End-to-end Open-vocabulary Video Visual Relationship Detection using Multi-modal Prompting

Yongqi Wang, Xinxiao Wu, *Member, IEEE*, Shuo Yang, Jiebo Luo *Fellow, IEEE*

***Abstract*—Open-vocabulary video visual relationship detection aims to expand video visual relationship detection beyond annotated categories by detecting unseen relationships between both seen and unseen objects in videos. Existing methods usually use trajectory detectors trained on closed datasets to detect object trajectories, and then feed these trajectories into large-scale pre-trained vision-language models to achieve open-vocabulary classification. Such heavy dependence on the pre-trained trajectory detectors limits their ability to generalize to novel object categories, leading to performance degradation. To address this challenge, we propose to unify object trajectory detection and relationship classification into an end-to-end open-vocabulary framework. Under this framework, we propose a relationship-aware open-vocabulary trajectory detector. It primarily consists of a query-based Transformer decoder, where the visual encoder of CLIP is distilled for frame-wise open-vocabulary object detection, and a trajectory associator. To exploit relationship context during trajectory detection, a relationship query is embedded into the Transformer decoder, and accordingly, an auxiliary relationship loss is designed to enable the decoder to perceive the relationships between objects explicitly. Moreover, we propose an open-vocabulary relationship classifier that leverages the rich semantic knowledge of CLIP to discover novel relationships. To adapt CLIP well to relationship classification, we design a multi-modal prompting method that employs spatio-temporal visual prompting for visual representation and vision-guided language prompting for language input. Extensive experiments on two public datasets, VidVRD and VidOR, demonstrate the effectiveness of our framework. Our framework is also applied to a more difficult cross-dataset scenario to further demonstrate its generalization ability. The code for this paper is available at https://github.com/wangyongqi558/EOV-MMP-VidVRD.**

***Index Terms*—Open-vocabulary video visual relationship detection; End-to-end framework; Multi-modal prompting; CLIP**

I. Introduction

VIDEO Visual Relationship Detection (VidVRD) aims to detect objects and their relationships in videos, typically represented as triplets in the format of $\langle subject, relationship, object \rangle$ [1]. Open-vocabulary Video Visual Relationship Detection (Open-VidVRD) expands VidVRD task by training on base categories of objects and relationships, and testing on both base and novel categories [2], which has wide applications in real-world scenarios.

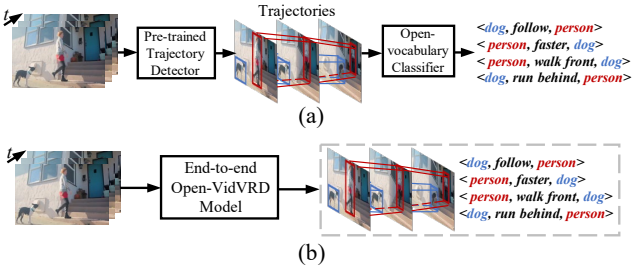


Fig. 1. (a) Existing Open-VidVRD methods rely on trajectory detectors trained on closed datasets. (b) The proposed end-to-end model performs Open-VidVRD directly on the original videos.

Recent significant progress has been made on open-vocabulary tasks [3]–[5] by integrating charming large-scale pre-trained vision-language models [6]–[10]. By learning joint vision-language embeddings, these pre-trained models can exploit extensive semantic knowledge of objects, scenes, actions, and interactions [11]–[16]. Existing Open-VidVRD methods [2], [17] typically employ prompt learning in pre-trained models to facilitate open-vocabulary classification of objects and relationships. These methods firstly use the trajectory detectors pre-trained on closed datasets to detect object trajectories from videos, and then feed the trajectories into pre-trained models like [8] for open-vocabulary classification of objects and relationships, as illustrated in Figure 1(a). Such heavy reliance on closed-set trajectory detectors limits their generalization capabilities to unseen object categories. Additionally, the domain gap between the training data of trajectory detectors and that of the Open-VidVRD task limits their adaptability to base categories. As a result, the detected object trajectories are suboptimal, hindering the subsequent relationship classification.

To address this challenge, we propose a novel end-to-end framework for Open-VidVRD, as illustrated in Figure 1(b). It jointly models object trajectory detection and relationship classification into a unified framework. Under this framework, we propose two key components: a relationship-aware open-vocabulary trajectory detector and an open-vocabulary rela-

Yongqi Wang and Xinxiao Wu are with the Beijing Key Laboratory of Intelligent Information Technology, School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China (e-mail: 3120230916@bit.edu.cn, wuxinxiao@bit.edu.cn).

Xinxiao Wu is also with the Guangdong Provincial Laboratory of Machine Perception and Intelligent Computing, Shenzhen MSU-BIT University, Shenzhen 518172, China.

Shuo Yang is with the Guangdong Provincial Laboratory of Machine Perception and Intelligent Computing, Shenzhen MSU-BIT University, Shenzhen 518172, China (e-mail: yangshuo@smbu.edu.cn).

Jiebo Luo is with the Department of Computer Science, University of Rochester, Rochester, NY 14627 USA (e-mail: jluo@cs.rochester.edu).

tionship classifier. The trajectory detector primarily consists of a query-based Transformer decoder in which the visual encoder of CLIP is distilled for frame-wise open-vocabulary object detection, and a trajectory associator for generating trajectories. The open-vocabulary relationship classifier leverages the rich semantic knowledge of CLIP to predict relationships between the generated object trajectories. By jointly training the trajectory detector and the relationship classifier, our framework does not suffer from the domain gap problem faced by existing methods that rely on pre-trained trajectory detectors. Moreover, by distilling the visual encoder of CLIP, our framework enhances the generalization to novel object categories thanks to CLIP's powerful representation capabilities.

To exploit the relationship context during trajectory detection, we propose to embed a relationship query into the query-based Transformer decoder, and design an auxiliary relationship loss accordingly to explicitly perceive the relationships between objects when decoding. By incorporating relationship context into trajectory detection, our framework enables mutual interactions between trajectory detection and relationship classification. This mutual interaction fosters a close coupling that facilitates the joint optimization of both processes within the end-to-end framework, ensuring that object trajectories and relationships are simultaneously refined and accurately detected.

To effectively leverage the knowledge of CLIP into the video domain during relationship classification, we propose a multi-modal prompting method that prompts CLIP on both visual and language sides. Specifically, we design spatio-temporal visual prompting to imbue CLIP with the capabilities of spatial and temporal modeling, effectively enhancing the image encoder of CLIP. Moreover, we design vision-guided language prompting to exploit CLIP's comprehensive semantic knowledge for discovering novel relationships in videos.

Extensive experiments on two public datasets, VidVRD [1] and VidOR [18], show that our end-to-end framework outperforms existing state-of-the-art methods, achieving 2.89% mAP gains on novel relationship categories on the VidVRD dataset. To further demonstrate the generalization ability of our method, our framework is also applied to a more difficult evaluation setting where the base categories of VidOR are used for training and unseen categories from VidVRD are used for testing. Under this setting, our framework achieves 9.77% mAP improvement on trajectory detection and 5.45% mAP improvement on relationship classification.

In summary, our main contributions are as follows:

1) We propose an end-to-end Open-VidVRD framework, which unifies trajectory detection and relationship classification, thus eliminating the need for pre-trained trajectory detectors and improving the generation to unseen categories.
2) We propose a relationship-aware open-vocabulary trajectory detector, which distills significant knowledge from CLIP and meanwhile perceives the relationship context via a dedicated relationship query and an auxiliary relationship loss.
3) We also propose an open-vocabulary relationship classifier with a multi-modal prompting method that prompts CLIP on both the visual and language sides to enhance the generalization to novel relationship categories.

A preliminary version of this paper, named OV-MMP [17], published in AAAI 2024. The differences between this paper and the previous version are summarized as follows: (1) This paper integrates the open-vocabulary relationship classifier with the multi-modal prompting method into a novel end-to-end framework, eliminating the reliance on pre-trained trajectory detector used in OV-MMP, enabling the joint optimization of trajectory detection and relationship classification. (2) This paper proposes a relationship-aware open-vocabulary trajectory detector that distills significant knowledge from CLIP visual encoder into the query-based Transformer decoder while explicitly perceiving the relationship context by designing a relationship query and an auxiliary relationship loss. (3) This paper further validates the generalization capability of our end-to-end framework by designing extensive experiments, including a new setting in which we train the model on the base categories of the VidOR dataset and test it on categories that are unseen during training from the VidVRD dataset.

II. Related Work

A. Video Visual Relationship Detection

Video Visual Relationship Detection (VidVRD) focuses on detecting interactions between objects over time, necessitating a comprehensive understanding of both spatial distribution and temporal dynamics of objects within videos [1]. Numerous studies have explored various VidVRD methods, which can be broadly categorized into spatio-temporal modeling, relationship refinement, video relationship debiasing, and end-to-end video relationship detection.

Spatio-temporal modeling methods design various architectures to learn dynamic interactions between objects across both spatial and temporal dimensions. Qian et al. [19], Tsai et al. [20], and Liu et al. [21] represent videos as fully connected spatio-temporal graphs and adopt graph convolution networks to reason about the relationships between objects. Cong et al. [22] use a spatial Transformer encoder to extract spatial context and intra-frame relationships, and a temporal decoder to understand inter-frame dynamic relationships.

Relationship refinement methods aim to learn fine-grained relationship representation between objects. Shang et al. [23] propose an iterative inference module that iteratively refines one component of a relationship triplet by using the prediction results of the other two components. Chen et al. [24] decouple complex relationships across multiple video frames into fine-grained relationships on single frames to capture frame-wise subtle interactions between objects.

Video relationship debiasing methods aim to address the long-tail distribution problem in video relationship datasets. Xu et al. [25] apply meta-learning to train an unbiased VidVRD model. They divide the training set into a support set and multiple query sets with different data distributions, where the support set is used to train the model, the query sets are used to optimize the model. Dong et al. [26] divide long-tail datasets into balanced sub-datasets, and individually train a relationship classifier for each subset. Then, they jointly

optimize the classifiers on the full training set and distill the unbiased knowledge in each classifier into a comprehensive classifier. Lin et al. [27] design an asymmetrical re-weighting loss function that adjusts the weights for each relationship category by using the effective number of samples proposed in [28].

End-to-end video relationship detection methods [29], [30] have been proposed in recent years. They jointly optimize both the trajectory detector and relationship classifier to improve the consistency of the object and relationship context.

All the above-mentioned methods are designed for closed settings where the training and test data share the same object and relationship categories, thus limiting their ability to generalize to unseen object and relationship categories. Consequently, they struggle to effectively adapt to the diverse and dynamic scenarios encountered in real-world videos.

B. Open-vocabulary Visual Relationship Detection

With the advancement of vision-language pre-training techniques, open-vocabulary visual tasks, such as object detection [31], spatio-temporal action localization [32], and video-text matching [33], have gained widespread attention for their ability to generalize beyond pre-defined categories.

The task of open-vocabulary visual relationship detection [2], [34] has been proposed recently, which focuses on detecting visual relationship instances involving objects and relationships that are unseen in the training data. The first study [34] on this task conducts contrastive learning on massive amounts of data to align the visual and textual representations of both objects and relationships, and identifies novel categories through similarity matching between visual content and textual descriptions. Further advancing this approach, Yuan et al. [35] propose to enhance relational language-image pre-training by accelerating convergence through early cross-modal fusion and improving scalability with pseudo-labeled relational annotations.

Due to the high computational demands of contrastive learning with large-scale data, many researchers resort to using rich semantic knowledge in existing pre-trained vision-language models [8], [9], [36] to recognize novel categories. Li et al. [37] use BLIP [9] to generate relationship triplets by feeding images and text prompts, and then replace synonyms in the generated triplets with the target categories through text similarity matching. Li et al. [38] enhance CLIP [8] to discover novel relationships by generating fine-grained visual and textual content. Specifically, they use object detection results to decompose the visual content into subject-related, object-related, and spatial-related fine-grained components, and adopt large language models (LLMs) to generate class-specific descriptive prompts for each component. Yu et al. [39] also use CLIP for open-vocabulary relationship classification, and propose a prompting method that concatenates learnable vectors to both textual and visual input to learn task-related knowledge. Moreover, they take the visual features and learnable text prompts into BERT [40] to generate comprehensive and fine-grained object relationships for expanding the training data. Zhao et al. [41] unify inconsistent label spaces across multiple datasets by leveraging the aligned vision-text semantic space in CLIP. Zhu et al. [42] employ a mask-based approach to unify multiple relationship understanding tasks, using CLIP text prompts to guide visual relationship segmentation and a query-based Transformer to generate relational triplets.

The above-mentioned methods are typically designed for images and can not be directly applied to video domains. In recent years, Gao et al. [2] pioneer open-vocabulary video visual relationship detection (Open-VidVRD) by using the pre-trained video-text model ALPro [36] for similarity matching between visual and linguistic modality features. However, this method relies heavily on a trajectory detector pre-trained on closed datasets, thus limiting the ability to generalize to unseen object categories.

In this paper, we propose a novel end-to-end Open-VidVRD framework that jointly models object trajectory detection and relationship classification, eliminating the reliance on pre-trained trajectory detectors. Moreover, by leveraging the knowledge in CLIP through a novel multi-modal prompting method, our framework adapts well to diverse real-world scenarios.

C. Prompting CLIP

Vision-language pre-trained models [8], [43]–[45] have demonstrated significant progress in many downstream vision-language tasks. As one of the most successful vision-language pre-trained models, CLIP [8], is extensively pre-trained using 400 million image-text pairs from the Internet, resulting in a vision-language embedding space with comprehensive semantic knowledge.

Various text prompting methods have emerged to effectively transfer knowledge from CLIP to downstream tasks. Zhou et al. [46] convert handcraft text prompts into learnable vectors to learn task-related knowledge. Zhou et al. [47] further propose conditional text prompts, which integrate learnable vectors with visual features, to learn the image-specific knowledge. Sun et al. [48] adapt CLIP to a multi-label image recognition task by learning pairs of positive and negative text prompts to ensure independent binary classification for each category.

Meanwhile, many visual prompting methods for CLIP have been widely explored. Jia et al. [49] integrate the input images with learnable vectors to learn task-related visual cues. Wang et al. [50] and Xu et al. [51] incorporate learnable tokens into the visual encoder to refine the visual features, making them more suitable for downstream tasks.

To fully exploit the multi-modal co-optimization potential of CLIP, multi-modal prompting methods [52]–[54] have been proposed. These methods introduce learnable vectors to both visual and textual modalities, and couple them to facilitate joint optimization. All these methods primarily focus on image domain tasks. In contrast, our multi-modal prompting method is specifically designed for the more challenging Open-VidVRD task.

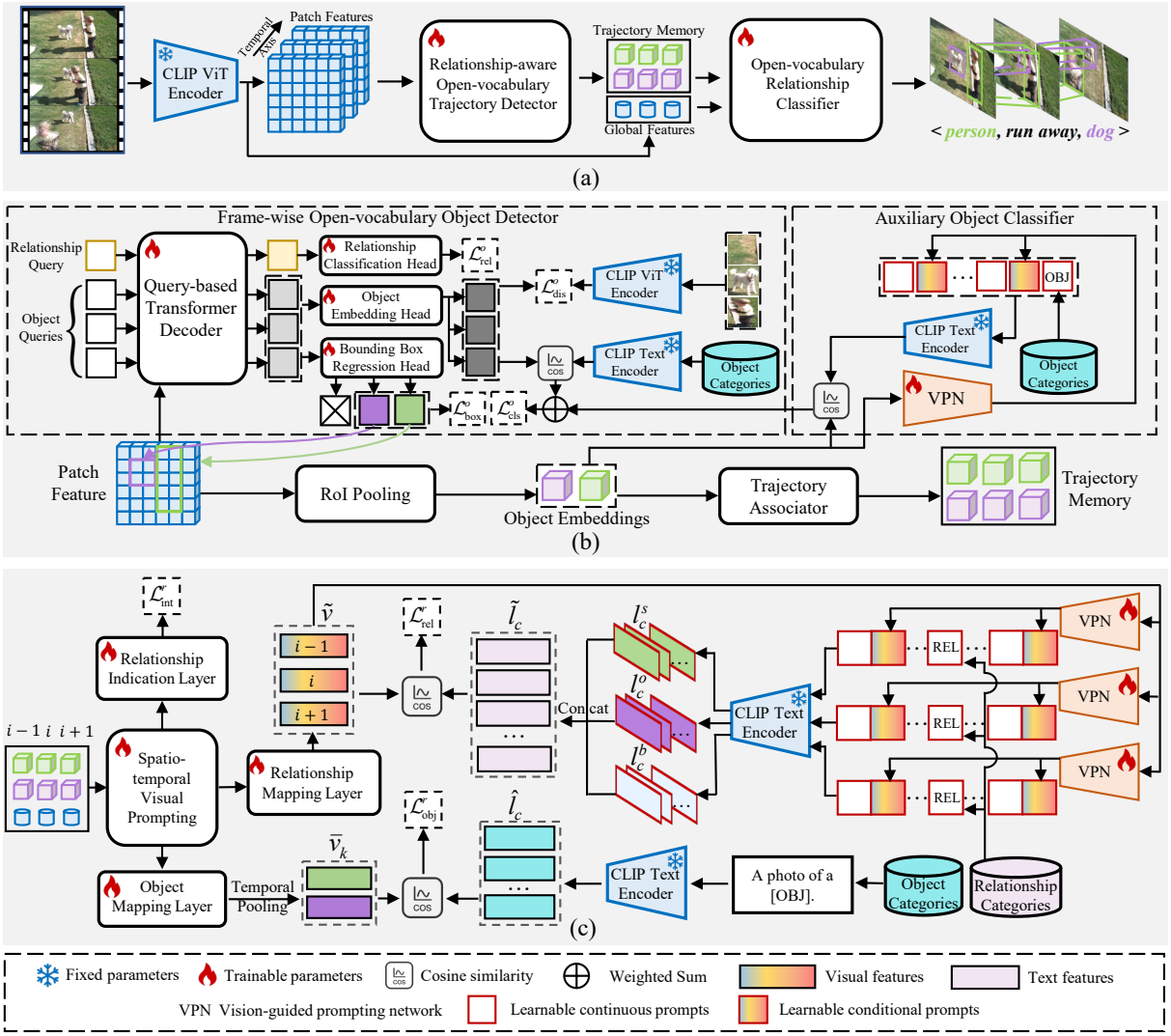


Fig. 2. (a) The proposed end-to-end framework, where the object trajectories and their categories are predicted by the relationship-aware open-vocabulary trajectory detector, and the relationship categories are predicted by the open-vocabulary relationship classifier. (b) The relationship-aware open-vocabulary trajectory detector. (c) The open-vocabulary relationship classifier.

III. Our Framework

A. Overview

Video Visual Relationship Detection (VidVRD) aims to detect instances of visual relationships within a video $V = \{f_t\}_{t=1}^{N_v}$, where f_t represents the frame at time t, and N_v is the number of frames in V. Each visual relationship instance is represented by a tuple $(c^s, c^r, c^o, T^s, T^o)$, where c^s, c^r, and c^o denote the categories of the subject, relationship, and object, respectively. T^k, with $k \in \{s, o\}$, represents the trajectory of subjects or objects, comprising a sequence of bounding boxes $(b^k_{t_s}, \ldots, b^k_{t_j}, \ldots, b^k_{t_e})$, where $b^k_{t_j}$ denotes the corresponding bounding box at time t_j, with t_s and t_e being the start time and end time of the trajectory, respectively. In Open-VidVRD, the categories of objects and relationships are divided into base and novel splits: base objects ($\mathcal{C}^O_b$), novel objects ($\mathcal{C}^O_n$), base relationships ($\mathcal{C}^R_b$), and novel relationships ($\mathcal{C}^R_n$). Only base categories are used during the training phase, and all categories are used during the test phase.

We propose an end-to-end Open-VidVRD framework that directly detects relationships between objects from input raw videos. It comprises two main components: a relationship-aware open-vocabulary trajectory detector (Sec. III-B) and an open-vocabulary relationship classifier (Sec. III-C). An overview of our framework is illustrated in Figure 2 (a).

B. Relationship-aware Open-vocabulary Trajectory Detection

For each input video V, we first use a ViT-based visual encoder [55] of CLIP to extract visual features for each frame, represented by

$$(F^g_t, F^p_t) = \mathcal{V}(V), \tag{1}$$

where $\mathcal{V}(\cdot)$ denotes the visual encoder of CLIP, F^g_t and F^p_t denote the global feature and the patch feature of the t-th frame, respectively. We then feed the patch features into a relationship-aware open-vocabulary trajectory detector to obtain object trajectories, represented by

$$(T_i, c_i, E_i) = \Phi(F^p_1, F^p_2, \ldots, F^p_{N_v}), \tag{2}$$

where $\Phi(\cdot)$ denotes the trajectory detector. T_i is the i-th trajectory, where $i \in \{1, ..., N_t\}$ and N_t is the trajectory number in the video. c_i denotes the object category of the i-th trajectory. E_i represents the visual feature of the i-th trajectory.

The trajectory detection process begins with a query-based Transformer decoder that distills the visual encoder of CLIP to perform frame-wise open-vocabulary object detection (Sec. III-B1). Then the frame-wise object detection results are enhanced by an auxiliary object classifier (Sec. III-B2) that leverages CLIP to discover novel object categories. Finally, a trajectory associator (Sec. III-B3) connects the frame-wise detection results to generate coherent object trajectories throughout the video. A trajectory memory is built to store the trajectory detection results, which are then used for subsequent open-vocabulary relationship classification.

We further propose a relationship query as input for the query-based Transformer decoder and design a corresponding auxiliary relationship loss (Sec. III-B4) to make the decoder explicitly perceive the relationship context. Figure 2 (b) illustrates the details of the proposed relationship-aware open-vocabulary trajectory detector.

1) Frame-wise Open-vocabulary Object Detection: We feed the patch feature together with object queries and a relationship query into the query-based Transformer decoder [56] to obtain the query results. The object query results are then processed through prediction heads to generate frame-wise object detection results, and the relationship query result is used to calculate the auxiliary relationship loss.

Object Query. We define a set of N_q learnable object queries, denoted by $Q = \{q^1, q^2, \ldots, q^{N_q}\}$, to process image context and output predictions in parallel. The object queries are shared across all video frames.

Relationship Query. We propose a relationship query, denoted as R, which interacts with the patch feature F^p to perceive the relationship context. The relationship query is shared across all video frames.

Query-based Transformer Decoder. The query-based Transformer decoder has N_l layers, and each layer is composed of alternating self-attention and cross-attention modules. In the l-th layer, the object queries Q^l and relationship query R^l first interact with each other through the self-attention module. The resulting outputs, $\tilde{Q}^l$ and $\tilde{R}^l$, are used to extract features from the patch feature F^p through the cross-attention module. This process is formulated as

$$\begin{aligned} (\tilde{Q}^l, \tilde{R}^l) &= \text{SelfAttn}_l\left([Q^l; R^l]\right), \\ (\hat{Q}^l, \hat{R}^l) &= \text{CrossAttn}_l\left([\tilde{Q}^l; \tilde{R}^l], F^p\right), \end{aligned} \tag{3}$$

where $\hat{Q}^l$ and $\hat{R}^l$ represent the output of the l-th decoder layer, and serve as the input Q^{l+1} and R^{l+1} for the $(l+1)$-th layer. The final output of the query-based Transformer decoder is represented by $\hat{Q}^{N_l} = \{\hat{q}^1, \hat{q}^2, \ldots, \hat{q}^{N_q}\}$ and $\hat{R}^{N_l}$.

Prediction Heads. For each object query result $\hat{q} \in \hat{Q}^{N_l}$, its corresponding object bounding box is predicted as

$$b = \mathcal{M}_{box}(\hat{q}), \tag{4}$$

where b is the predicted bounding box, $\mathcal{M}_{box}(\cdot)$ is the bounding box regression head, consisting of two linear layers. The classification score of object category $c \in C$ is represented as

$$p(c) = \frac{\exp(\cos(\mathcal{M}_{emb}(\hat{q}), e^c_{\text{txt}})/\tau)}{\sum_{c' \in C} \exp(\cos(\mathcal{M}_{emb}(\hat{q}), e^{c'}_{\text{txt}})/\tau)}, \tag{5}$$

where $C = \mathcal{C}^O_b$ during the training phase and $C = \mathcal{C}^O_b \cup \mathcal{C}^O_n$ during the test phase, τ is a temperature parameter, $\mathcal{M}_{emb}(\cdot)$ is the object embedding head consisting of two linear layers, $\cos(\cdot, \cdot)$ is the cosine similarity matching function, and e^c_{txt} denotes the text feature of object category c extracted by CLIP. We retain only the bounding boxes where the maximum value of p exceeds a threshold ϵ (an ablation is presented in Table VIII), and discard the other bounding boxes. For the retained bounding boxes, we create a mask M and apply Region of Interest (RoI) Pooling on the original patch feature F^p to extract a fixed-size feature vector as the object embedding $\mathcal{E}$, formulated as

$$\mathcal{E} = \mathcal{P}(F^p, M), \tag{6}$$

where $\mathcal{P}(\cdot)$ denotes the RoI Pooling function.

For the relationship query result $\hat{R}^{N_l}$, we predict the score of relationship category $r \in \mathcal{C}^R_b$ by

$$p(r) = \mathcal{M}_{rel}(\hat{R}^{N_l}), \tag{7}$$

where $\mathcal{M}_{rel}(\cdot)$ is the relationship classification head, consisting of two linear layers.

2) Auxiliary Object Classification: To further improve the classification performance of novel object categories, we design an auxiliary object classifier that uses CLIP to classify the object embeddings obtained in Eq. 6 by calculating their similarity with text features. To fully leverage the rich semantic knowledge of CLIP, we propose a vision-guided prompting method. Specifically, we feed the object embeddings into a vision-guided prompting network (VPN) to generate learnable conditional language prompts. These generated prompts are then combined with learnable continuous language prompts as input for the text encoder of CLIP.

Learnable Continuous Language Prompts. For each object category [OBJ], N_ς-token language prompts are initialized by $\varsigma = [\varsigma_1, \varsigma_2, \cdots, \varsigma_{N_\varsigma}]$, where [OBJ] $\in \mathcal{C}^O_b$ when training and [OBJ] $\in \mathcal{C}^O_n \cup \mathcal{C}^O_b$ when testing,

Learnable Conditional Language Prompts. For each object category [OBJ], N_ζ-token learnable conditional language prompts are learned by taking into account the corresponding visual feature, represented as

$$\boldsymbol{\zeta} = [\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2, \cdots, \boldsymbol{\zeta}_{N_\zeta}] = \varphi(\mathcal{E}), \tag{8}$$

where $\varphi(\cdot)$ denotes the vision-guided prompting network, consisting of two linear layers. $\mathcal{E}$ is the object embedding.

Learnable Vision-guided Language Prompts. We concatenate the tokens of learnable continuous language prompts and tokens of learnable conditional language prompts interlaced, and then insert the [OBJ] token into the end of the token sequence, to obtain the final language prompts $\boldsymbol{\jmath}_{\text{OBJ}} = \left[\varsigma_1, \boldsymbol{\zeta}_1, \varsigma_2, \boldsymbol{\zeta}_2, \cdots, \varsigma_{N_\varsigma}, \boldsymbol{\zeta}_{N_\zeta}, \text{OBJ}\right]$. The text feature of the object category c is denoted as

$$\boldsymbol{\jmath}_c = \mathcal{T}(\boldsymbol{\jmath}_c), \tag{9}$$

where $\mathcal{T}(\cdot)$ is the text encoder of CLIP.

The auxiliary classification score of object category $c \in C$ is represented as

$$\tilde{p}(c) = \frac{\exp(\cos(\mathcal{E}, \boldsymbol{j}_c)/\tau)}{\sum_{c' \in C} \exp(\cos(\mathcal{E}, \boldsymbol{j}_{c'})/\tau)}. \tag{10}$$

The final frame-wise object classification score is represented as

$$\hat{p}(c) = \begin{cases} (1-\alpha)p(c) + \alpha\tilde{p}(c) \text{ if } c \in \mathcal{C}_b^O, \\ (1-\beta)p(c) + \beta\tilde{p}(c) \text{ if } c \in \mathcal{C}_n^O, \end{cases} \tag{11}$$

where $\alpha, \beta \in [0, 1]$ are weighting factors for the base and novel object categories, respectively.

3) Trajectory Association: We employ a feature-based association algorithm [57] that links frame-wise detection results that are spatially close and visually similar to generate object trajectories $(T_1, T_2, ..., T_{N_t})$, where N_t is the number of trajectories in the video.

For the i-th trajectory, its object classification score P_i is calculated by averaging the final frame-wise object classification scores, formulated as

$$\hat{P}_i = \frac{1}{t_\mathrm{e} - t_\mathrm{s} + 1} \sum_{t=t_\mathrm{s}}^{t_\mathrm{e}} \hat{p}_t^i, \tag{12}$$

where $\hat{p}_t^i$ represents the final classification score of the i-th trajectory at the t-th frame, t_s and t_e are the start and end time of the trajectory, respectively. The predicted object category of the i-th trajectory is given by

$$c_i = \arg\max_c \hat{P}_i(c). \tag{13}$$

The visual feature of the i-th trajectory is represented as

$$E_i = \{\mathcal{E}_t^i\}_{t=t_s}^{t_e}, \tag{14}$$

where $\mathcal{E}_t^i$ is the object embedding of the i-th trajectory at the t-th frame. (T_i, c_i, E_i) are stored in the trajectory memory for subsequent relationship classification.

4) Training Loss: We use a focal loss [58] for object classification, and an L1 loss and a GIoU loss [59] for box regression.

Distillation Loss. We design a distillation loss to distill the knowledge from CLIP's visual encoder for frame-wise open-vocabulary object detection, formulated as

$$\mathcal{L}_{dis}^o = \frac{1}{N_b} \cdot \sum_{n=1}^{N_b} \|\mathbf{e}^n - \mathbf{z}^n\|_1, \tag{15}$$

where $\mathcal{L}_{dis}^o$ represents the distillation loss, N_b denotes the total number of the retained bounding boxes in the video, $\mathbf{e}^n$ is the visual feature of the n-th bounding box extracted by CLIP, and $\mathbf{z}^n$ is the corresponding object query result encoded by the object embedding head, *i.e.*, $\mathcal{M}_{emb}(\hat{q})$ in Eq. 5.

Auxiliary Relationship Loss. We propose an auxiliary relationship loss, calculated using Binary Cross-Entropy (BCE), to enable the decoder to explicitly perceive the relationships between objects. The object relationships are categorized into dynamic and static types. Dynamic relationships change significantly over time and require multiple video frames to be assessed together to make a judgment. For example, a dynamic relationship such as "run past" involves motion that unfolds over multiple frames, which means that the relationship between objects can only be understood by analyzing how the scene evolves over time. In contrast, static relationships remain relatively constant and can be determined from a single video frame. For example, a static relationship such as "lie behind" describes a stable spatial configuration between objects that does not require temporal tracking. Once the relative position is established in a single frame, the relationship remains clear.

To emphasize the importance of static categories for understanding frame-wise relationships, we adjust the BCE loss with a predefined weight λ_s, ensuring that static relationships receive appropriate emphasis in the learning process. The auxiliary relationship loss $\mathcal{L}_{rel}^o$ is formulated as

$$\begin{aligned} \mathcal{L}_{rel}^o &= \mathcal{L}_d + \lambda_s \mathcal{L}_s, \\ \mathcal{L}_d &= \frac{1}{N_v} \cdot \sum_{t=1}^{N_v} \mathrm{BCE}(r_t^d, \hat{r}_t^d), \\ \mathcal{L}_s &= \frac{1}{N_v} \cdot \sum_{t=1}^{N_v} \mathrm{BCE}(r_t^s, \hat{r}_t^s), \end{aligned} \tag{16}$$

where r_t^d and r_t^s represent the predicted scores of dynamic and static relationship categories in the t-th video frame, respectively, and $\hat{r}_t^d$ and $\hat{r}_t^s$ represent the ground-truth labels of dynamic and static relationship categories, respectively. N_v is the number of frames.

The overall training loss of our relationship-aware open-vocabulary trajectory detector is given by

$$\mathcal{L}^o = \lambda_1 \mathcal{L}_{foc}^o + \lambda_2 \mathcal{L}_{l1}^o + \lambda_3 \mathcal{L}_{iou}^o + \lambda_4 \mathcal{L}_{dis}^o + \lambda_5 \mathcal{L}_{rel}^o, \tag{17}$$

where $\mathcal{L}_{foc}^o$, $\mathcal{L}_{l1}^o$, $\mathcal{L}_{iou}^o$, $\mathcal{L}_{dis}^o$, and $\mathcal{L}_{rel}^o$ represent the focal loss, L1 loss, GIoU loss, distillation loss and auxiliary relationship loss, respectively. Note that all the aforementioned losses are calculated using the frame-level object detection results.

C. Open-vocabulary Relationship Classification

We pair the detected object trajectories with temporal overlap in the trajectory memory and denote each trajectory pair as $(T^s, c^s, E^s, T^o, c^o, E^o)$, where T^s, c^s, and E^s represent the trajectory, object category, and visual feature of the subject in the trajectory pair, respectively, and T^o, c^o, and E^o represent the trajectory, object category, and visual feature of the object in the trajectory pair. We extract the visual features of the background of trajectory pair by aggregating the global feature of each frame, denoted as $E^h = \{F_t^g\}_{t=t_s}^{t_e}$, where F_t^g is the global visual feature (extracted in Eq. 1) of the t-th video frame, t_s and t_e are the start and end frames of the trajectory pair. Then we feed the visual features of the subject, object, and background into an open-vocabulary relationship classifier which uses CLIP to generate the relationship classification results by calculating the similarity between the visual features and text features, represented by

$$c^r = \Psi(E^s, E^o, E^h), \tag{18}$$

where c^r represents the predicted relationship category label, $\Psi(\cdot)$ represents the open-vocabulary relationship classifier.

To adapt CLIP well to relationship classification, we propose a multi-modal prompting method that applies prompt

learning to both visual and textual branches of CLIP. Specifically, we propose a spatio-temporal visual prompting method (Sec. III-C1) to capture dynamic contexts, and a vision-guided language prompting method (Sec. III-C2) to exploit CLIP's comprehensive semantic knowledge for discovering unseen relationship categories. Figure 2 (c) illustrates the details of the proposed open-vocabulary relationship classifier.

1) Spatio-temporal Visual Prompting: We use standard Transformer blocks to model the sptaio-temporal relationships between objects. To reduce the computational complexity, we decouple the spatio-temporal modeling into separate and successive modules, namely spatial modeling and temporal modeling.

Spatial Modeling. Spatial relationships between objects are typically defined by their positional orientations, such as being in front of or above each other. Therefore, spatial modeling requires combining three key elements: features of the subject region, features of the object region, and features representing the background (*i.e.* the whole image). This process involves modeling interactions between objects and their background to capture spatial context, thus enhancing object features.

Given the features of the trajectory pair, denoted by $E^k, k \in \{s, o, h\}$, we add two types of learnable embeddings: positional embedding ϱ^k related to the normalized bounding box, and role embedding ρ^k. These two types of embeddings are learned and shared across all video frames. The visual features are updated as follows:

$$(\dot{\boldsymbol{v}}^s, \dot{\boldsymbol{v}}^o, \dot{\boldsymbol{v}}^h) = \mathrm{STrans}(\boldsymbol{I}^s, \boldsymbol{I}^o, \boldsymbol{I}^h), \tag{19}$$

where $\boldsymbol{I}^k = E^k + \varrho^k + \rho^k, k \in \{s, o, h\}$, and STrans$(\cdot)$ denotes the spatial Transformer blocks.

Temporal Modeling. Temporal relationships of objects are time-dependent, such as moving toward or away, so the inputs for temporal modeling include visual features and temporal embeddings. For simplicity, the same temporal modeling is applied to different roles, *i.e.*, subject, object, and their background in this paper. Through the exploration of dynamic state transformations, the visual features are systematically updated.

Given the spatially encoded visual features $\dot{\boldsymbol{v}} = \{\dot{\boldsymbol{v}}_t^s, \dot{\boldsymbol{v}}_t^o, \dot{\boldsymbol{v}}_t^h\}_{t=t_s}^{t_e}$, for each role, we collect the corresponding features across all frames, denoted as $\dot{\boldsymbol{v}}^k = \{\dot{\boldsymbol{v}}_t^k\}_{t=t_s}^{t_e}$, where $k \in \{s, o, h\}$. We then add temporal embedding θ_t, which is related to frame t and shared across all roles. For each role, the visual features are updated by

$$\ddot{\boldsymbol{v}}^k = \{\ddot{\boldsymbol{v}}_t^k\}_{t=t_s}^{t_e} = \mathrm{TTrans}(\dot{\boldsymbol{I}}_{t_s}^k, \dot{\boldsymbol{I}}_{t_s+1}^k, \cdots, \dot{\boldsymbol{I}}_{t_e}^k), \tag{20}$$

where $\dot{\boldsymbol{I}}_t^k = \dot{\boldsymbol{v}}_t^k + \theta_t$, and TTrans$(\cdot)$ denotes the temporal Transformer blocks.

2) Vision-guided Language Prompting: Similar to Sec. III-B2, we construct vision-guided language prompts as

$$\boldsymbol{\ell}_{\mathrm{REL}}^k = \left[\varsigma_1^k, \boldsymbol{\zeta}_1^k, \varsigma_2^k, \boldsymbol{\zeta}_2^k, \cdots, \mathrm{REL}, \cdots, \varsigma_{N_\varsigma}^k, \boldsymbol{\zeta}_{N_\zeta}^k\right], \tag{21}$$

where $k \in \{s, o, h\}$ and [REL] $\in \mathcal{C}_b^R$ during training and [REL] $\in \mathcal{C}_n^R \cup \mathcal{C}_b^R$ during testing. For each visual region, the final text features of relationship category r are given by

$$\boldsymbol{l}_r^k = \mathcal{T}(\boldsymbol{\ell}_r^k), \tag{22}$$

where $\mathcal{T}(\cdot)$ is the text encoder of CLIP.

3) Training Loss: The training loss of the open-vocabulary relationship classifier consists of three parts: a relationship classification loss $\mathcal{L}_{rel}^r$, an object classification loss $\mathcal{L}_{obj}^r$, and an interaction loss $\mathcal{L}_{int}^r$, as shown in Figure 2 (c). The overall training loss is given by

$$\mathcal{L}^r = \mathcal{L}_{rel}^r + \gamma \mathcal{L}_{obj}^r + \delta \mathcal{L}_{int}^r. \tag{23}$$

Relationship Classification Loss. Given the visual features $\ddot{\boldsymbol{v}}^k$ and the text features $\boldsymbol{l}_r^k$, the prediction score of the relationship category r is calculated by

$$\hat{y}_r^{rel} = \sigma(\cos(\tilde{\boldsymbol{v}}, \tilde{\boldsymbol{l}}_{\boldsymbol{r}})), \tag{24}$$

where $\tilde{\boldsymbol{v}} = \psi([\ddot{\boldsymbol{v}}^s; \ddot{\boldsymbol{v}}^o; \ddot{\boldsymbol{v}}^h])$, $\psi(\cdot)$ denotes the relationship mapping layer in Figure 2 (c), $\tilde{\boldsymbol{l}}_{\boldsymbol{r}} = [\boldsymbol{l}_r^s; \boldsymbol{l}_r^o; \boldsymbol{l}_r^h]$, $\sigma(\cdot)$ is the sigmoid function, $\cos(\cdot, \cdot)$ is the cosine similarity. The relationship classification loss is formulated by using the BCE loss:

$$\mathcal{L}_{rel}^r = \frac{1}{|\mathcal{C}_b^R|} \cdot \sum\nolimits_{r \in \mathcal{C}_b^R} \mathrm{BCE}(\hat{y}_r^{rel}, y_r^{rel}), \tag{25}$$

where $y_r^{rel} = 1$ when r equals to the ground-truth relationship category, otherwise $y_r^{rel} = 0$.

Object Classification Loss. To avoid the visual feature drift caused by spatio-temporal visual prompting, we introduce an object classification loss to enforce the visual features after spatial modeling to have the same object distinguishing capability as the original CLIP. Specifically, after the spatial modeling, we collect the subject and object features from all frames and average them as $\bar{\boldsymbol{v}}^k = \mathrm{avg}(\{\phi(\dot{\boldsymbol{v}}_t^k)\}_{t=0}^T)$, $k \in \{s, o\}$ and $\phi(\cdot)$ denotes the object mapping layer, as shown in Figure 2 (c). Meanwhile, we extract the text features for all subject or object categories by feeding the handcrafted prompts (*i.e.*, "a photo of [OBJ]") into the text encoder of CLIP, where [OBJ] can be replaced with the names of subjects or objects. The similarity between the visual features and the text features of object category c is calculated by $\hat{y}_c^k = \cos(\bar{\boldsymbol{v}}^k, \hat{\boldsymbol{l}}_c), k \in \{s, o\}$. Finally, the object classification loss is computed over all object categories using the cross-entropy loss (CE):

$$\mathcal{L}_{obj}^r = \mathrm{CE}(\hat{\boldsymbol{y}}^s, y^s) + \mathrm{CE}(\hat{\boldsymbol{y}}^o, y^o), \tag{26}$$

where $\hat{\boldsymbol{y}}^s$ is the predicted subject similarity between visual features and text features of base object categories ($\mathcal{C}_b^O$), and $\hat{\boldsymbol{y}}^o$ is the corresponding predicted object similarity. y^s and y^o denote the ground-truth category labels of the subject and object, respectively.

Interaction Loss. There may be no annotated relationships between some subjects and objects, that is, there is no interaction. For each pair of subject and object, if there are any relationship categories between them in video frame t, we set the ground-truth interaction by $y_t^{int} = 1$, otherwise $y_t^{int} = 0$. To learn this weak interaction, we concatenate all the features in frame t and predict the interaction probability by $\hat{y}_t^{int} = \psi([\ddot{\boldsymbol{v}}_t^s; \ddot{\boldsymbol{v}}_t^o; \ddot{\boldsymbol{v}}_t^h])$, where $\psi(\cdot)$ denotes the relationship indication layer in Figure 2 (c). The interaction loss is then computed using the binary cross-entropy loss (BCE):

$$\mathcal{L}_{int} = \frac{1}{t_e - t_s} \cdot \sum\nolimits_{t=t_s}^{t_e} \mathrm{BCE}(\hat{y}_t^{int}, y_t^{int}), \tag{27}$$

where t_s and t_e represent the start and end time of the trajectory pair, respectively.

D. Training Strategy

We adopt a four-step scheme for training. **Step one:** We train the query-based Transformer decoder and prediction heads using video frames with frame-wise object and relationship annotations via the training loss $\mathcal{L}_{step_1} = \mathcal{L}^o$, as detailed in Sec. III-B4. **Step two:** We train the auxiliary object classifier using video frames with provided ground-truth bounding boxes via the training loss $\mathcal{L}_{step_2} = \mathcal{L}^o_{cls}$, as detailed in Sec. III-B4. **Step three:** We train the open-vocabulary relationship classifier using videos with provided ground-truth object trajectories via the training loss $\mathcal{L}_{step_3} = \mathcal{L}^r$, as detailed in Sec. III-C3. **Step four:** We jointly fine-tune the entire end-to-end framework via the overall training loss $\mathcal{L}_{step_4} = \mathcal{L}^o + \mathcal{L}^r$.

E. Computational Complexity Analysis

The computational complexity of our framework is determined by three main stages: (1) frame-wise object detection, (2) trajectory association, and (3) spatio-temporal visual prompting for paired object trajectories. Below, we analyze each component in detail.

Frame-Wise Object Detection Complexity. For each frame, objects are detected independently. Let N_v be the number of frames in a video and N_q be the number of object queries, which determines the maximum number of objects that the model can query in each frame, the computational complexity of this stage is $\mathcal{O}(N_v \cdot N_q)$. Since the input resolution is normalized to a fixed size of 336×336 by CLIP's ViT encoder, the complexity is independent of the original frame resolution.

Trajectory Association Complexity. After objects are detected in each frame, trajectories are constructed by associating objects across frames. Let N_o denote the number of objects detected per frame, which is typically much smaller than the number of object queries N_q. The trajectory association step involves pairwise comparisons of detected objects between consecutive frames, with a computational complexity of approximately $\mathcal{O}(N_v \cdot N_o^2)$.

Spatio-Temporal Visual Prompting Complexity. After trajectories are constructed, spatio-temporal visual prompting is performed on object pairs. Let N_t be the total number of trajectories in the video, and the maximum number of trajectory pairs is $N_t \times (N_t - 1)$. The computational complexity of this step is about $\mathcal{O}(N_v^2 \cdot N_t^2)$, as temporal prompting requires modeling features across different frames, resulting in quadratic complexity with the number of frames N_v.

Overall Complexity. Combining the above stages, the total computational complexity is $\mathcal{O}(N_v \cdot N_q + N_v \cdot N_o^2 + N_v^2 \cdot N_t^2)$.

Scalability. While frame-wise object detection scales linearly with N_v and N_q, trajectory association grows quadratically with N_o, and spatio-temporal visual prompting scales quadratically, especially with respect to N_v and N_t. This poses scalability challenges for longer videos and larger object sets, which is a common challenge faced by current Open-VidVRD methods [2], [17]. In our future work, incorporating linear or group attention mechanisms can reduce the complexity of temporal modeling and improve efficiency in handling long videos. Additionally, designing a selection mechanism to identify the most likely trajectory pairs for classification, rather than classifying relationships pairwise for all trajectories, can help improve the efficiency in scenarios with large object sets.

F. Discussion

In this paper, we use CLIP for the Open-VidVRD task. Video-text pre-trained models such as InternVideo [60] and VideoCLIP [61] can also be applied to our framework. Compared with them, CLIP trained with images and texts performs better in preserving critical details of object positions and appearances in video frames, enabling our framework to effectively capture subtle visual information for object trajectory detection, thereby facilitating relationship classification.

IV. Experiment

A. Datasets and Evaluation Metrics

1) Datasets: We evaluate our method on the **VidVRD** [1] and **VidOR** [18] datasets. The VidVRD dataset contains 1000 videos, with 800 videos for training and 200 for testing, covering 35 object categories and 132 predicate categories. The average video length in VidVRD is 9.7 seconds. The VidOR dataset contains 10000 videos, with 7000 videos for training, 835 for validation, and 2165 for testing, covering 80 object categories and 50 predicate categories. The videos in VidOR are much longer, with an average length of 34.6 seconds.

2) Evaluation Settings: For the open-vocabulary evaluation, the base and novel categories are selected based on frequency. Following RePro [2], we choose the common object and relationship categories as base categories and the rare ones as novel categories. Training is performed on the base categories and testing is performed under two settings: (1) **Novel**-split evaluation involves novel object categories for trajectory detection, and all object categories along with novel relationship categories for relationship classification. (2) **All**-split evaluation involves all object categories and all relationship categories, which is a standard evaluation. Note that the test is performed on both the VidVRD test set and the VidOR validation set (the annotations of the VidOR test set are not available).

3) Evaluation Tasks: Following Motif-Net [62], we evaluate the model on three standard VidVRD tasks: scene graph detection (**SGDet**), scene graph classification (**SGCls**), and predicate classification (**PredCls**). Specifically, SGDet detects object trajectories from raw videos and classifies the relationships between these objects. SGCls classifies the objects within the provided ground-truth trajectories and then predicts the relationships between these objects. PredCls predicts the relationships between known objects, where both the ground-truth trajectories and corresponding object categories are provided.

4) Metrics: We use mean Average Precision (**mAP**) and Recall@K (**R@K**) with K = 50, 100 as evaluation metrics for relationship classification. The detected triplet is considered correct if it matches a ground-truth triplet and the IoU between

the trajectories is greater than a threshold (*i.e.,* 0.5). These metrics are applied across all tasks. For SGDet and SGCls tasks, we introduce an additional metric, called mean Average Precision of object trajectory (**mAP**$_o$), to evaluate the quality of object trajectories.

B. Implementation Details

For all experiments, video frames are sampled every 30 frames. We adopt the ViT-L/14 version of CLIP with fixed parameters.

For the query-based Transformer decoder, we use six Transformer layers and 300 object queries. The temperature parameter τ in both Eq. 5 and Eq. 10 is set to 0.01. The threshold ϵ used to filter the bounding boxes is set to 0.35. For the auxiliary object classifier, we use eight tokens each for learnable continuous prompts and learnable conditional prompts, positioning the object token [OBJ] at the end of the sequence. The weighting factors α and β in Eq. 11 are set to 0.3 and 0.6, respectively. The coefficient λ_s in Eq. 16 is set to two. The coefficients λ_1, λ_2, λ_3, λ_4 and λ_5 in Eq. 17 are set to three, five, five, two and two, respectively. The number of Transformer blocks for spatio-temporal visual prompting is set to one for VidVRD and two for VidOR. For the open-vocabulary relationship classifier, we use eight tokens each for learnable continuous prompts and learnable conditional prompts, positioning the relationship token [REL] at 75% of the token length. The coefficients γ and δ in Eq. 23 are set to 0.2 and 0.1, respectively.

For all training steps in Sec. III-D, we use the AdamW [63] algorithm for optimization. In step one, we initialize the query-based Transformer decoder, object embedding head, and bounding box regression head with pre-trained parameters from MS-COCO [64] (excluding novel object categories in Open-VidVRD), while the relationship classification head is initialized with random parameters. The learning rate is set to 1e-5, and the model is trained for ten epochs with a batch size of 16. In step two, the auxiliary object classifier is trained for five epochs with a learning rate of 1e-3 and a batch size of 12. In step three, the open-vocabulary relationship classifier is trained with an initial learning rate of 1e-4, following a multi-step decay schedule that reduces the learning rate by a factor of 0.1 at epochs 15, 20, and 25, with a batch size of 32. In step four, we fine-tune the end-to-end framework for 5 epochs with an initial learning rate of 1e-5 and a batch size of one.

C. Comparison with Existing Methods

We compare our method with existing Open-VidVRD methods, including RePro [2], VidVRD-II [23], CLIP [8], and ALPro [36], and our previous work OV-MMP [17]. Existing methods rely on trajectory detectors pre-trained on closed datasets that encompass all object categories in Open-VidVRD. To ensure a fair comparison, we reproduce the compared existing methods by removing the novel object categories from the training data and retraining the trajectory detector. On the VidOR dataset, the models and codes of ALPro, VidVRD-II, and RePro have not been ready to use, and only the results for R@50 and R@100 are available from their original papers.

Table I and Table II show the comparison results on the VidVRD and VidOR datasets, respectively. We have several interesting observations as follows: (1) Our method outperforms all existing methods that rely on trajectory detectors pre-trained on closed datasets across all metrics on both datasets, demonstrating the superiority of unifying object trajectory detection and relationship classification in an end-to-end framework for Open-VidVRD; (2) For the novel split, our method consistently achieves the best results across all datasets, especially improving mAP$_o$ by 21.94% and mAP by 2.89% on the SGDet task on the VidVRD dataset, and improving mAP$_o$ by 1.22% and mAP by 1.61% on the VidOR dataset. This highlights its strong generalization capability in open-vocabulary scenarios, which benefits from the proposed relationship-aware open-vocabulary trajectory detector and the proposed multi-modal prompting based open-vocabulary relationship classifier; On both SGCls and PredCls tasks where the ground-truth object trajectories are provided, our method also achieves better performance than the existing methods, which suggests that the open-vocabulary relationship classifier benefits from joint learning of trajectory detection and relationship classification. (4) On the VidVRD dataset, for the novel split, our method achieves higher mAP$_o$ on the SGDet task than on the SGCls task. This is because the detected bounding boxes are of high quality, and the classification results of SGDet benefit from ensembling the classification results of both the object queries and the auxiliary object classifier, whereas the results of SGCls rely solely on the auxiliary object classifier. In contrast, for the VidOR dataset, the mAP$_o$ results on the SGDet task are limited by the trajectory association due to more blurs and occlusions in longer videos.

D. Ablation Studies

1) Effectiveness of End-to-end Training: To evaluate the effectiveness of the end-to-end training of the trajectory detector and relationship classifier, we design a separate training strategy for comparison, where the trajectory detector is trained using the frame-wise object and relationship annotations, and the relationship classifier is trained using the ground-truth trajectories and video-level relationship annotations. Table III shows the results on the VidVRD and VidOR datasets, and the proposed end-to-end training performs better in both trajectory detection and relationship classification, further verifying the advantage of unifying trajectory detection and relationship classification.

2) Effectiveness of Relationship-aware Open-vocabulary Trajectory Detector: We propose a relationship query and a corresponding auxiliary relationship loss (denoted as "Rna") to help the trajectory detector explicitly perceive the relationships between objects. We further use an auxiliary object classifier (denoted as "Aoc") to enhance the object classification of the trajectory detector. The ablation study results of "Rna" and "Aoc" on the VidVRD dataset are shown in Table IV. It is interesting to observe that both the proposed relationship query with the corresponding loss and the introduced object classifier enhance the relationship detection performance.

TABLE I
RESULTS OF DIFFERENT METHODS ON THE VIDVRD DATASET.

Split	Method	SGDet				SGCls				PredCls		
		mAP_o	mAP	R@50	R@100	mAP_o	mAP	R@50	R@100	mAP	R@50	R@100
Novel	ALPro	10.36	0.98	2.79	4.33	21.06	3.69	7.27	8.92	4.09	9.42	10.41
	CLIP	14.37	2.13	3.26	4.50	24.96	3.84	6.03	9.44	4.54	7.27	11.74
	VidVRD-II	10.36	3.11	7.93	11.38	21.06	5.70	13.22	18.34	7.35	18.84	26.44
	RePro	10.36	5.87	12.75	16.23	21.06	10.32	19.17	25.28	12.74	25.12	33.88
	OV-MMP	14.37	12.15	13.72	15.21	24.96	17.57	21.98	28.43	21.14	30.41	37.85
	Ours	**36.31**	**15.04**	**16.03**	**18.18**	**31.73**	**17.96**	**30.74**	**36.86**	**21.65**	**35.37**	**43.64**
All	ALPro	18.18	3.03	2.57	3.11	68.99	3.92	3.88	4.75	4.97	4.50	5.79
	CLIP	34.61	4.86	2.97	3.55	70.39	5.80	4.37	5.38	6.49	5.21	6.54
	VidVRD-II	18.18	12.66	9.72	12.50	68.99	17.26	14.93	19.68	19.73	18.17	24.90
	RePro	18.18	21.12	12.63	15.42	68.99	30.15	19.75	25.00	34.90	25.50	32.49
	OV-MMP	34.61	22.10	13.26	16.08	70.39	29.38	23.56	28.89	38.08	30.47	37.46
	Ours	**52.72**	**26.34**	**16.48**	**19.54**	**74.25**	**31.95**	**25.96**	**31.66**	**39.83**	**31.66**	**39.69**

TABLE II
RESULTS OF DIFFERENT METHODS ON THE VIDOR DATASET. FOR ALPRO, VIDVRD-II, AND REPRO, ONLY THE RESULTS OF R@50 AND R@100 ON THE **SGCLS** AND **PREDCLS** TASKS ARE AVAILABLE FROM THEIR ORIGINAL PAPERS.

Split	Method	SGDet				SGCls				PredCls		
		mAP_o	mAP	R@50	R@100	mAP_o	mAP	R@50	R@100	mAP	R@50	R@100
Novel	ALPro	-	-	-	-	-	-	3.17	3.74	-	8.35	9.79
	CLIP	1.11	0.17	0.68	0.77	6.04	0.43	1.79	2.36	1.08	5.48	7.20
	VidVRD-II	-	-	-	-	-	-	1.44	2.01	-	4.32	4.89
	RePro	-	-	-	-	-	-	2.01	2.30	-	7.20	8.35
	OV-MMP	1.11	0.84	1.44	1.44	6.04	2.40	5.48	6.92	3.58	9.22	11.53
	Ours	**2.33**	**2.45**	**4.79**	**4.79**	**6.83**	**2.72**	**5.76**	**8.65**	**4.11**	**9.80**	**14.41**
All	ALPro	-	-	-	-	-	-	0.95	1.32	-	2.61	3.66
	CLIP	3.38	0.22	0.35	0.51	25.86	0.63	0.74	0.99	1.29	1.71	3.13
	VidVRD-II	-	-	-	-	-	-	9.40	12.78	-	24.81	34.11
	RePro	-	-	-	-	-	-	10.03	12.91	-	27.11	35.76
	OV-MMP	3.38	7.15	6.54	8.29	25.86	24.00	23.04	30.14	38.52	33.44	43.80
	Ours	**12.99**	**11.08**	**8.43**	**9.82**	**26.17**	**25.21**	**23.78**	**30.17**	**39.75**	**33.68**	**43.87**

TABLE III
PERFORMANCE (mAP_o AND mAP) OF ABLATION STUDY FOR END-TO-END TRAINING ON THE VIDVRD AND VIDOR DATASETS.

Dataset	End-to-end training	Novel		All	
		mAP_o	mAP	mAP_o	mAP
VidVRD		33.06	14.43	46.76	25.39
	✓	**36.31**	**15.04**	**52.72**	**26.34**
VidOR		1.00	1.99	10.14	10.45
	✓	**2.33**	**2.45**	**12.99**	**11.08**

TABLE IV
PERFORMANCE (mAP_o AND mAP) OF ABLATION STUDY FOR THE RELATIONSHIP-AWARE OPEN-VOCABULARY TRAJECTORY DETECTOR. "RNA" DENOTES THE RELATIONSHIP QUERY AND CORRESPONDING AUXILIARY RELATIONSHIP LOSS. "AOC" DENOTES THE AUXILIARY OBJECT CLASSIFIER.

Rna	Aoc	Novel		All	
		mAP_o	mAP	mAP_o	mAP
		26.17	14.23	43.09	24.57
✓		27.33	14.56	46.16	25.36
	✓	30.29	14.58	48.11	25.47
✓	✓	**36.31**	**15.04**	**52.72**	**26.34**

TABLE V
PERFORMANCE (MAP) OF ABLATION STUDY FOR MULTI-MODAL PROMPTING ON THE VIDVRD DATASET. "VIS" AND "LAN" DENOTE VISUAL PROMPTING AND LANGUAGE PROMPTING, RESPECTIVELY.

Vis	Lan	Novel		All	
		SGDet	PredCls	SGDet	PredCls
		9.14	13.36	22.86	35.24
✓		11.46	14.59	24.28	36.64
	✓	9.72	15.60	24.81	38.38
✓	✓	**15.04**	**21.65**	**26.34**	**39.83**

TABLE VI
PERFORMANCE (MAP) OF ABLATION STUDY FOR THE SPATIO-TEMPORAL VISUAL PROMPTING ON THE VIDVRD DATASET. "SPA" AND "TEM" DENOTE SPATIAL MODELING AND TEMPORAL MODELING, RESPECTIVELY.

Spa	Tem	Novel		All	
		SGDet	PredCls	SGDet	PredCls
		9.72	15.60	24.81	38.38
✓		12.14	17.91	25.02	37.22
	✓	11.58	16.59	23.47	34.32
✓	✓	**15.04**	**21.65**	**26.34**	**39.83**

3) Effectiveness of Multi-modal Prompting: To evaluate the multi-modal prompting, we replace the spatio-temporal visual prompting (denoted as "Vis") with linear layers and replace the vision-guided language prompting (denoted as "Lan") with handcraft language prompting for comparison. The results on the VidVRD dataset are reported in Table V, demonstrating the effectiveness of the proposed visual promoting and language prompting.

4) Effectiveness of Spatio-temporal Visual Prompting: To further evaluate the spatio-temporal visual prompting, we replace the spatial modeling module (denoted as "Spa") or

TABLE VII
PERFORMANCE (mAP$_o$ AND mAP) OF ABLATION STUDY FOR THE VISION-GUIDED LANGUAGE PROMPTING ON THE VIDVRD DATASET.

Variants	Novel		All	
	mAP$_o$	mAP	mAP$_o$	mAP
Manual	31.97	11.46	49.69	24.28
Continuous	35.66	12.79	51.86	25.56
Conditional	34.83	13.40	50.96	25.52
Ours	**36.31**	**15.04**	**52.72**	**26.34**

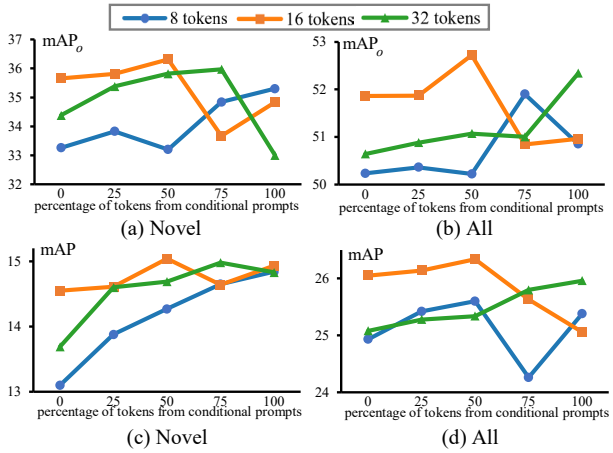


Fig. 3. Results of different token numbers of vision-guided language prompts on the VidVRD dataset. Different colors denote different token numbers, *i.e.*, the blue, orange, and green colors represent the 8,16, and 32 tokens. The horizontal axis represents the percentage of tokens from conditional prompts, *i.e.*, from 0 (all tokens are from learnable continuous prompts) to 100% (all tokens are from learnable conditional prompts). (a) and (b) show the results of using different tokens on the mAP$_o$ metric in the auxiliary object classifier. (c) and (d) show the results of using different tokens on the mAP metric in the open-vocabulary relationship classifier.

TABLE VIII
PARAMETER ANALYSIS RESULTS (mAP$_o$ AND mAP) OF ϵ ON THE VIDVRD DATASET.

ϵ	Novel		All	
	mAP$_o$	mAP	mAP$_o$	mAP
0.20	33.57	14.18	51.93	25.91
0.25	34.81	14.93	52.40	26.19
0.30	35.15	14.89	52.69	26.30
0.35	**36.31**	**15.04**	**52.72**	**26.34**
0.40	34.86	14.61	50.27	25.98
0.45	32.24	12.95	45.90	20.89

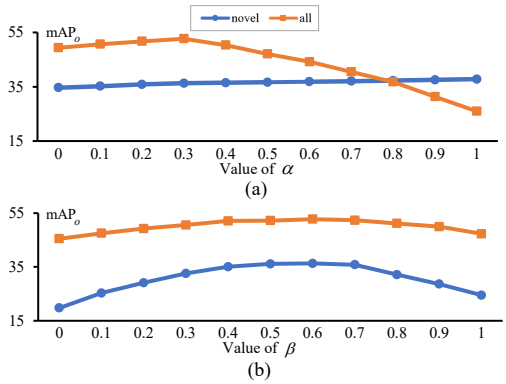


Fig. 4. Results of different values of the hyperparameters α and β on the VidVRD dataset. The horizontal axis represents the values of the parameter, and the vertical axis represents the mAP$_o$ performance. (a) shows the results of different values of α while keeping $\beta = 0.5$. (b) shows the results of different values of β while keeping $\beta = 0.3$.

the temporal modeling module (denoted as "Tem") with linear layers. According to the results presented in Table VI, our method achieves a 2.9% improvement in mAP on the novel split for the SGDet task when performing both spatial and temporal modeling. Furthermore, we observe that the performance drops significantly when only temporal modeling is performed without incorporating spatial modeling. This is in line with expectations, as it is difficult to recognize object relationships based only on the dynamic state changes of individual objects.

5) Effectiveness of Vision-guided Language Prompting: To further evaluate the vision-guided language prompting, we design three variants of our method for comparison: (1) "Manual" involves pre-defined templates for the auxiliary object classifier (*i.e.*, "An image of a [OBJ]") and the relationship classifier (*i.e.*, "An image of a person or object [REL] something" for subjects, "An image of something [REL] a person or object" for objects, and "An image of the visual relationship [REL] between two objects" for background); (2) "Continuous" involves learnable continuous prompts; (3) "Conditional" tailors all prompts to input visual features. The results in Table VII demonstrate that integrating the proposed vision-guided language prompting ("Ours") into the auxiliary object classifier and the relationship classifier significantly enhances the performances of object trajectory classification and relationship classification. Notably, there is an improvement of over 1.6% in mAP in the novel split when using detected trajectories.

E. Hyperparameters

1) The Token Number of Vision-guided Language Prompts: To analyze the effects of different token numbers of vision-guided language prompts on performance, we conduct experiments using 8, 16, and 32 tokens for comparison. We also set the percentage of tokens from the learnable conditional prompts to 0, 25%, 50%, 75%, and 100% for comparison. Figure 3 shows the results of mAP$_o$ and mAP on the VidVRD dataset. We observe that as the number of tokens increases, the performance first increases and then decreases, with the best performance when the number of tokens is 16. We also observe that as the percentage of tokens from learnable conditional prompts increases, the results first increase and then become unstable, and the result is best when half of the tokens come from learnable conditional prompts. These observations highlight the importance of combining task-specific knowledge and visual cues, further validating the effectiveness of the proposed vision-guided prompting in combining learnable continuous prompts and learnable conditional prompts.

2) The Filtering Threshold of Bounding Boxes : To analyze the effect of the filtering threshold of bounding boxes in frame-wise open-vocabulary object detection, *i.e.*, the hyper-parameter ϵ, we conduct experiments by varying the value of ϵ in $\{0.20, 0.25, 0.30, 0.35, 0.40, 0.45\}$. The results on the VidVRD dataset are shown in Table VIII. From these results,

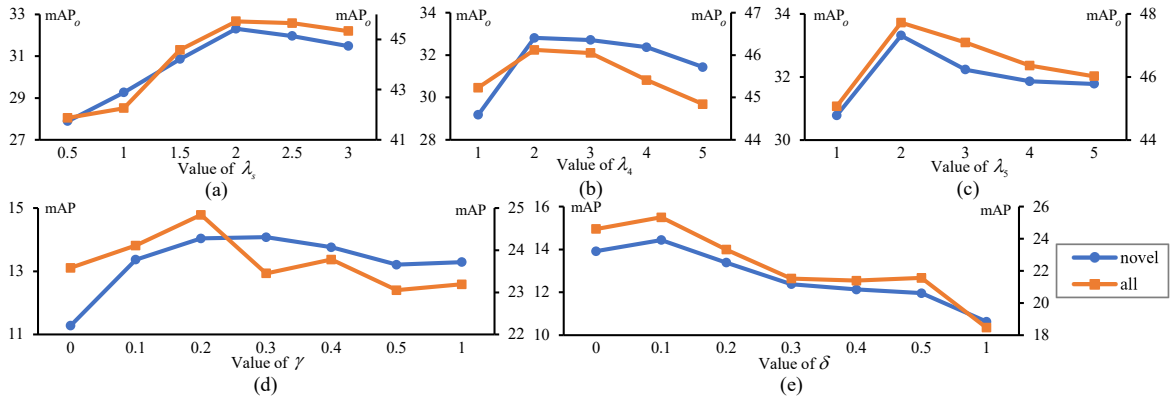


Fig. 5. Results of different values of loss function coefficients on the VidVRD dataset. The horizontal axis represents the values of the coefficients. The left and right vertical axes represent the results of the **novel** and **all** categories. (a) shows the results of different values of λ_s while keeping $\lambda_4 = 1$ and $\lambda_5 = 1$. (b) shows the results of different values of λ_4 while keeping $\lambda_s = 2$ and $\lambda_5 = 1$. (c) shows the results of different values of λ_5 while keeping $\lambda_4 = 2$ and $\lambda_5 = 2$. (d) shows the results of different values of γ while keeping $\delta = 0$. (e) shows the results of different values of δ while keeping $\gamma = 0.2$.

we observe that the performance initially improves as the threshold increases, but then decreases. This is because as ϵ increases, the exclusion of more false positive bounding boxes enhances performance. However, beyond a certain point, further increases in ϵ begin to eliminate true positive boxes, leading to a degradation in performance. The optimal value of ϵ is 0.35.

3) The Coefficients for Ensembling Object Classification Results: To analyze the effect of the coefficients for ensembling object classification results in frame-wise open-vocabulary object classficiation, *i.e.*, the hyperparameters α and β in Eq. 11, we conduct experiments by varying the value of α in the range of $\{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$ while keeping $\beta = 0.5$, and the results on the VidVRD dataset are shown in Figure 4 (a). We observe that the optimal performance is achieved when α is set to 0.3. Then we vary the value of β in the same range while keeping $\alpha = 0.3$, and the results are shown in Figure 4 (b). The overall performance reaches its peak value when β is set to 0.6. It is worth noting that the ensemble is more effective for the novel categories, which can be seen from the significant impact of β on the mAP$_o$ results, highlighting the positive impact of the rich semantic information in CLIP on these categories.

4) The Coefficients of Loss Functions: According to DETR [56], the coefficients λ_1, λ_2, and λ_3 in Eq. 17 are set to three, five, and five, respectively. To analyze the impact of the other coefficients for loss functions of the relationship-aware open-vocabulary trajectory detector, *i.e.*, λ_s in Eq. 16, and λ_4 and λ_5 in Eq. 17, we independently train the trajectory detector and vary λ_s in $\{0.5, 1.0, 1.5, 2.0, 2.5, 3.0\}$, while keeping $\lambda_4 = 1$ and $\lambda_5 = 1$. The results on the VidVRD dataset are shown in Figure 5 (a), indicating that the optimal performance is achieved when λ_s is set to two. Then we vary λ_4 in $\{1, 2, 3, 4, 5\}$ with $\lambda_s = 2$ and $\lambda_5 = 1$, and the results are shown in Figure 5 (b), which indicates that the performance peaks when λ_4 is set to two. Subsequently, we vary λ_5 over $\{1, 2, 3, 4, 5\}$ with $\lambda_s = 2$ and $\lambda_4 = 2$. The results in Figure 5 (c) show that the performance is maximized when λ_5 is two.

Similarly, to analyze the effect of coefficients for loss functions of the open-vocabulary relationship classifier, *i.e.*, the hyperparameters γ and δ in Eq. 23, we independently train the relationship classifier and vary γ over $\{0.1, 0.2, 0.3, 0.4, 0.5, 1.0\}$, while keeping $\delta = 0$. The results on the VidVRD dataset are shown in Figure 5 (d), which indicates that optimal performance is achieved when γ is set to 0.2. Then we vary the value of δ in the same range while keeping $\gamma = 0.2$, the results are shown in Figure 5 (e). The best performance occurs when δ is set to 0.1.

F. Qualitative Analysis

1) Trajectory Visualization: We visualize the trajectories generated by different methods on the VidVRD dataset. Figure 6 (a) shows a cat observing a lizard. RePro [2] and OV-MMP [17] fail to detect the lizard, which belongs to a novel object category. Moreover, OV-MMP detects an incomplete trajectory of the cat and misidentifies an object out of interest, resulting in subsequent classification errors. In contrast, our method detects both objects accurately and classifies them correctly. Figures 6 (b) and (c) also show that both RePro and OV-MMP fail to detect certain objects, while our method detects all objects correctly without redundancy or omission. These examples demonstrate the strong generalization capability of our method to novel object categories and complex scenes. Figure 6 (d) illustrates a case of severe occlusion, where none of the methods are able to fully detect the trajectory of the panda inside the barrel, suggesting the limitations of our method in such challenging scenarios. In the future, more advanced trajectory association algorithms could help improve trajectory detection performance by better handling occlusions and capturing motion dynamics over time.

2) Feature Distribution Visualization: We visualize the feature distributions of randomly selected 10 predicate categories by projecting the features of the union regions onto a 2D plane using T-SNE [65], to demonstrate how well our spatio-temporal visual prompting method adapts the image encoder of CLIP. As shown in Figure 7, features of our method (the right parts of Figure 7 (a), (b)) within the same categories are pulled

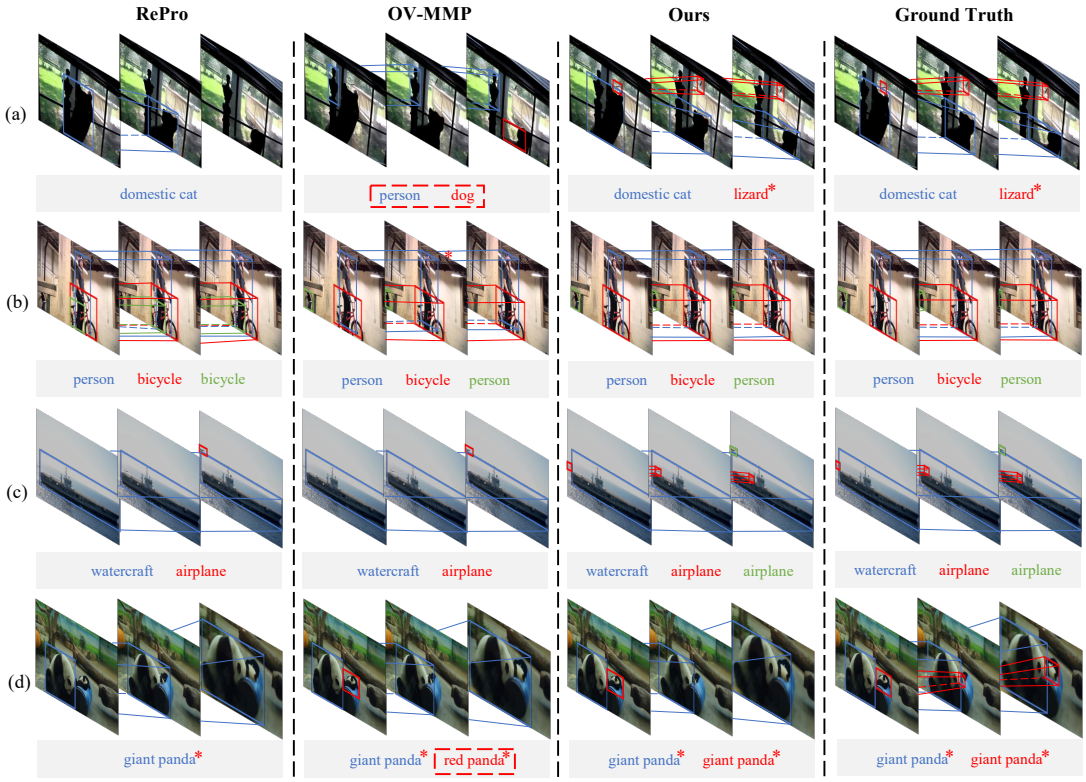
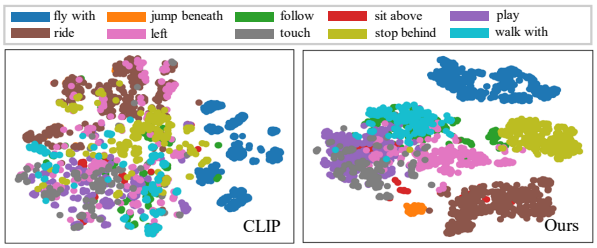
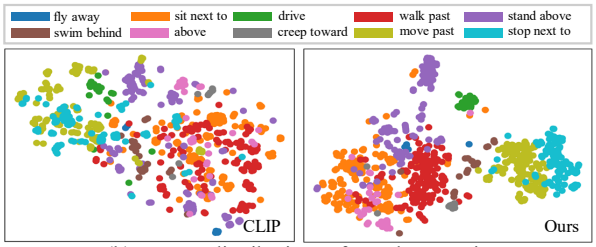


Fig. 6. Visualization of trajectories from different methods. The objects classified incorrectly are enclosed within the red dashed box. * represents the novel object category.



(a) Feature distributions of base categories.



(b) Feature distributions of novel categories.

Fig. 7. Qualitative results of visual feature (union region of subject and object) distributions by T-SNE.

closer while features across different categories are pushed further apart, improving the discrimination on both base and novel categories. These qualitative results further verify the effectiveness of our spatio-temporal visual prompting method.

3) Relationship Visualization: We visualize the correctly detected relationships using ground-truth trajectories on the VidVRD dataset. The comparison includes results from the original CLIP model, our method without language prompting ("w/o language prompting"), our method without visual prompting ("w/o language prompting"), and our method. Figure 8 (a), (b) and (c) show that CLIP performs poorly in relationship classification. Introducing language prompting or visual prompting can detect more correct relationships. Our method achieves the best performance, especially for the novel categories. However, Figure 8 (d) shows a scene with severe occlusion and blur, where our method only identifies one correct relationship. This suggests that there is still room for improvement, particularly in challenging scenarios with significant occlusion or ambiguity. Future advancements in multi-view fusion or more robust temporal modeling could enhance relationship classification accuracy.

4) Case Studies: We conduct case studies on the VidVRD dataset to highlight the strengths of our end-to-end method and show the scenarios that lead to detection failures. As shown in Figure 9 (a) and (b), our method effectively detects object trajectories and accurately classifies the relationships between objects, including both base and novel categories, successfully overcoming challenges such as detecting partially visible objects and handling complex dynamic relationships like "move past". Figure 9 (c) shows an example of video relationship detection in a backlit scene. Due to the strong sunlight, objects in the video appear as black silhouettes, losing visual texture information. Despite these challenging conditions, our method still successfully detects object trajectories, accurately classifies object categories, and correctly predicts most relationship categories, demonstrating its robustness to various scenes. However, due to the lack of texture information, it is difficult

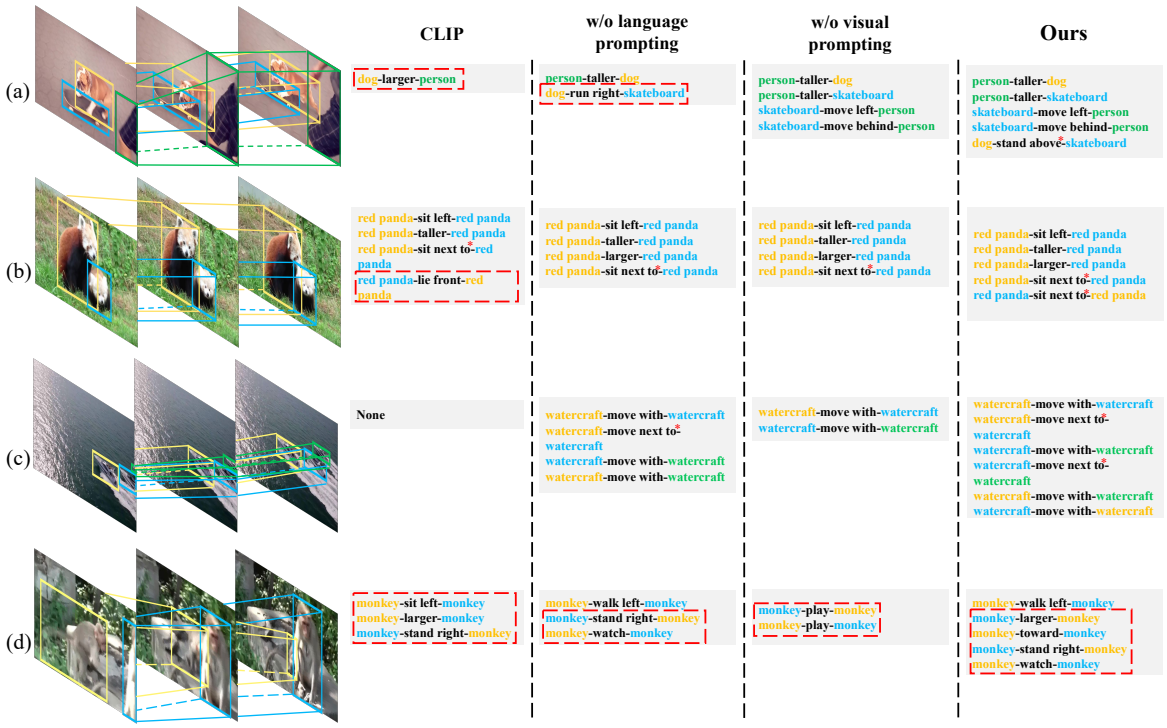


Fig. 8. Visualization of relationship classification results using ground truth trajectories. The relationships classified incorrectly are enclosed within the red dashed box. * represents the novel relationship category.

to determine the front-to-back positions of objects, resulting in some errors in relationship classification. Figure 9 (d) shows a challenging scene where the bodies of two antelopes are facing away from the camera, making object classification difficult. Our method incorrectly classifies the larger antelope on the left into a lion and the striped antelope on the right into a zebra. These misclassifications of objects lead to subsequent errors in triplet detection, despite the correct classification of the relationship categories.

G. Cross-dataset Evaluation

To evaluate the effectiveness of our method in detecting video relationships in real-world scenes, which exhibit a significant domain gap from the training set, we conduct a cross-dataset evaluation by training on the base categories of the VidOR dataset and testing directly on the VidVRD dataset, where categories that overlap with the training data are excluded. In terms of categories, only 18 object categories in the VidVRD dataset appear in the novel categories of the VidOR dataset, while 17 object categories do not appear. Similarly, only 14 relationship categories in VidVRD appear in the novel categories of VidOR, while 118 relationship categories do not overlap. Additionally, the average video length differs significantly, with VidVRD videos averaging 9.7 seconds and VidOR videos averaging 34.6 seconds. These significant discrepancies highlight the challenge of generalizing models to unseen object categories, unseen relationship categories, and unfamiliar video scenes, providing a rigorous evaluation of our method's performance in real-world video relationship detection scenarios.

TABLE IX
COMPARISON OF CROSS-DATASET TRANSFERRED MODELS AND UPPER BOUND MODELS ON THE SGDET TASK OF THE VIDVRD DATASET.

Setting	Method	mAP_o	mAP
Cross dataset	ALPro	3.88	0.29
	VidVRD-II	3.88	0.88
	RePro	3.88	1.11
	OV-MMP	2.74	1.14
	Ours	**13.65**	**6.59**
Upper bound	ALPro	10.36	0.98
	VidVRD-II	10.36	3.11
	RePro	10.36	5.87
	OV-MMP	14.37	12.15
	Ours	**36.31**	**15.04**

We present mAP_o and mAP for object and relationship categories not seen in the training data as the results of cross-dataset experiments, using the results from the novel split without cross-dataset training as the upper bound, as shown in Table IX. It can be observed that our method achieves the best results on all metrics in cross-dataset experiments, even surpassing the upper bound results of ALPro, VidVRD-II, and RePro, demonstrating the strong generalization capability of our end-to-end framework.

V. CONCLUSION

We present an end-to-end Open-VidVRD framework that unifies trajectory detection and relationship classification, eliminating the dependency on trajectory detectors pre-trained on closed datasets in the previous methods. Under this framework, we propose a relationship-aware open-vocabulary trajectory detector that can capture the relationship contexts via

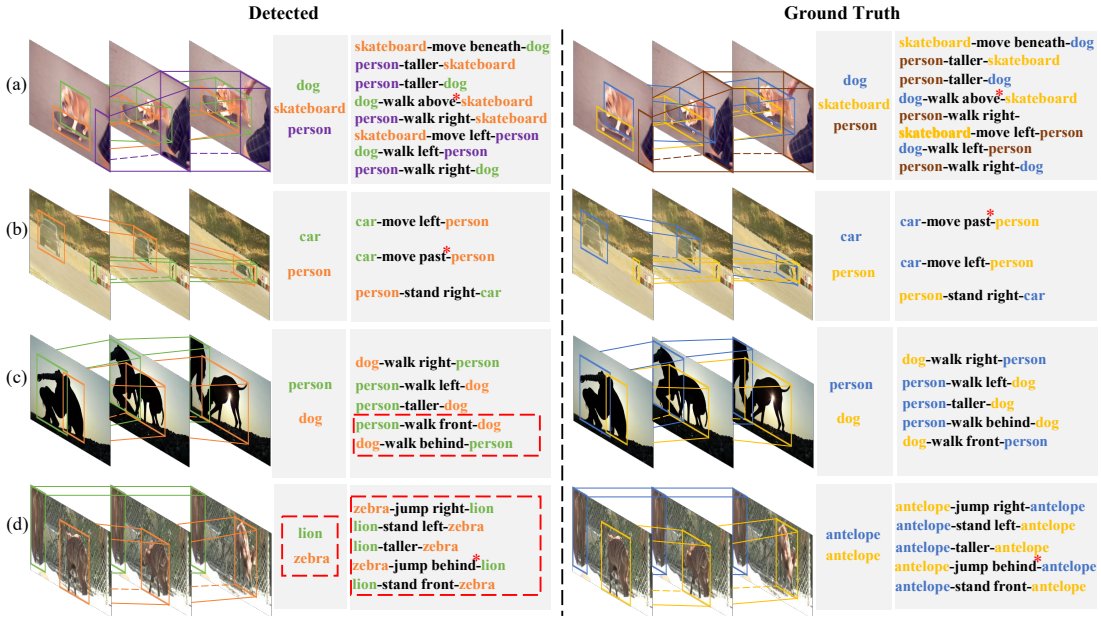


Fig. 9. Examples of detected relationships by our method. The objects or relationship triplets detected incorrectly are enclosed within the red dashed box. * represents the novel relationship category.

a relationship query and a corresponding auxiliary relationship loss to improve the trajectory detection performance. Moreover, we propose an open-vocabulary relationship classifier with a multi-modal prompting method that can prompt CLIP on both the visual and language sides to enhance the generalization to novel categories. Experiments on VidVRD and VidOR datasets demonstrate significant improvements in the overall performance and generalization capability. In the future, we plan to unify trajectory detection and relationship classification within a Transformer decoder to further improve their mutual performance and make our method more practical for processing real-world videos, especially long videos with more complex object relationships.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62072041.

References

[1] X. Shang, T. Ren, J. Guo, H. Zhang, and T.-S. Chua, "Video visual relation detection," in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, pp. 1300–1308, 2017.

[2] K. Gao, L. Chen, H. Zhang, J. Xiao, and Q. Sun, "Compositional prompt tuning with motion cues for open-vocabulary video relation detection," *arXiv preprint arXiv:2302.00268*, 2023.

[3] Y. Zang, W. Li, K. Zhou, C. Huang, and C. C. Loy, "Open-vocabulary detr with conditional matching," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 106–122, Springer, 2022.

[4] S. Li, T. Fischer, L. Ke, H. Ding, M. Danelljan, and F. Yu, "Ovtrack: Open-vocabulary multiple object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5567–5577, 2023.

[5] S. Zheng, B. Xu, and Q. Jin, "Open-category human-object interaction pre-training via language modeling framework," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19392–19402, 2023.

[6] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig, "Scaling up visual and vision-language representation learning with noisy text supervision," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 4904–4916, PMLR, 2021.

[7] H. Pham, Z. Dai, G. Ghiasi, K. Kawaguchi, H. Liu, A. W. Yu, J. Yu, Y.-T. Chen, M.-T. Luong, Y. Wu, *et al.*, "Combined scaling for zero-shot transfer learning," *Neurocomputing*, vol. 555, p. 126658, 2023.

[8] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 8748–8763, PMLR, 2021.

[9] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 12888–12900, PMLR, 2022.

[10] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 19730–19742, PMLR, 2023.

[11] M. Gao, C. Xing, J. C. Niebles, J. Li, R. Xu, W. Liu, and C. Xiong, "Open vocabulary object detection with pseudo bounding-box labels," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 266–282, Springer, 2022.

[12] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui, "Open-vocabulary object detection via vision and language knowledge distillation," *arXiv preprint arXiv:2104.13921*, 2021.

[13] W. Kuo, Y. Cui, X. Gu, A. Piergiovanni, and A. Angelova, "F-vlm: Open-vocabulary object detection upon frozen vision and language models," *arXiv preprint arXiv:2209.15639*, 2022.

[14] B. Ni, H. Peng, M. Chen, S. Zhang, G. Meng, J. Fu, S. Xiang, and H. Ling, "Expanding language-image pretrained models for general video recognition," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 1–18, Springer, 2022.

[15] Z. Weng, X. Yang, A. Li, Z. Wu, and Y.-G. Jiang, "Open-vclip: Transforming clip to an open-vocabulary video model via interpolated weight optimization," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 36978–36989, PMLR, 2023.

[16] M. Xu, Z. Zhang, F. Wei, H. Hu, and X. Bai, "Side adapter network for open-vocabulary semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2945–2954, 2023.

[17] S. Yang, Y. Wang, X. Ji, and X. Wu, "Multi-modal prompting for open-vocabulary video visual relationship detection," in *Proceedings of the*

AAAI Conference on Artificial Intelligence (AAAI), vol. 38, pp. 6513–6521, 2024.

[18] X. Shang, J. Xiao, D. Di, and T.-S. Chua, "Relation understanding in videos: A grand challenge overview," in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, pp. 2652–2656, 2019.

[19] X. Qian, Y. Zhuang, Y. Li, S. Xiao, S. Pu, and J. Xiao, "Video relation detection with spatio-temporal graph," in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, pp. 84–93, 2019.

[20] Y.-H. H. Tsai, S. Divvala, L.-P. Morency, R. Salakhutdinov, and A. Farhadi, "Video relationship reasoning using gated spatio-temporal energy graph," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10424–10433, 2019.

[21] C. Liu, Y. Jin, K. Xu, G. Gong, and Y. Mu, "Beyond short-term snippet: Video relation detection with spatio-temporal global context," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10840–10849, 2020.

[22] Y. Cong, W. Liao, H. Ackermann, B. Rosenhahn, and M. Y. Yang, "Spatial-temporal transformer for dynamic scene graph generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16372–16382, 2021.

[23] X. Shang, Y. Li, J. Xiao, W. Ji, and T.-S. Chua, "Video visual relation detection via iterative inference," in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, pp. 3654–3663, 2021.

[24] S. Chen, Z. Shi, P. Mettes, and C. G. Snoek, "Social fabric: Tubelet compositions for video relation detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13485–13494, 2021.

[25] L. Xu, H. Qu, J. Kuen, J. Gu, and J. Liu, "Meta spatio-temporal debiasing for video scene graph generation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 374–390, Springer, 2022.

[26] X. Dong, T. Gan, X. Song, J. Wu, Y. Cheng, and L. Nie, "Stacked hybrid-attention and group collaborative learning for unbiased scene graph generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19427–19436, 2022.

[27] X. Lin, C. Shi, Y. Zhan, Z. Yang, Y. Wu, and D. Tao, "Td^2-net: Toward denoising and debiasing for video scene graph generation," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 38, pp. 3495–3503, 2024.

[28] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9268–9277, 2019.

[29] S. Zheng, S. Chen, and Q. Jin, "Vrdformer: End-to-end video visual relation detection with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18836–18846, 2022.

[30] Y. Zhang, Y. Pan, T. Yao, R. Huang, T. Mei, and C.-W. Chen, "End-to-end video scene graph generation with temporal propagation transformer," *IEEE Transactions on Multimedia (TMM)*, 2023.

[31] T. Ren, Y. Chen, Q. Jiang, Z. Zeng, Y. Xiong, W. Liu, Z. Ma, J. Shen, Y. Gao, X. Jiang, *et al.*, "Dino-x: A unified vision model for open-world object detection and understanding," *arXiv preprint arXiv:2411.14347*, 2024.

[32] T. Wu, S. Ge, J. Qin, G. Wu, and L. Wang, "Open-vocabulary spatio-temporal action detection," *arXiv preprint arXiv:2405.10832*, 2024.

[33] Z. Wu, Z. Weng, W. Peng, X. Yang, A. Li, L. S. Davis, and Y.-G. Jiang, "Building an open-vocabulary video clip model with better architectures, optimization and data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[34] T. He, L. Gao, J. Song, and Y.-F. Li, "Towards open-vocabulary scene graph generation with prompt-based finetuning," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 56–73, Springer, 2022.

[35] H. Yuan, S. Zhang, X. Wang, S. Albanie, Y. Pan, T. Feng, J. Jiang, D. Ni, Y. Zhang, and D. Zhao, "Rlipv2: Fast scaling of relational language-image pre-training," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 21649–21661, 2023.

[36] D. Li, J. Li, H. Li, J. C. Niebles, and S. C. Hoi, "Align and prompt: Video-and-language pre-training with entity prompts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4953–4963, 2022.

[37] R. Li, S. Zhang, D. Lin, K. Chen, and X. He, "From pixels to graphs: Open-vocabulary scene graph generation with vision-language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28076–28086, 2024.

[38] L. Li, J. Xiao, G. Chen, J. Shao, Y. Zhuang, and L. Chen, "Zero-shot visual relation detection via composite visual cues from large language models," *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2024.

[39] Q. Yu, J. Li, Y. Wu, S. Tang, W. Ji, and Y. Zhuang, "Visually-prompted language model for fine-grained scene graph generation in an open world," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21560–21571, 2023.

[40] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[41] L. Zhao, L. Yuan, B. Gong, Y. Cui, F. Schroff, M.-H. Yang, H. Adam, and T. Liu, "Unified visual relationship detection with vision and language models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6962–6973, 2023.

[42] F. Zhu, J. Yang, and H. Jiang, "Towards flexible visual relationship segmentation," *arXiv preprint arXiv:2408.08305*, 2024.

[43] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, *et al.*, "Flamingo: a visual language model for few-shot learning," in *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, pp. 23716–23736, 2022.

[44] L. Li, Y.-C. Chen, Y. Cheng, Z. Gan, L. Yu, and J. Liu, "Hero: Hierarchical encoder for video+language omni-representation pre-training," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2046–2065, 2020.

[45] H. Luo, L. Ji, B. Shi, H. Huang, N. Duan, T. Li, J. Li, T. Bharti, and M. Zhou, "Univl: A unified video and language pre-training model for multimodal understanding and generation," *arXiv preprint arXiv:2002.06353*, 2020.

[46] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Learning to prompt for vision-language models," *International Journal of Computer Vision (IJCV)*, vol. 130, no. 9, pp. 2337–2348, 2022.

[47] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Conditional prompt learning for vision-language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16816–16825, 2022.

[48] X. Sun, P. Hu, and K. Saenko, "Dualcoop: Fast adaptation to multi-label recognition with limited annotations," *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, pp. 30569–30582, 2022.

[49] M. Jia, L. Tang, B.-C. Chen, C. Cardie, S. Belongie, B. Hariharan, and S.-N. Lim, "Visual prompt tuning," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 709–727, Springer, 2022.

[50] S. Wang, Y. Duan, H. Ding, Y.-P. Tan, K.-H. Yap, and J. Yuan, "Learning transferable human-object interaction detector with natural language supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 939–948, 2022.

[51] C. Xu, H. Shen, F. Shi, B. Chen, Y. Liao, X. Chen, and L. Wang, "Progressive visual prompt learning with contrastive feature re-formation," *arXiv preprint arXiv:2304.08386*, 2023.

[52] M. U. Khattak, H. Rasheed, M. Maaz, S. Khan, and F. S. Khan, "Maple: Multi-modal prompt learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19113–19122, 2023.

[53] Y. Li, R. Quan, L. Zhu, and Y. Yang, "Efficient multimodal fusion via interactive prompting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2604–2613, 2023.

[54] Y. Xin, J. Du, Q. Wang, K. Yan, and S. Ding, "Mmap: Multi-modal alignment prompt for cross-domain multi-task learning," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 38, pp. 16076–16084, 2024.

[55] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[56] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 213–229, Springer, 2020.

[57] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 3645–3649, 2017.

[58] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2980–2988, 2017.
[59] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 658–666, 2019.
[60] Y. Wang, K. Li, Y. Li, Y. He, B. Huang, Z. Zhao, H. Zhang, J. Xu, Y. Liu, Z. Wang, *et al.*, "Internvideo: General video foundation models via generative and discriminative learning," *arXiv preprint arXiv:2212.03191*, 2022.
[61] H. Xu, G. Ghosh, P.-Y. Huang, D. Okhonko, A. Aghajanyan, F. Metze, L. Zettlemoyer, and C. Feichtenhofer, "Videoclip: Contrastive pre-training for zero-shot video-text understanding," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6787–6800, 2021.
[62] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi, "Neural motifs: Scene graph parsing with global context," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5831–5840, 2018.
[63] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.
[64] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 740–755, Springer, 2014.
[65] G. E. Hinton and S. Roweis, "Stochastic neighbor embedding," in *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, vol. 15, pp. 833–840, 2002.

Yongqi Wang received the B.S. degree in computer science in 2023 from the Beijing Institute of Technology (BIT), Beijing, China, where he is currently working toward the M.S. degree in computer science. His research interests include vision and language, and video understanding.

Xinxiao Wu (Member, IEEE) received the B.S. degree in computer science from the Nanjing University of Information Science and Technology, Nanjing, China, in 2005, and the Ph.D. degree in computer science from the Beijing Institute of Technology (BIT), Beijing, China, in 2010. From 2010 to 2011, she was a Postdoctoral Research Fellow with Nanyang Technological University, Singapore. She is currently a Full Professor with the School of Computer Science, BIT. Her research interests include vision and language, machine learning, and video understanding. She serves on the Editorial Boards of the IEEE Transactions on Multimedia.

Shuo Yang is an associate professor now at Shenzhen MSU-BIT University, Guangdong, China, from 2024. He received a B.S. degree in computer science from the Beijing Union University, Beijing, China, in 2014, an M.S. degree in computer science from the Institute of Software, Chinese Academic of Science, Beijing, China, in 2017, and a Ph.D. degree in computer science from the Beijing Institute of Technology (BIT), Beijing, China, in 2024. His research interests include visual understanding, and vision and language.

Jiebo Luo (Fellow, IEEE) is currently the Albert Arendt Hopeman Professor of Engineering and Professor of Computer Science with the University of Rochester, Rochester, NY, USA, which he joined in 2011 after a prolific career of 15 years with the Kodak Research Laboratories. He has authored over 600 technical papers and holds more than 90 U.S. patents. His research interests include computer vision, natural language processing (NLP), machine learning, data mining, computational social science, and digital health. Dr. Luo is a fellow of NAI, ACM, AAAI, SPIE, and IAPR. He has been involved in numerous technical conferences, including serving as Program Co-Chair for ACM Multimedia 2010, IEEE CVPR 2012, ACM ICMR 2016, and IEEE ICIP 2017, and General Co-Chair for ACM Multimedia 2018 and IEEE ICME 2024. He has served on the editorial boards of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (TPAMI), IEEE TRANSACTIONS ON MULTIMEDIA (TMM), IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (TCSVT), IEEE TRANSACTIONS ON BIG DATA (TBD), ACM Transactions on Intelligent Systems and Technology (TIST), Pattern Recognition, Knowledge and Information Systems (KAIS), Machine Vision and Applications, and Intelligent Medicine. He was an Editor-in-Chief of IEEE TRANSACTIONS ON MULTIMEDIA from 2020 to 2022.