Psalm: Pixelwise segmentation with large multi-modal model. In *ECCV*, 2024.

[44] Chaoyang Zhu, Yiyi Zhou, Yunhang Shen, Gen Luo, Xingjia Pan, Mingbao Lin, Chao Chen, Liujuan Cao, Xiaoshuai Sun, and Rongrong Ji. Seqtr: A simple yet universal network for visual grounding. In *ECCV*, 2022.

Appendix

To better understanding of this work, we offer additional details, analysis, and results as follow:

- **A Examples of Generated Textual Description**
  In this section, we show several examples of the generated textual descriptions of target.
- **B Architectures of Our Baseline**
  In this section, we show the detailed architecture of the baseline.
- **C Performance on Different Scenarios**
  In this section, we show the results on different scenarios in Ego-Exo4D.
- **D More Ablation Studies**
  In this section, we present more ablation studies of our method.
- **E Additional Results and Discussions**
  In this section, we present additional results and discussions.
- **F Complexity Analysis**
  In this section, we provide analysis of the model complexity for our proposed method and its comparisons with other state-of-the-art approaches.
- **G Qualitative Analysis**
  This section displays qualitative results of our method on different benchmarks.
- **H Broader Impact**
  This section discusses the broader impact of our method.

A Examples of Generated Textual Description

Since text is not offered in the standard CVOC task, we apply an off-the-shelf vision-language model, DAM [22], to generate the textual description of the target for its excellent captioning ability. Fig. 5 displays several examples of generated textual descriptions of the targets.

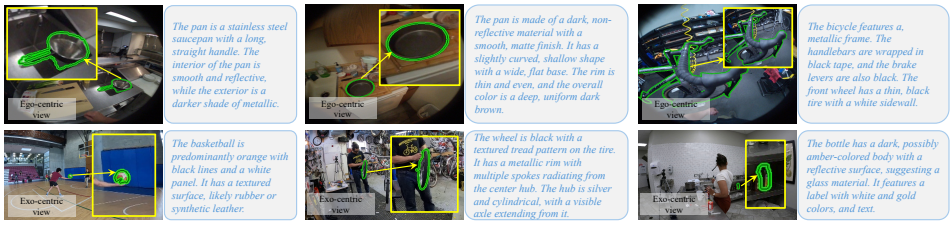


Figure 5: Examples of generated textual description.

B Architecture of Our Baseline

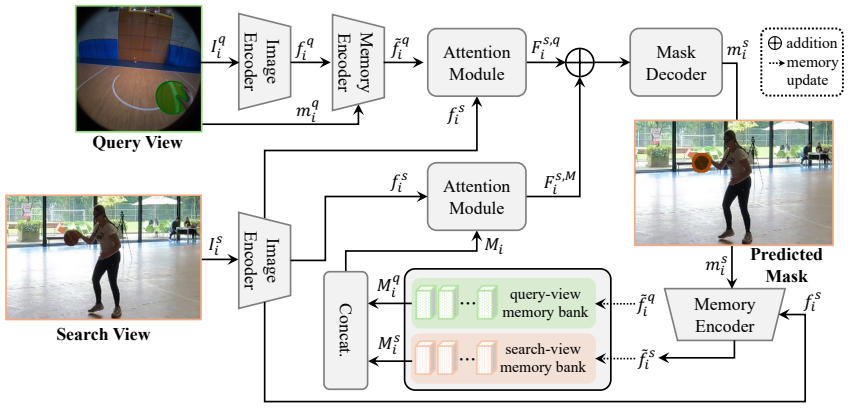


Figure 6: Illustration of the detailed architecture of our baseline.

Table 9: Performance on different scenarios in Ego-Exo4D.

Scenario	Ego→Exo						Exo→Ego					
	Baseline			TeSCo			Baseline			TeSCo		
	IoU (↑)	Cont.A (↑)	Loc.E (↓)	IoU (↑)	Cont.A (↑)	Loc.E (↓)	IoU (↑)	Cont.A (↑)	Loc.E (↓)	IoU (↑)	Cont.A (↑)	Loc.E (↓)
Cooking	44.93	0.685	0.027	57.29	0.829	0.131	53.76	0.665	0.005	68.54	0.805	0.025
Music	72.72	0.789	0.020	88.93	0.940	0.010	60.96	0.708	0.034	74.55	0.843	0.017
Health	48.07	0.669	0.019	61.93	0.847	0.008	57.52	0.678	0.041	74.11	0.859	0.018
Basketball	70.35	0.875	0.019	79.32	0.956	0.007	80.08	0.889	0.010	90.29	0.971	0.004
Bike Repair	26.39	0.427	0.068	53.08	0.774	0.021	30.75	0.403	0.122	61.85	0.732	0.038
Soccer	53.24	0.672	0.026	72.06	0.911	0.008	58.47	0.686	0.039	71.38	0.853	0.018

This section illustrates the architecture of our baseline, which, based on the SAM 2 [34], exploits visual features, *i.e.*, the query-view feature and temporal memory feature, for cross-view segmentation. Compared to TeSCo, the textual description generation, textual feature extraction, ITFR and TCF modules in the baseline are removed. Please refer to Fig. 6 for the framework of the baseline.

C Performance on Different Scenarios

The Ego-Exo4D benchmark [10] comprises diverse scenarios, including *Cooking*, *Music*, *Health*, *Basketball*, *Bike Repair*, and *Soccer*. In order to further analyze our method, we show the performance on different scenarios. Tab. 9 reports the results of TeSCo and our baseline. From Tab. 9, we can see that, TeSCo significantly the baseline model on all scenarios, showing the effectiveness of exploiting textual semantics in improving cross-view correspondence.

D More Ablation Studies

In this section, we show more ablation studies of our method in Ego→Exo on the test set of Ego-Exo4D. Our final configuration in each ablation is highlighted in light blue.

Textual Description Generation Interval. In this work, we generate the textual description from each query-view frame. This is because the target may reveal different visible attributes at different moments, and the frame-specific description can provide more suitable semantics for localization in the corresponding search-view frame. To validate this strategy, we vary the textual description generation interval r among 1, 5, and 10 frames. An interval of 1 ($r = 1$) corresponds to per-frame generation, whereas for intervals of 5 ($r = 5$) and 10 ($r = 10$), textual descriptions are generated every 5 and 10 frames, respectively, and the resulting textual features are reused for the subsequent intermediate frames until the next description is generated. Tab. 10 reports the results under the Ego→Exo setting. As shown, generating textual descriptions at every frame achieves the best performance (❶), while increasing the interval to 5 or 10 frames leads to a gradual performance drop (❷ and ❸). This suggests that frame-specific textual descriptions benefits more the cross-view object correspondence.

Table 10: Results with different textual description generation interval r.

	Generation Interval	IoU (↑)	Cont.A (↑)	Loc.E (↓)
❶	$r = 1$	60.40	0.847	0.011
❷	$r = 5$	59.10	0.835	0.012
❸	$r = 10$	59.00	0.835	0.013

Impact of Memory Banks. In TeSCo, we employ both query-view and search-view memory banks to construct temporal memory features for robust cross-view object correspondence. To examine the impact of this dual-memory design, we conduct an ablation study by using only the query-view memory bank, only the search-view memory bank, and both memory banks. As shown in Tab. 11, using both memory banks achieves the best performance, outperforming the variants that use only one of them. Specifically, using only the query-view memory bank yields an IoU of 55.47% (❶), while using only the search-view memory bank improves the IoU to 58.78% (❷). When both memory banks are applied, TeSCo further achieves the best IoU of 60.40% (❸). These results indicate justifies the dual-memory design in TeSCo.

Table 11: Impact of memory banks.

	Memory Bank	IoU (↑)	Cont.A (↑)	Loc.E (↓)
❶	query-view bank	55.47	0.751	0.022
❷	search-view bank	58.78	0.800	0.016
❸	using both banks	60.40	0.847	0.017

E Additional Results and Discussions

Targeted Stress Test. To better understand our TeSCo, we perform textual and visual stress tests in Ego→Exo on Ego4D.

For textual stress tests, we construct three controlled textual perturbation settings: a) noisy captions, by randomly inserting one irrelevant semantic attribute into the description; b) wrong captions, by randomly replacing one key semantic attribute (*e.g.*, category, color, or texture) with an incorrect one while preserving the remaining content; and c) partial captions, by randomly removing one key semantic attribute (*e.g.*, category, color, or texture). Tab. 12 reports the results on textual stress tests. From Tab. 12, we can observe that, although the performance decreases under challenging textual conditions, our method still outperforms the baseline, showing its robustness.

Table 12: Results on textual stress tests.

	Method	Text Condition	IoU (↑)
❶	Baseline	No text	52.70
❷	TeSCo	Original	60.40
❸	TeSCo	Noisy	59.22
❹	TeSCo	Wrong	56.20
❺	TeSCo	Partial	57.50

For visual stress tests, we manually construct two subsets, Occlusion and Distractor subsets, from the full test set by inspecting all videos. Specifically, the Occlusion subset contains videos with occlusion, where the queried target is occluded by other objects. The Distractor subset contains videos with distractors which have similar appearance with the queried target. Tab. 13 shows the results challenging visual sub- sets. We can see that, in challenging scenarios, both TeSCo and its baseline are degraded. However, TeSCo still significantly outperform its baseline, evidencing the effectiveness of textual semantics and the robustness of our method.

Table 13: Results on challenging visual subsets using IoU.

	Evaluation Set	Baseline	TeSCo
❶	Full test set	52.70	60.40
❷	Occlusion subset	48.25	58.50
❸	Distractor subset	47.60	57.33

Validation of Generality. To validate the generality of our method, we conducting experiments by integrating the same textual modules of TeSCo, including, DAM-generated text, ITFR, and TCF, to LM-EEC [15]. The results in Ego→Exo on Ego4D is shown in Tab. 14. As shown, the performance improves from 54.98% to 59.25%, demonstrating that the gain mainly comes from textual semantics.

Table 14: Results of integrating our textual modules into LM-EEC.

	Method	IoU
❶	LM-EEC	54.98
❷	LM-EEC + Our Textual Modules	59.25

Discussion of Textual Semantics in Challenging Scenarios. In TeSCo, we leverage textual semantics of the target for cross-view segmentation. In challenging scenarios with the presence of similar objects, the discriminative power of text may decrease. However, in TeSCo, textual semantics serve as complementary guidance rather than the sole cue. The final correspondence is jointly determined by both the visual features and the textual semantics. Therefore, even when the textual descriptions become less discriminative, the visual information can provide fine-grained cues, and these two cues together can help localize the target object for robust segmentation.

Discussion on Failure Cases. Although TeSCo achieves promising overall results, it may encounter failure cases in challenging scenarios, such as severe occlusion, visually similar distractors, and very small targets, as shown in Fig. 7. In these cases, reliable visual cues become insufficient, and the complementary textual semantics may also be less informative, making accurate cross-view correspondence more challenging and thus resulting in failures. To mitigate this, a future direction is to explore relational and temporal cues in videos to improve the discrimination power of the model.



(a) Scenario of severe occlusion (b) Scenario of similar distractors (c) Scenario of small target

Figure 7: Representative failure cases of the proposed TeSCo under severe occlusion (see image (a)), similar distractors (see image (b)), and small target objects (see image (c)). The prediction results of TeSCo and groundtruth masks are highlighted in pink and green, respectively.

Discussion on More Implementation Details. Here we provide more implementation details of our model, including (i) DAM checkpoint: the DAM checkpoint is "nvidia/DAM-3B (the image variant)"; (ii) Text prompt: the DAM text prompt is "use one to two sentence descriptions to describe what the object is (from appearance) from the given masked image region, including category, color, texture, and material if available"; (iii) Caption preprocessing: we do not perform preprocessing on the caption; (iv) RoBERTa pooling strategy: we apply the CLS-style pooling strategy; (v) Frozen and training components: during training, the DAM model and text encoder are frozen, and other components are trainable; (vi) Memory bank size: the memory bank size is 7; (vii) Loss weights: Following SAM 2, we use a focal mask loss (weight 20) that supervises pixel-wise mask classification, Dice loss (weight 1) that encourages mask overlap, IoU loss (weight 1) that improves mask localization quality, and occlusion binary cross-entropy loss (weight 1) that supervises occlusion prediction; (viii) Runtime computation: text generation is included in the runtime computation.

Discussion on Uncertainty Estimates. We report uncertainty estimates of our method on Ego4D. Specifically, our method achieves 60.32±0.19 mIoU in Ego→Exo and 70.34±0.11 in Exo→Ego (mean±standard deviation over three runs), indicating stable performance across different runs.

Discussion on Consistency of Textual Semantics. Our method is motivated by the observation that textual semantics provide complementary information to visual features, thereby improving target localization under cross-view appearance variations. Quantitatively, the performance improvements of TeSCo over the baseline, as reported in Tab. 1, support the effectiveness of incorporating textual semantics. Qualitatively, the attention map visualizations in Fig. 4 show that textual semantics help the model attend more accurately to the target despite substantial appearance differences across views, suggesting textual semantics provide relatively consistent guidance across views.

Impact of Different Textual Attributes. To study the impact of textual attributes in TeSCo, we conduct a systematic textual attribute ablation by progressively retaining more information from the descriptions: (i) Category, (ii) Category and Color, and (iii) Complete Description (*i.e.*, all generated semantic attributes). Here, we select category and color because they are two of the most frequently occurring semantic attributes in the generated descriptions. The results are shown in Tab. 15. We can see that category information already provides substantial semantic cues for cross-view correspondence. Incorporating color further improves performance by enhancing object discrimination, while using the complete description achieves the best performance, demonstrating that richer semantic attributes provide complementary information beyond category and color.

Table 15: Results of integrating our textual modules into LM-EEC.

	Method	IoU
❶	Baseline	52.70
❷	TeSCo with Category	57.82
❸	TeSCo with Category and Color	59.65
❹	TeSCo with Complete Description	60.40

More Discussion with ObjectRelator [8]. Our TeSCo share a similar spirit with ObjectRelator [8] in exploring textual cues for cross-view segmentation. Despite this, TeSCo substantially differs from ObjectRelator, as discussed earlier in the paper. In experiments, compared to ObjectRelator, TeSCo shows more superior performance. Specifically on Ego4D, as shown in Tab. 1, TeSCo with mIoU of 65.35% surpasses ObjectRelator with 37.80% by 27.55%. On HANDAL-X, TeSCo also significantly outperforms ObjectRelator in Tab. 2. These results show the advantages of our TeSCo in applying textual semantics for robust segmentation.

Effect of Text Refinement. Fig. 8 visualizes the text-visual attention responses before and after ITFR refinement. Before refinement, the responses are scattered across the search view, with strong activations outside the target region. After refinement, the responses become more concentrated on the target object, indicating that ITFR effectively incorporates search-view visual context to enhance text-visual alignment.

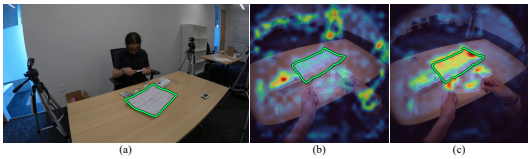


Figure 8: (a) Query view with the target outlined in green. (b) Text-visual attention before ITFR. (c) Text-visual attention after ITFR.

Discussion on (a) why the baseline model is strong and (b) why TeSCo improves substantially. (a) The baseline is built upon the SAM 2, a general and powerful segmentation model. Since cross-view correspondence is essentially a segmentation task, it naturally benefits from the strong visual matching capability and temporal modeling ability of SAM2 for segmentation. As a result, the baseline, to some degree, can effectively establish robust visual correspondences across frames and viewpoints, making

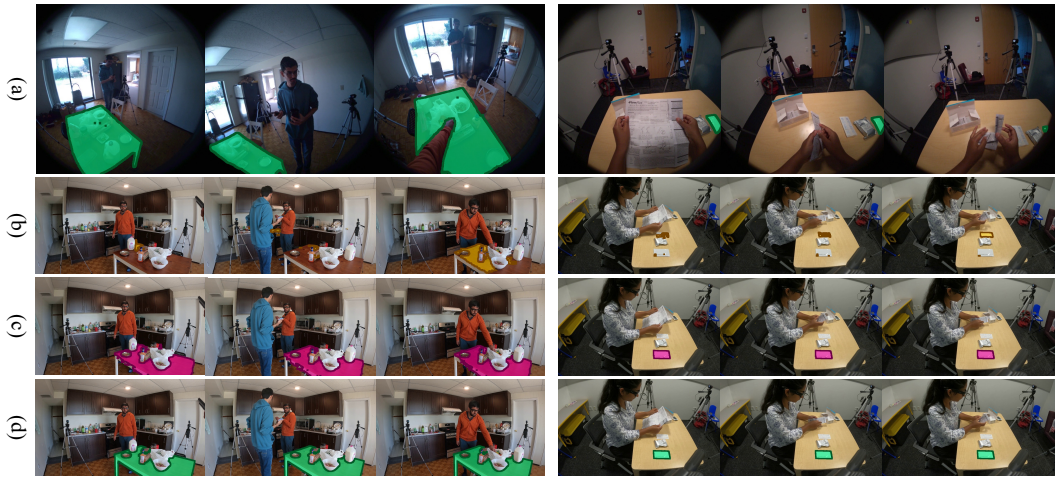


Figure 9: Qualitative results in Ego→Exo. Image (a): query-view (Ego-centric); image (b): results of baseline; image (c): results of TeSCo; image (d): search-view (Exo-centric) and groundtruth mask.

it a strong model for CVOC. (b) Although the baseline is strong, visual matching alone may still become ambiguous under large viewpoint and appearance variations. In these challenging scenarios, textual semantics provide relatively more stable cues that are less dependent on appearance and help preserve object identity across views. Building upon the strong baseline, TeSCo integrates the textual semantics into, rather than replacing, visual features, leading to stronger matching capability between different views and consistent improvements over the baseline model.

F Complexity Analysis

In this work, we train the proposed TeSCo using 4 Nvidia A6000 (48GB) GPUs for around 36 hours. The total number of parameters is around 600M. In inference, the GPU memory requirement for running our TeSCo is 3.3GB and the inference speed is around 3 frames per second (*fps*). A detailed inference-time breakdown include DAM with 272.7 ms on average, text encoder with 1 ms, ITFR with 0.5 ms, TCF with 0.6 ms, and SAM2 components (image encoder, memory attention, mask decoder, and memory encoder) with 24.4 ms. We can see that the runtime is dominated by DAM-based text generation, while the additional overhead of the text encoder, ITFR, and TCF is negligible. Compared with existing methods, including LM-EEC, V^2-SAM, CCMP, ObjectRelator, and O-MaMa, which run at 10, 2, 16, 1, and 4 FPS, respectively, TeSCo achieves a favorable balance between accuracy and inference speed.

G Qualitative Analysis

We further show the qualitative results of our method and comparison with the baseline model in Fig. 9, Fig. 10, and Fig. 11. Specifically, Fig. 9 and Fig. 10 respectively display the results in Ego→Exo and Exo→Ego on Ego-Exo4D, and Fig. 11 shows the results on HANDAL-X. As shown, the baseline model struggles in challenging scenarios. However, our model, owing to textual semantics, is able to accurately segment the targets in cross-view videos.

H Broader Impact

Our TeSCo aims to establish cross-view object correspondence between different views and has a wide range of applications, such as intelligent robotics, augmented reality (AR), and assistive technologies. By enhancing the ability of AI systems to reason about objects from multiple viewpoints, TeSCo is able to contribute to developing more robust human-AI interaction systems.

It is worth noting that our work may raise potential ethical risks, particularly related to privacy. Specifically, the ability to associate different views could be misused for unauthorized tracking or

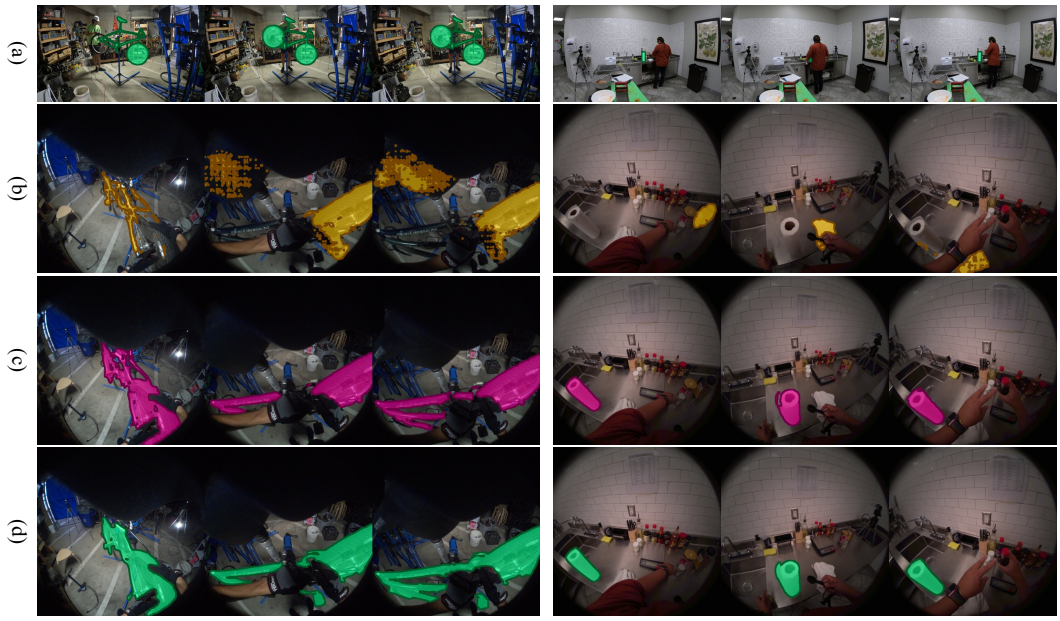


Figure 10: Qualitative results in Exo→Ego. Image (a): query-view (Exo-centric); image (b): results of baseline; image (c): results of TeSCo; image (d): search-view (Ego-centric) and groundtruth mask.

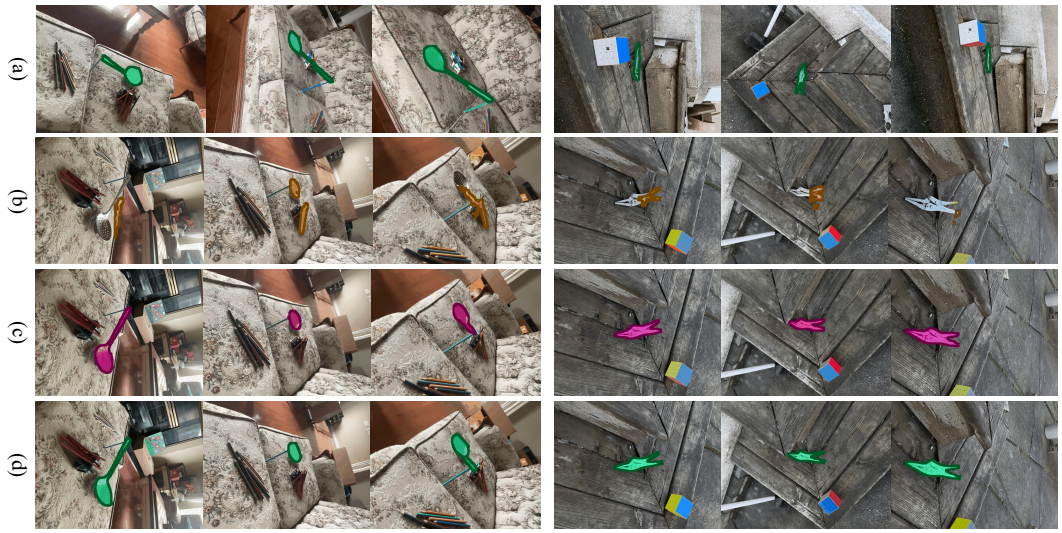


Figure 11: Qualitative results in HANDAL-X. Image (a): query-view; image (b): results of baseline; image (c): results of TeSCo; image (d): search-view and groundtruth mask.

monitoring of individuals beyond the intended scope of application. Therefore, our proposed method must be deployed in an ethical way.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes. The abstract and introduction state the paper's main claims and contributions, and these are supported by the method and experimental results reported in Sec. 3 and Sec. 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Yes. The paper explicitly discusses limitations in Sec. 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: NA. This work does not present formal theorems or theoretical proofs; the contribution is primarily methodological and empirical.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Yes. The paper reports method design, training protocol, datasets, evaluation metrics, and extensive comparisons/ablations, with additional implementation details and analyses in the supplementary material. We also state that code and model will be released upon acceptance to further support reproducibility.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: No. Code and model are not publicly released at submission time. We state that they will be released upon acceptance to support reproducibility.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes. The paper specifies datasets/splits, architecture components, optimizer, learning rates, training epochs, batch size, input resolution, and key module hyperparameters in Sec. 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: No. The paper reports point estimates of evaluation metrics but does not provide error bars, confidence intervals, or formal statistical significance tests.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Yes. The paper reports the compute worker type and training setup (4 NVIDIA A6000 GPUs, batch size, and input resolution) and provides sufficient implementation details for reproducing experiments under similar hardware conditions.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Yes. The research follows standard benchmark-based computer vision experimentation and does not involve practices that conflict with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Yes. We discuss the broader impact of our method in the appendix.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: NA. The work does not release high-risk generative models or scraped datasets that would require special misuse safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [No]

    Justification: No. Prior methods and datasets are cited, but explicit license and terms-of-use information for all used external assets are not comprehensively listed in the manuscript.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: No. The paper introduces a new model, but full asset documentation (e.g., release package documentation and terms) is not yet provided at submission time. We will provide documentation alongside the release in the camera-ready version.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: NA. The work does not involve crowdsourcing experiments or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: NA. The work does not involve crowdsourcing or human-subject studies requiring IRB or equivalent review.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: NA. The core methodological contribution does not introduce or rely on an original LLM component; the vision-language model is used only as an off-the-shelf tool to generate object descriptions.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.