SlotVTG: Object-Centric Adapter for Generalizable Video Temporal Grounding

Jiwook Han[1*] Geo Ahn[1*] Youngrae Kim[2*] Jinwoo Choi[1†]
[1]Kyung Hee University [2]University of Southern California
{mreraser,ahngeo11,jinwoochoi}@khu.ac.kr, youngrae@usc.edu

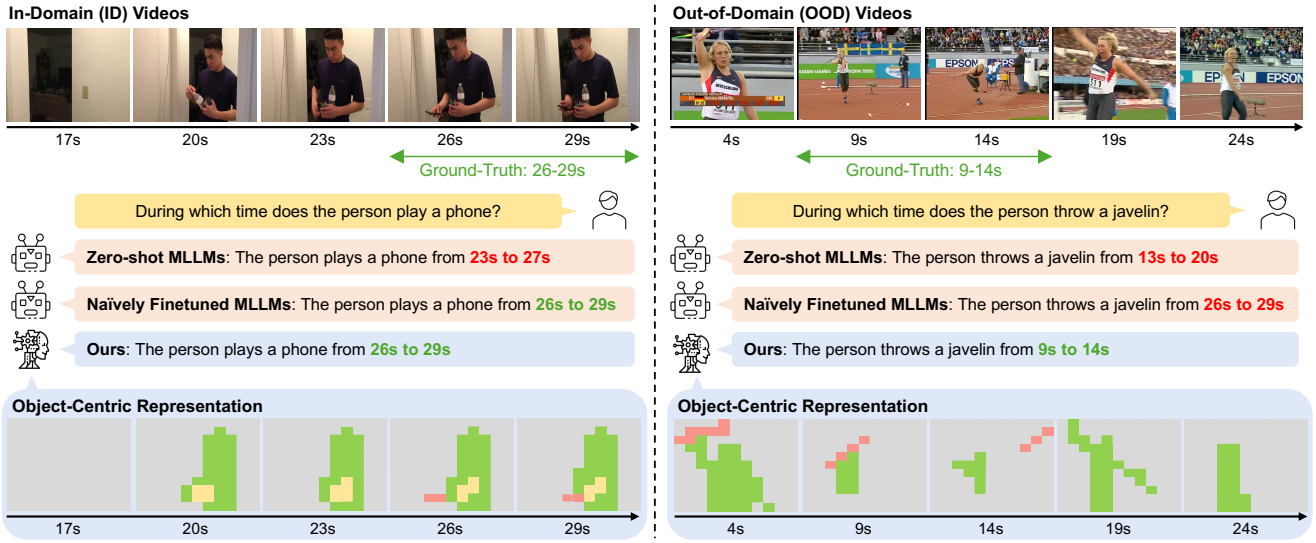

Figure 1. **Motivation.** Zero-shot MLLMs lack fine-grained temporal understanding, producing incorrect timestamps in both settings. Fine-tuning on a VTG dataset resolves this for In-Domain (ID) videos (left), but on Out-of-Domain (OOD) videos the model predicts timestamps based on dataset-specific ***shortcuts*** rather than the actual visual content (right). Our method leverages ***object-centric visual representations*** (bottom) that decompose each frame into semantic entities, encouraging genuine visual grounding in both seen and unseen settings.

Abstract

Multimodal Large Language Models (MLLMs) have shown strong performance on Video Temporal Grounding (VTG). However, their coarse recognition capabilities are insufficient for fine-grained temporal understanding, making task-specific fine-tuning indispensable. This fine-tuning causes models to memorize dataset-specific shortcuts rather than faithfully grounding in the actual visual content, leading to poor Out-of-Domain (OOD) generalization. Object-centric learning offers a promising remedy by decomposing scenes into entity-level representations, but existing approaches require re-running the entire multi-stage training pipeline from scratch. We propose SlotVTG, a framework that steers MLLMs toward object-centric, input-grounded visual reasoning at minimal cost. SlotVTG introduces a lightweight slot adapter that decomposes visual tokens into abstract slots via slot attention and reconstructs the original sequence, where objectness priors from a self-supervised vision model encourage semantically coherent slot formation. Cross-domain evaluation on standard VTG benchmarks demonstrates that our approach significantly improves OOD robustness while maintaining competitive In-Domain (ID) performance with minimal overhead.

1. Introduction

Video Temporal Grounding (VTG), the task of localizing temporal moments in untrimmed videos given natural language queries, has been predominantly addressed by DETR-based specialist models [24, 30, 36, 44]. Recently, Multimodal Large Language Models (MLLMs) have emerged as a compelling alternative [13, 17, 34, 42, 45, 55],

*Equally contributed first authors. †Corresponding author.

owing to their powerful visual representations learned from massive image and video corpora.

However, naively applying MLLMs to VTG yields suboptimal results. Temporal grounding demands fine-grained temporal understanding that goes beyond the coarse recognition capabilities of general-purpose MLLMs, making task-specific fine-tuning indispensable [17, 42, 45, 55]. Yet VTG annotations require precise start-end timestamps for each query, making large-scale data collection prohibitively expensive and preventing models from being exposed to diverse data distributions. This leads to severe overfitting to dataset-specific shortcuts, as these limited-scale datasets inevitably contain various forms of bias, such as temporal location bias [6, 14, 39], query text bias [6, 19, 26], and appearance bias [3, 40]. Consequently, these models exhibit severe performance degradation when encountering Out-of-Domain (OOD) test samples (Figs. 1 and 2(a)).

In this work, we focus on investigating how the *visual domain gap* leads to VTG performance degradation in OOD settings through comprehensive empirical analyses. As shown in Fig. 2(b), the fine-tuned MLLM exhibits a performance gap of around 13% on OOD samples, depending on whether they are visually similar or dissimilar to the source dataset. To further diagnose whether the model genuinely grounds in visual contents, we inject noise into ground-truth segments and compare against perturbing random non-ground-truth segments (Fig. 2(c)). On ID, ground-truth perturbation causes a significantly larger drop than random perturbation, confirming the model *does* attend to the target moment. On OOD, however, the two cause nearly identical drops, indicating the model is *not* actively grounding the visual inputs but has rather lost its scene recognition capability for unseen domains.

To encourage the model to genuinely ground on visual contents regardless of domain shifts, it is crucial to extract domain-invariant visual cues. A promising direction is *object-centric learning* [32], which decomposes scenes into discrete entity-level representations and has been shown to improve domain generalization in video understanding tasks in MLLMs [7, 49]. However, these approaches integrate object-centric representations between the visual encoder and the language model, requiring the entire vision-language alignment and instruction tuning pipeline to be retrained from scratch.

We propose **SlotVTG**, a framework that brings object-centric representation learning into the MLLM framework at minimal cost. SlotVTG introduces a lightweight **Slot Adapter** that decomposes visual tokens into a compact set of abstract slots via slot attention, then reconstructs the original token sequence from these slots. This bottleneck guides visual information through entity-level representations, encouraging the model to suppress spurious correlations and instead attend to the actual visual content relevant to the query. To further encourage semantically coherent tokens to be grouped into the same slot, we introduce a **Slot Alignment (SA) loss** that aligns the slot attention maps with self-supervised objectness priors from pre-trained DINOv2 [38] features.

We validate our approach through cross-domain evaluation on standard VTG benchmarks, training on one source (*e.g.*, Charades-STA [11] and QVHighlights [24]) and evaluating on different targets. Our experiments demonstrate that Slot Adapter improves OOD robustness while maintaining competitive ID performance, with minimal memory overhead and additional parameters. Our main contributions are as follows:

- We identify that fine-tuned MLLMs memorize dataset-specific visual shortcuts rather than grounding in the actual visual content.
- We propose SlotVTG, a parameter-efficient framework consisting of a Slot Adapter that decomposes visual tokens into entity-level slots, and a Slot Alignment Loss that encourages semantically coherent slot formation via objectness priors from pre-trained DINOv2 features.
- We demonstrate through cross-domain evaluation that SlotVTG significantly improves OOD robustness while maintaining competitive ID performance with minimal overhead.

2. Related Work

2.1. Video Temporal Grounding

Video Temporal Grounding (VTG) aims to localize temporal moments in untrimmed videos given natural language queries. Early approaches rely on proposal-based or regression-based architectures [11, 15, 54, 57]. Inspired by the success of DETR [4] in object detection, Moment-DETR [24] pioneered the use of set prediction for joint moment retrieval and highlight detection, establishing the QVHighlights benchmark. Subsequent DETR-based methods have advanced the paradigm through query-dependent representations [36], event-aware attention [18], unified multi-task frameworks [30], task-reciprocal decoding [44], correlation-guided calibration [35], and joint task exploration [48, 50].

More recently, Multimodal Large Language Models (MLLMs) have emerged as a compelling alternative. VTimeLLM [17] and TimeChat [42] demonstrate that MLLMs can generate temporal boundaries as text tokens through task-specific instruction tuning. This generative paradigm has been extended by interleaved frame-timestamp representations [34], causal event modeling [13], grounded tuning for long videos [55], chain-of-LoRA reasoning [31], and reinforcement learning with verifiable temporal rewards [46]. While these methods improve temporal understanding within MLLMs, they focus on architectural

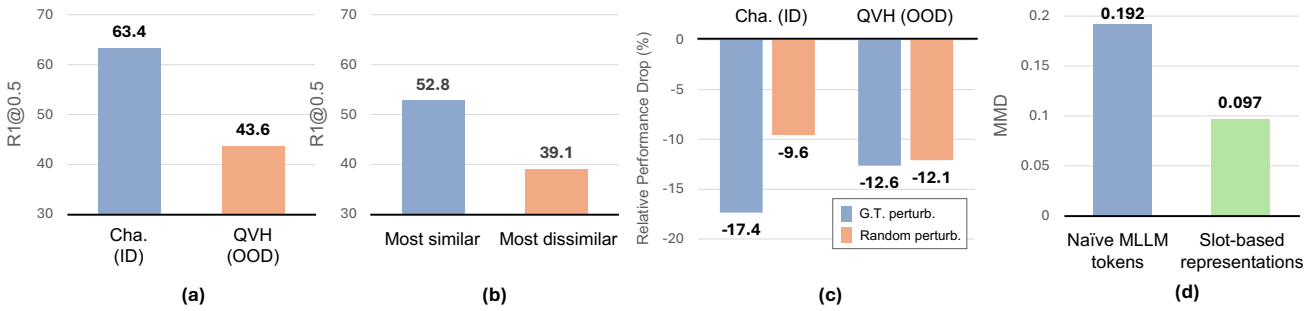

Figure 2. **Observations.** We naively fine-tune Qwen2.5-VL-3B [2] on Charades-STA (Cha.) [11] (source) and evaluate on QVHighlights (QVH) [24] (target). **(a) ID vs. OOD performance.** The model achieves 63.4 R1@0.5 on ID but drops to 43.6 on OOD, confirming severe overfitting to dataset-specific patterns. **(b) Visual similarity analysis.** We extract visual features from the vision encoder and compute cosine similarity between ID and OOD samples; performance on the most similar 20% of OOD samples (52.8) far exceeds the most dissimilar 20% (39.1), indicating that the model fails when visual distribution shifts. **(c) Noise perturbation.** We report R1@0.7 for a stricter localization threshold. On ID, ground-truth perturbation causes a 17.4% drop while random perturbation causes only 9.6%, a significant gap confirming the model attends to the target moment. On OOD, however, the two cause nearly identical drops (12.6% vs. 12.1%), revealing that the model does not attend to the actual visual content under distribution shift. **(d) Domain gap.** MMD distance [12] of our slot-based representations (0.097) is substantially lower than the baseline (0.192), showing that object-centric decomposition reduces the domain gap between source and target distributions.

and training innovations without addressing the fundamental problem of dataset-specific shortcut learning during fine-tuning.

2.2. Bias in Video Understanding

Dataset bias has been widely studied across video understanding tasks. In action recognition, Choi *et al*. [9] reveal that models exploit scene context as a shortcut, achieving high accuracy without attending to the actual action. Li *et al*. [28] formalize representation bias in video datasets and introduce the Diving48 benchmark to mitigate it. Bae *et al*. [1] further address this through disentangled action-scene representations. Beyond action recognition, Lei *et al*. [25] show that single-frame models perform surprisingly well on video-language tasks, exposing static appearance bias.

In the VTG domain specifically, Otani *et al*. [39] demonstrate that blind baselines without video input can match trained models by exploiting annotation distribution patterns. This finding prompted the creation of out-of-distribution evaluation splits [53] and spurred numerous debiasing methods. Causal inference approaches [37, 52] use backdoor adjustment to remove confounding effects of moment location. Adversarial and augmentation strategies [14, 23, 40] synthesize bias-conflict samples or shuffle temporal structure to discourage shortcut exploitation. Chae *et al*. [6] provide a comprehensive benchmark across seven datasets, analyzing annotation bias and query text patterns. These studies collectively reveal that VTG datasets contain diverse biases spanning annotation distributions, language patterns, and visual modalities, and that existing models remain vulnerable to exploiting such shortcuts rather than performing genuine cross-modal grounding.

2.3. Object-Centric Learning

Object-centric learning aims to decompose scenes into discrete entity-level representations. Slot Attention [32] introduces an iterative competitive attention mechanism where learnable slots compete to explain input tokens. DINOSAUR [43] extends slot attention to real-world images by reconstructing self-supervised DINO [5] features instead of raw pixels. In the video domain, SAVi [21] conditions slot initialization on optical flow for temporal consistency, while SlotFormer [47] learns unsupervised visual dynamics through autoregressive slot prediction.

Integrating object-centric representations into vision-language models is an emerging direction. Slot-VLM [49] designs dual-branch object-event slots that decompose video tokens into object-centric and event-centric representations for LLM reasoning. Slot-MLLM [7] combines Q-Former with slot attention to produce discrete object-centric visual tokens for unified multimodal generation. However, both approaches require training the entire vision-language pipeline from scratch, including visual token alignment and instruction tuning. Our work differs in that we introduce a lightweight slot-based adapter that can be attached to existing fine-tuned MLLMs at minimal cost, without modifying the base training pipeline.

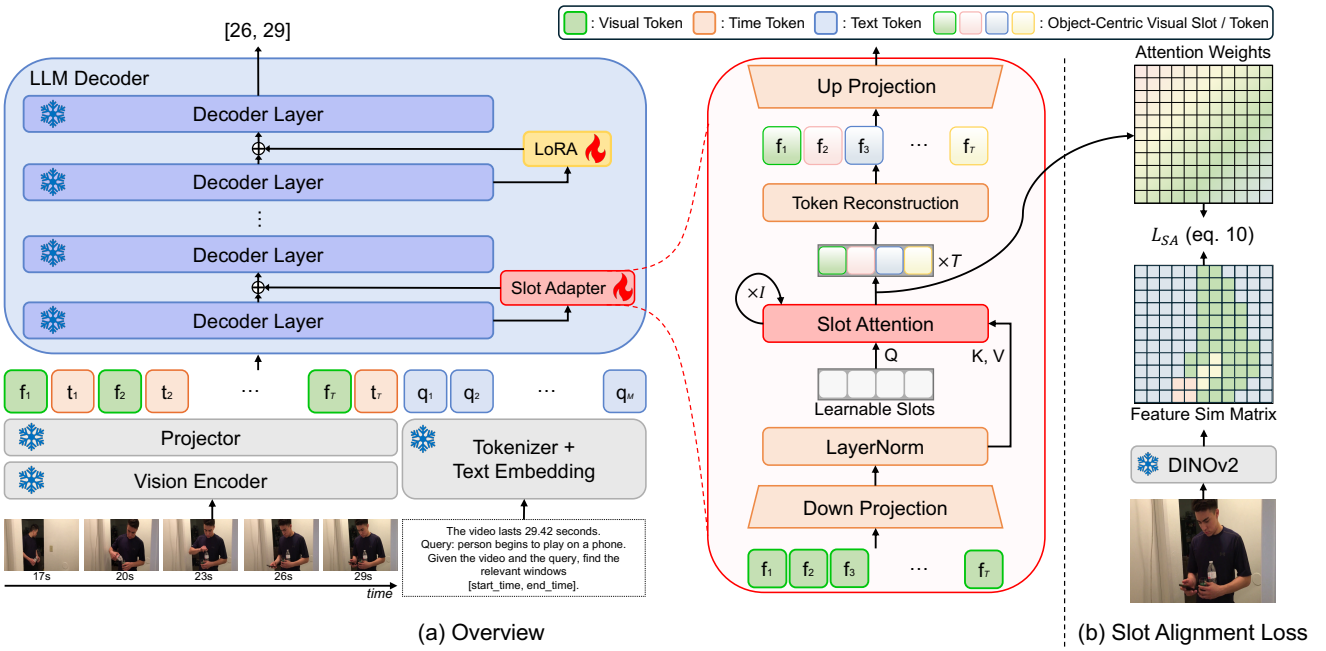

Figure 3. **(a) Overview of SlotVTG.** Video frames are encoded into visual tokens and projected into the LLM decoder. In the early decoder layers, a lightweight *Slot Adapter* decomposes visual tokens into entity-level slots via iterative slot attention, then reconstructs the token sequence. The resulting tokens carry disentangled, entity-aware representations before entering the later layers, which are fine-tuned with LoRA for temporal reasoning and answer generation. Text tokens bypass the Slot Adapter throughout. **(b) Slot Alignment Loss.** Token-pair similarity derived from slot attention weights is aligned with that from a pre-trained DINOv2 model, encouraging semantically coherent tokens to be grouped into the same slot.

3. Preliminaries

3.1. Generative VTG Framework

We follow the generative Video Temporal Grounding (VTG) paradigm, where an MLLM directly generates target timestamps as text tokens. Given T uniformly sampled video frames and a natural language query, we encode each frame into N visual tokens $\mathbf{f}_i \in \mathbb{R}^{N\times D}$ via a frozen vision encoder and linear projection, where D is the hidden dimension of the LLM decoder, and tokenize its timestamp into a short text sequence $\mathbf{t}_i$. Each frame's sampling time in seconds is tokenized into a short text sequence $\mathbf{t}_i$ (e.g., "2.5s"). The input to the LLM decoder is constructed by interleaving each frame's visual tokens with its timestamp tokens, followed by the query tokens $\mathbf{q}$:

$$\mathbf{x} = [\mathbf{f}_1, \mathbf{t}_1, \mathbf{f}_2, \mathbf{t}_2, \ldots, \mathbf{f}_T, \mathbf{t}_T, \mathbf{q}], \tag{1}$$

This interleaved layout has been shown to be effective for temporal grounding [34, 56]. The model autoregressively decodes the target temporal window $[t_{\text{start}}, t_{\text{end}}]$.

3.2. Observations

To understand why naïvely fine-tuned MLLMs fail under distribution shift, we conduct a series of diagnostic experiments. We fine-tune Qwen2.5-VL-3B on Charades-STA [11] and evaluate on QVHighlights [24].

OOD performance degradation. Fig. 2(a) compares ID and OOD performance. The fine-tuned model achieves 63.4 R1@0.5 on ID but only 43.6 on OOD—a 31.2% relative drop. This confirms that the model overfits to source-domain patterns rather than learning generalizable temporal grounding.

Visual similarity matters. To investigate whether this degradation correlates with visual distribution shift, we extract features from the vision encoder and rank OOD samples by cosine similarity to the training set. As shown in Fig. 2(b), the most similar 20% of OOD samples achieve 52.8 R1@0.5, while the most dissimilar 20% drop to 39.1. This reveals that the model's predictions degrade proportionally with visual domain distance, suggesting it relies on surface-level visual patterns seen during training.

The model ignores visual content on OOD. We design a noise perturbation experiment to directly test whether the model attends to the visual content within ground-truth segments. Specifically, we add Gaussian noise to the visual tokens corresponding to the annotated temporal window and measure the performance change. We report R1@0.7 for

this experiment, as a stricter IoU threshold better captures whether the model precisely localizes the target moment. As shown in Fig. 2(c), corrupting the ground-truth segment on ID causes a 17.4% drop, while corrupting random non-ground-truth segments causes only a 9.6% drop—a 7.8%p gap confirming the model *does* rely on the target moment. On OOD, however, ground-truth perturbation (12.6%) and random perturbation (12.1%) cause nearly identical degradation, with only a 0.5%p gap—the model is effectively ignoring the visual content of the target moment and instead relying on dataset-specific shortcuts.

Object-centric representations reduce domain gap. The above findings motivate our approach: if the model fails because it relies on domain-specific visual patterns, decomposing the representation into object-centric slots should yield more transferable features. Fig. 2(d) validates this hypothesis. We compute a per-video representation by averaging the vision token hidden states and measure the Maximum Mean Discrepancy (MMD) [12] between source and target distributions (see Sec. 5.1 for details). The baseline exhibits an MMD of 0.192, while our slot-based representation reduces it to 0.097 (-49.6%), demonstrating that object-centric decomposition substantially narrows the domain gap.

4. SlotVTG

We introduce **SlotVTG**, a parameter-efficient framework that brings object-centric visual representation into pre-trained MLLMs at minimal cost. Fig. 3(a) provides an overview. We describe the Slot Adapter in Sec. 4.1, the Slot Alignment Loss in Sec. 4.2, and the training objective in Sec. 4.3.

4.1. Slot Adapter

Let $\mathbf{X} \in \mathbb{R}^{T\times N\times D}$ denote the visual tokens at a given decoder layer, where N is the number of tokens per frame and D is the hidden dimension. Instead of letting the LLM decoder process these tokens directly, we decompose them into a compact set of N_s abstract slots via iterative slot attention [32].

Down Projection. We first project the visual tokens $\mathbf{X}$ into a lower-dimensional bottleneck space:

$$\mathbf{X}_{down} = \mathbf{X}\mathbf{W}_{down} \in \mathbb{R}^{T\times N\times d} \tag{2}$$

where $\mathbf{W}_{down} \in \mathbb{R}^{D\times d}$ and $d \ll D$.

Slot Attention. A set of N_s learnable slot queries $\mathbf{S}^{(0)} \in \mathbb{R}^{T\times N_s\times d}$ attend to the projected tokens through I iterations. At each iteration, we project the slots and tokens into a common space with dimension d_h: $\mathbf{Q} = \mathbf{S}^{(i)}\mathbf{W}_Q \in \mathbb{R}^{T\times N_s\times d_h}$, $\mathbf{K} = \mathbf{X}_{down}\mathbf{W}_K \in \mathbb{R}^{T\times N\times d_h}$, and $\mathbf{V} = \mathbf{X}_{down}\mathbf{W}_V \in \mathbb{R}^{T\times N\times d_h}$, where $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ are the projection matrices. The attention scores are computed as:

$$\mathbf{M} = \mathbf{K}\mathbf{Q}^T/\sqrt{d_h} \in \mathbb{R}^{T\times N\times N_s} \tag{3}$$

We normalize $\mathbf{M}$ along the *slot axis* via softmax, fostering competitive assignment of tokens to slots:

$$A(n,k) = \frac{\exp(M(n,k))}{\sum_{j=1}^{N_s}\exp(M(n,j))} \tag{4}$$

We then normalize $\mathbf{A}$ along the token axis such that $\hat{A}(\cdot,k)$ sums to one:

$$\hat{A}(n,k) = \frac{A(n,k)}{\sum_{j=1}^{N}A(j,k)} \tag{5}$$

The updated slot representations are computed as a weighted mean aggregation $\mathbf{Z} = \hat{\mathbf{A}}^T\mathbf{V}$. Slots are updated via a Gated Recurrent Unit (GRU) [8] based recurrence. This competition mechanism encourages each slot to specialize in a distinct semantic entity within the frame.

Token Reconstruction. Since the LLM decoder expects the original token sequence length, we reconstruct the visual tokens from the slots via cross-attention, where the original tokens act as queries to retrieve entity-aware information from the final slots $\mathbf{S}^{(I)}$:

$$\hat{\mathbf{X}} = \text{CrossAttn}(\mathbf{X}_{down},\ \mathbf{S}^{(I)}) \in \mathbb{R}^{T\times N\times d} \tag{6}$$

The reconstructed tokens are projected back to the original dimension via an up projection. The adapter output is then added to the original tokens via a residual connection with a zero-initialized projection:

$$\mathbf{X}_{out} = \mathbf{X} + \hat{\mathbf{X}}\mathbf{W}_{up} \tag{7}$$

where $\mathbf{W}_{up} \in \mathbb{R}^{d\times D}$ projects back to the original dimension and is initialized to zero, so the adapter acts as an identity mapping at the start of training. This ensures training stability while gradually steering the representations toward entity-level decomposition.

Early-Layer Insertion. We attach the Slot Adapter only to the early decoder layers. Recent findings [20] show that cross-frame interactions occur in these early layers, while deeper layers handle language integration and answer generation. By inserting the Slot Adapter at this stage, each slot captures temporally coherent semantics across frames rather than frame-independent decompositions. The deeper layers, fine-tuned with LoRA [16, 51], then reason over these disentangled representations. Text tokens bypass the Slot Adapter throughout.

4.2. Slot Alignment Loss

While the Slot Adapter encourages decomposition through its bottleneck structure, the slots may form arbitrary clusters without additional guidance. We introduce Slot Alignment (SA) loss, which distills objectness priors from a

Table 1. **Performance comparison on video temporal grounding benchmarks.** We evaluate SlotVTG against state-of-the-art models on Charades-STA [11], QVHighlights [24], and ActivityNet Captions [22]. We report both In-Domain (ID) settings, where the source and target datasets are the same, and Out-of-Distribution (OOD) settings, where they differ. DETR-based methods (EATR [18] and CG-DETR [35]) are reproduced using pre-extracted CLIP [41] + SlowFast [10] features at 0.5 fps, following their original implementation. The performance of zero-shot VTG models is reported for reference. Our results are highlighted in green. The best results under the same cross-domain evaluation setting (source → target, LLM size) are highlighted in **bold**.

Source dataset	Method	LLM size	Target dataset											
			Charades-STA				ActivityNet-Captions				QVHighlights			
			R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU
Zero-Shot	HawkEye [45]	7B	50.6	31.4	14.5	33.7	49.1	29.3	10.7	32.7	-	-	-	-
	TimeSuite [55]	7B	69.9	48.7	24.0	-	-	16.6	9.3	22.0	-	12.3	9.2	21.3
	UniTime [29]	7B	-	59.1	31.9	52.2	-	22.8	14.1	27.3	-	41.0	31.5	43.7
	VideoMind [31]	2B	67.6	51.1	26.0	45.2	44.0	26.5	12.6	30.1	-	-	-	-
	VideoMind [31]	7B	73.5	59.1	31.2	50.2	48.4	30.3	15.7	33.3	-	-	-	-
Charades-STA	EaTR [18]	-	67.7	55.2	33.1	47.7	36.9	18.8	7.3	24.1	31.7	17.0	6.4	21.5
	CG-DETR [35]	-	69.7	57.6	35.1	49.5	32.6	16.8	6.8	22.1	37.4	22.8	10.5	25.2
	Chrono-BLIP [34]	4B	77.5	68.8	48.5	57.2	41.8	22.4	9.7	27.7	**66.6**	43.9	23.7	43.9
	Chrono-Qwen [34]	3B	77.2	63.4	40.3	55.2	44.4	26.3	13.1	30.1	63.3	43.6	23.3	42.7
	SlotVTG (Ours)	3B	77.2	64.0	41.2	55.4	**47.7**	**28.7**	**14.4**	**32.2**	66.0	**47.9**	**26.2**	**45.0**
	Chrono-Qwen [34]	7B	79.1	67.8	46.9	58.1	46.5	29.2	14.6	32.6	70.3	53.5	29.6	49.0
	SlotVTG (Ours)	7B	79.5	67.6	46.7	58.3	**52.0**	**33.2**	**16.7**	**35.5**	**74.0**	**57.6**	**32.2**	**51.3**
QVHighlights	EaTR [18]	-	40.8	27.2	13.0	28.0	36.7	20.9	9.7	25.3	70.3	59.6	40.3	53.1
	CG-DETR [35]	-	42.8	25.5	12.2	28.5	37.7	21.5	10.4	26.0	77.5	65.6	52.1	61.3
	Chrono-BLIP [34]	4B	61.5	37.0	19.8	41.4	41.8	22.4	9.7	27.7	86.1	76.8	62.8	70.8
	Chrono-Qwen [34]	3B	70.6	45.7	21.8	45.7	55.2	35.3	20.8	39.2	87.6	79.1	64.8	71.7
	SlotVTG (Ours)	3B	**70.7**	**46.6**	**22.6**	**46.1**	**56.1**	**35.7**	**21.1**	**40.0**	87.3	79.5	64.6	71.7
	Chrono-Qwen [34]	7B	75.2	53.3	27.4	49.9	60.7	41.4	24.8	43.4	90.7	81.8	67.6	74.9
	SlotVTG (Ours)	7B	**76.0**	**53.7**	**28.2**	**50.4**	**61.7**	**42.0**	**25.3**	**44.1**	91.3	82.9	69.3	76.0

self-supervised vision model (DINOv2 [38]) to encourage semantically coherent slot formation, as illustrated in Fig. 3(b).

Slot-based Similarity. Let $\mathbf{A} \in \mathbb{R}^{T\times N_s\times N}$ denote the slot attention weights from the final iteration. We transpose and L_2-normalize along the slot dimension, yielding $\bar{\mathbf{A}} \in \mathbb{R}^{T\times N\times N_s}$. Token-pair similarity under the slot assignments is computed and rescaled to $[-1, 1]$ to match the range of cosine similarity:

$$\mathbf{M}_{slot} = 2(\bar{\mathbf{A}}\bar{\mathbf{A}}^T) - 1 \in \mathbb{R}^{T\times N\times N} \tag{8}$$

DINO-based Similarity. We extract features from the last transformer block of a pre-trained DINOv2 [38] model and L_2-normalize them, yielding $\bar{\mathbf{F}}_{dino} \in \mathbb{R}^{T\times N\times d_{dino}}$. The target token-pair similarity is then computed as:

$$\mathbf{M}_{dino} = \bar{\mathbf{F}}_{dino}\bar{\mathbf{F}}_{dino}^T \in \mathbb{R}^{T\times N\times N} \tag{9}$$

Loss. The SA loss aligns these two structures:

$$\mathcal{L}_{SA} = 1 - \frac{1}{T}\sum_{t=1}^{T} \cos\left((\mathbf{M}_{slot}^{(t)}), (\mathbf{M}_{dino}^{(t)})\right) \tag{10}$$

4.3. Training Objective

The framework is trained end-to-end with the vision encoder frozen. The Slot Adapters and LoRA parameters are updated jointly. The total loss combines the standard autoregressive cross-entropy loss with the slot alignment regularization:

$$\mathcal{L}_{total} = \mathcal{L}_{CE} + \lambda\mathcal{L}_{SA} \tag{11}$$

where λ controls the strength of the objectness prior.

5. Experiments

5.1. Experimental Setup

Implementation Details. We build upon Qwen2.5-VL-Instruct [2] (3B and 7B) as the backbone MLLM, where the LLM decoder has a hidden dimension of D=2048 and D=3584, respectively. The vision encoder processes each 224×224 frame into N=64 visual tokens (8×8 spatial grid). For the Slot Adapter, we set the bottleneck dimension to d=512, the number of slots to K=4, and use 8 attention heads with I=3 iterations of GRU-based refinement. We inject the Slot Adapter into layers 1–7, while LoRA [16] (rank 16, α=64) is applied to the remaining deeper layers. The visual token hidden states used for the MMD analysis

in Sec. 3.2 are extracted from the last adapter layer (layer 7). The Slot Alignment loss uses DINOv2 [38]-base affinity matrices and is applied at the last layer where the Slot Adapter is inserted (layer 7) with λ=0.1. For video processing, we uniformly sample 20 and 60 frames for the models trained on Charades-STA [11] and QVHighlights [24], respectively. We train for 5 epochs with AdamW [33] (learning rate 5×10^{-5}) and a global batch size of 32 on 8 NVIDIA 3090/4090 GPUs. In total, the trainable parameters (Slot Adapters + LoRA) amount to approximately 7.6M (0.25% of the total) for the 3B model and 23.3M (0.33% of the total) for the 7B model.

Evaluation Protocol. We use Charades-STA (Cha.) [11] and QVHighlights (QVH.) [24] as source datasets for fine-tuning, and evaluate on three target datasets: Cha., QVH., and ActivityNet Captions (ANet) [22]. We denote each setting by its source-target pair (*e.g.*, Cha.→ANet). For each pair, we report both ID performance (source = target) and OOD performance (source $\neq$ target). All results are reported using standard moment retrieval metrics: R1@0.3, R1@0.5, R1@0.7, and mIoU.

Baselines. We compare against three categories of methods. (1) Zero-shot MLLMs that perform VTG without task-specific fine-tuning: HawkEye [45], TimeSuite [55], UniTime [29], and VideoMind [31]. (2) DETR-based specialists trained on a single source dataset: EaTR [18] and CG-DETR [35]. (3) MLLM-based methods fine-tuned on a single source dataset: Chrono [34] with both BLIP-2 [27] and Qwen2.5-VL-Instruct [2] (3B and 7B) backbones.

5.2. Results

Comparison with State-of-the-Art. Tab. 1 summarizes the results. SlotVTG consistently improves OOD performance across all source-target configurations while maintaining competitive ID performance.

When trained on Cha., SlotVTG (3B) achieves substantial OOD gains over the Chrono-Qwen [34] baseline: +2.4 R1@0.5 on ANet and +4.3 R1@0.5 on QVH., while preserving ID performance on Cha. (64.0 vs. 63.4 R1@0.5). This trend scales to the 7B model, where OOD improvements are even more pronounced (+4.0 R1@0.5 on ANet and +4.1 R1@0.5 on QVH.), demonstrating that SlotVTG benefits from larger model capacity.

When trained on QVH., SlotVTG (3B) again improves OOD generalization to both Cha. (+0.9 R1@0.5) and ANet (+0.4 R1@0.5) without sacrificing ID performance. SlotVTG (7B) also achieves OOD gains over the baseline: +0.4 R1@0.5 on Cha. and +0.6 R1@0.5 on ANet. The smaller OOD gains in this setting are expected, as QVH. is a more diverse dataset with broader domain coverage, leaving less room for improvement.

Notably, SlotVTG with a 3B backbone trained on Cha. already surpasses several zero-shot 7B models in OOD settings (*e.g.*, 28.7 vs. 30.3 R1@0.5 on ANet for SlotVTG 3B vs. VideoMind [31] 7B), despite being fine-tuned on a single source dataset. Compared to DETR-based specialists (EaTR [18], CG-DETR [35]), SlotVTG achieves significantly better OOD performance across all settings. These results highlight that object-centric decomposition enables the model to genuinely ground in visual content rather than relying on dataset-specific patterns, resulting in robust generalization across domains.

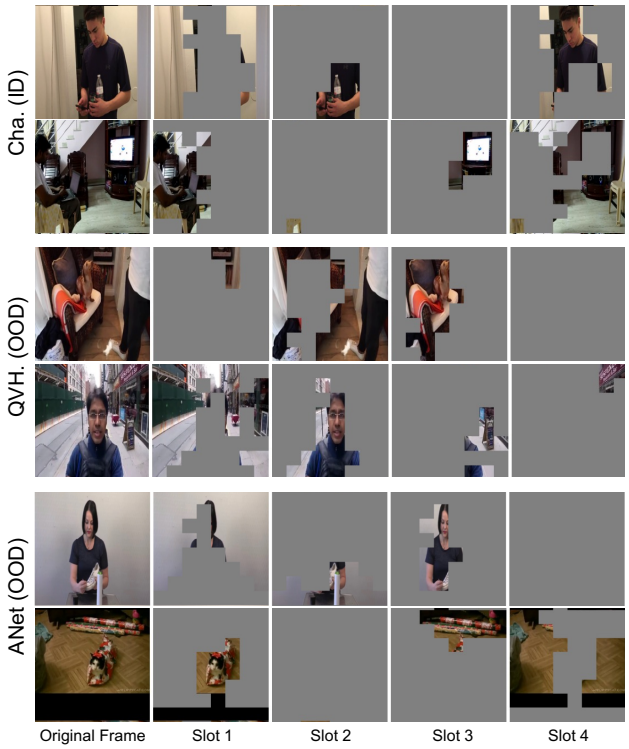

Figure 4. **Slot attention visualization.** We visualize the slot assignments on samples from Cha. (ID), QVH. (OOD), and ANet (OOD) by masking each frame with its highest-attending slot. Each column corresponds to one of the four learned slots.

What Do Slots Learn? We visualize the slot attention maps in Fig. 4 by masking each frame region with its highest-attending slot. Across both ID and OOD samples, the slots decompose scenes into semantically coherent regions such as people, objects, and backgrounds, though the specific slot-to-entity mapping varies across frames. Importantly, this decomposition generalizes to unseen domains (QVH., ANet) without any domain-specific supervision, confirming that the Slot Adapter learns transferable entity-level representations rather than dataset-specific patterns.

5.3. Ablation Study

We conduct extensive ablation studies to verify the effectiveness of each component in SlotVTG (Tab. 2). Unless

Table 2. **Ablation study.** To validate the effect of each component in SlotVTG, we show the results on the Cha.→ANet setting. 'Cha.' and 'ANet' denote Charades-STA [11] and ActivityNet Captions [22]. We use Qwen2.5-VL [2] 3B as a backbone MLLM. We report R1@0.5 and R1@0.7 scores on both ID and OOD settings. The best numbers are **highlighted**.

(a) Effects of Slot Adapter.

Adapter Type	Cha. (ID)		ANet (OOD)	
	R1@0.5	R1@0.7	R1@0.5	R1@0.7
LoRA [16]	63.4	40.3	26.3	13.1
Adapter w/ self attention	63.5	40.9	26.5	13.7
Slot Adapter	**64.0**	**41.2**	**28.7**	**14.4**

(b) Effects of SA loss.

$\mathcal{L}_{SA}$	Loss scale λ	Cha. (ID)		ANet (OOD)	
		R1@0.5	R1@0.7	R1@0.5	R1@0.7
×		63.3	41.0	28.0	14.0
✓	0.1	64.0	41.2	**28.7**	**14.4**
✓	0.2	**64.3**	**42.2**	26.1	13.0

(c) Effects of Slot Adapter insertion layers.

Layer index l	Cha. (ID)		ANet (OOD)	
	R1@0.5	R1@0.7	R1@0.5	R1@0.7
1-7	63.3	41.0	**28.7**	**14.4**
10-17	63.3	41.0	27.5	14.0
20-36	63.3	39.5	28.4	14.0

(d) Effects of number of slots and bottleneck dimensions.

# of slots N_s	Dimension d	Cha. (ID)		ANet (OOD)	
		R1@0.5	R1@0.7	R1@0.5	R1@0.7
4	128	64.1	41.7	28.5	14.1
4	512	63.3	41.0	**28.7**	**14.4**
8	512	63.7	39.8	28.6	14.5

(e) Effects of token reconstruction design.

Method	Cha. (ID)		ANet (OOD)	
	R1@0.5	R1@0.7	R1@0.5	R1@0.7
Repeat & Proj.	**63.6**	40.8	28.2	13.7
Cross attention	63.3	**41.0**	**29.3**	**14.9**

(f) Effects of SA loss placement.

Layer index l	Cha. (ID)		ANet (OOD)	
	R1@0.5	R1@0.7	R1@0.5	R1@0.7
1-7 (all adapter layers)	**64.0**	**41.5**	28.5	14.3
7 (last adapter layer)	**64.0**	41.2	**29.3**	**14.9**

otherwise stated, we use Qwen2.5-VL-3B [2] as the backbone and train on Cha., reporting both ID (Cha.) and OOD (ANet) performance in R1@0.5 and R1@0.7.

Effects of Slot Adapter. We compare our Slot Adapter against two baselines: LoRA-only fine-tuning and an adapter with standard self-attention instead of slot attention (Tab. 2a). While all three achieve comparable ID performance, the Slot Adapter yields the best OOD performance, confirming that the competitive slot decomposition mechanism is key to improving generalization.

Effects of SA Loss. Removing the SA loss noticeably degrades out-of-distribution (OOD) performance (28.0 vs. 28.7 in R1@0.5), as shown in Tab. 2b. While increasing λ to 0.2 improves in-distribution (ID) R1@0.5 performance to 64.3, it harms OOD performance, dropping it to 26.1. This trade-off suggests that enforcing an excessively strong objectness prior may lead to overfitting on source-domain patterns. Therefore, we set $\lambda = 0.1$ as our default value.

Effects of Slot Adapter Insertion Layers. Integrating the Slot Adapter into the early layers (1–7) yields the best out-of-distribution (OOD) performance, as shown in Tab. 2c. This aligns with the findings of [20], which demonstrate that cross-frame interactions predominantly occur in the early decoder layers. Conversely, applying the adapter to the middle (10–17) or later (20–36) layers degrades OOD performance, indicating that late-stage interventions likely introduce unnecessary noise.

Effects of Number of Slots and Bottleneck Dimensions. We vary the number of slots N_s and bottleneck dimension d (Tab. 2d). Using N_s=4 and d=512 achieves the best OOD performance. A smaller dimension (d=128) slightly improves ID but degrades OOD, while increasing to N_s=8 slots hurts ID, likely because excessive slots dilute the decomposition.

Effects of Token Reconstruction Design. We compare two strategies for reconstructing the original token sequence from the N_s slots (Tab. 2e): (1) repeating each slot N/N_s times followed by a linear projection, and (2) cross-attention where original tokens query the slots. Cross-attention achieves better OOD performance (29.3 vs. 28.2 R1@0.5), as it allows each token to selectively retrieve entity-aware information from its most relevant slot rather than receiving a uniform representation.

Effects of SA Loss Placement. Applying the SA loss only at the last adapter layer (layer 7) outperforms applying it across all adapter layers (1–7) in OOD (Tab. 2f). While full-layer constraints force premature alignment, applying them only to the last layer allows earlier layers to learn more flexible representations.

6. Conclusion

We presented SlotVTG, a parameter-efficient framework that introduces object-centric decomposition into pre-trained MLLMs for generalizable Video Temporal Grounding. Our failure analysis reveals that naïvely fine-tuned MLLMs exploit dataset-specific shortcuts rather than grounding in visual content. SlotVTG addresses this via a lightweight Slot Adapter that decomposes visual tokens into entity-level slots in the early decoder layers, guided by a Slot Alignment Loss that distills objectness priors. Extensive experiments demonstrate that SlotVTG consistently improves OOD generalization while maintaining competitive ID performance.

Acknowledgment. This work was supported in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea Government (MSIT) (RS-2024-00353131, 50%) and the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT) (RS-2025-22362968, 50%).

References

[1] Kyungho Bae, Geo Ahn, Youngrae Kim, and Jinwoo Choi. Devias: Learning disentangled video representations of action and scene. In *ECCV*, 2024. 3

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 3, 6, 7, 8

[3] Peijun Bao and Yadong Mu. Learning sample importance for cross-scenario video temporal grounding. *Proceedings of the 2022 International Conference on Multimedia Retrieval (ICMR)*, 2022. 2

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 3

[6] Jinyeong Chae, Donghwa Kim, Kwanseok Kim, Doyeon Lee, Sangho Lee, Seongsu Ha, Jonghwan Mun, Wooyoung Kang, Byungseok Roh, and Joonseok Lee. Towards a complete benchmark on video moment localization. In *International Conference on Artificial Intelligence and Statistics*, pages 4168–4176. PMLR, 2024. 2, 3

[7] Donghwan Chi, Hyomin Kim, Yoonjin Oh, Yongjin Kim, Donghoon Lee, Daejin Jo, Jongmin Kim, Junyeob Baek, Sungjin Ahn, and Sungwoong Kim. Slot-mllm: Object-centric visual tokenization for multimodal llm. *arXiv preprint arXiv:2505.17726*, 2025. 2, 3

[8] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *EMNLP*, 2014. 5

[9] Jinwoo Choi, Chen Gao, Joseph CE Messou, and Jia-Bin Huang. Why can't i dance in the mall? learning to mitigate scene bias in action recognition. *NIPS*, 2019. 3

[10] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019. 6

[11] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, 2017. 2, 3, 4, 6, 7, 8

[12] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The journal of machine learning research*, 13(1):723–773, 2012. 3, 5

[13] Yongxin Guo, Jingyu Liu, Mingda Li, Xiaoying Tang, Qingbin Liu, and Xi Chen. Trace: Temporal grounding video llm via causal event modeling. *arXiv preprint arXiv:2410.05643*, 2024. 1, 2

[14] Jiachang Hao, Haifeng Sun, Pengfei Ren, Jingyu Wang, Qi Qi, and Jianxin Liao. Can shuffling video benefit temporal bias problem: A novel training framework for temporal grounding. In *ECCV*, 2022. 2, 3

[15] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017. 2

[16] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022. 5, 6, 8

[17] Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. Vtimellm: Empower llm to grasp video moments. In *CVPR*, 2024. 1, 2

[18] Jinhyun Jang, Jungin Park, Jin Kim, Hyeongjun Kwon, and Kwanghoon Sohn. Knowing where to focus: Event-aware transformer for video grounding. In *ICCV*, 2023. 2, 6, 7

[19] Hao Jiang, Yang Yizhang, and Yadong Mu. Transferable video moment localization by moment-guided query prompting. In *AAAI*, 2024. 2

[20] Minji Kim, Taekyung Kim, and Bohyung Han. Map the flow: Revealing hidden pathways of information in video-ollms. *arXiv preprint arXiv:2510.13251*, 2025. 5, 8

[21] Thomas Kipf, Gamaleldin F Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. *arXiv preprint arXiv:2111.12594*, 2021. 3

[22] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, 2017. 6, 7, 8

[23] Xiaohan Lan, Yitian Yuan, Hong Chen, Xin Wang, Zequn Jie, Lin Ma, Zhi Wang, and Wenwu Zhu. Curriculum multi-negative augmentation for debiased video grounding. In *AAAI*, 2023. 3

[24] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. In *NIPS*, 2021. 1, 2, 3, 4, 6, 7

[25] Jie Lei, Tamara Berg, and Mohit Bansal. Revealing single frame bias for video-and-language learning. In *ACL*, 2023. 3

[26] Juncheng Li, Junlin Xie, Long Qian, Linchao Zhu, Siliang Tang, Fei Wu, Yi Yang, Yueting Zhuang, and Xin Eric Wang. Compositional temporal grounding with structured variational cross-graph correspondence learning. In *CVPR*, 2022. 2

[27] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 7

[28] Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards action recognition without representation bias. In *ECCV*, 2018. 3

[29] Zeqian Li, Shangzhe Di, Zhonghua Zhai, Weilin Huang, Yanfeng Wang, and Weidi Xie. Universal video temporal grounding with generative multi-modal large language models. In *NIPS*, 2025. 6, 7

[30] Kevin Qinghong Lin, Pengchuan Zhang, Joya Chen, Shraman Pramanick, Difei Gao, Alex Jinpeng Wang, Rui Yan, and Mike Zheng Shou. Univtg: Towards unified video-language temporal grounding. In *ICCV*, 2023. 1, 2

[31] Ye Liu, Kevin Qinghong Lin, Chang Wen Chen, and Mike Zheng Shou. Videomind: A chain-of-lora agent for temporal-grounded video reasoning. In *ICLR*, 2026. 2, 6, 7

[32] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *NIPS*, 2020. 2, 3, 5

[33] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 7

[34] Boris Meinardus, Hector Rodriguez, Anil Batra, Anna Rohrbach, and Marcus Rohrbach. Chrono: A simple blueprint for representing time in mllms. *arXiv preprint arXiv:2406.18113*, 2024. 1, 2, 4, 6, 7

[35] WonJun Moon, Sangeek Hyun, SuBeen Lee, and Jae-Pil Heo. Correlation-guided query-dependency calibration for video temporal grounding. *arXiv preprint arXiv:2311.08835*, 2023. 2, 6, 7

[36] WonJun Moon, Sangeek Hyun, SangUk Park, Dongchan Park, and Jae-Pil Heo. Query-dependent video representation for moment retrieval and highlight detection. In *CVPR*, 2023. 1, 2

[37] Guoshun Nan, Rui Qiao, Yao Xiao, Jun Liu, Sicong Leng, Hao Zhang, and Wei Lu. Interventional video grounding with dual contrastive learning. In *CVPR*, 2021. 3

[38] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *arXiv:2304.07193*, 2023. 2, 6, 7

[39] Mayu Otani, Yuta Nakashima, Esa Rahtu, and Janne Heikkilä. Uncovering hidden challenges in query-based video moment retrieval. *arXiv preprint arXiv:2009.00325*, 2020. 2, 3

[40] Zhaobo Qi, Yibo Yuan, Xiaowen Ruan, ShuHui Wang, Weigang Zhang, and QingMing Huan. Bias-conflict sample synthesis and adversarial removal debias strategy for temporal sentence grounding in video. In *AAAI*, 2024. 2, 3

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 6

[42] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *CVPR*, 2024. 1, 2

[43] Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, et al. Bridging the gap to real-world object-centric learning. *arXiv preprint arXiv:2209.14860*, 2022. 3

[44] Hao Sun, Mingyao Zhou, Wenjing Chen, and Wei Xie. Tr-detr: Task-reciprocal transformer for joint moment retrieval and highlight detection. In *AAAI*, 2024. 1, 2

[45] Yueqian Wang, Xiaojun Meng, Jianxin Liang, Yuxuan Wang, Qun Liu, and Dongyan Zhao. Hawkeye: Training video-text llms for grounding text in videos. *arXiv preprint arXiv:2403.10228*, 2024. 1, 2, 6, 7

[46] Ye Wang, Ziheng Wang, Boshen Xu, Yang Du, Kejun Lin, Zihan Xiao, Zihao Yue, Jianzhong Ju, Liang Zhang, Dingyi Yang, Xiangnan Fang, Zewen He, Zhenbo Luo, Wenxuan Wang, Junqi Lin, Jian Luan, and Qin Jin. Time-r1: Post-training large vision language model for temporal video grounding. In *NIPS*, 2025. 2

[47] Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models. *arXiv preprint arXiv:2210.05861*, 2022. 3

[48] Yicheng Xiao, Zhuoyan Luo, Yong Liu, Yue Ma, Hengwei Bian, Yatai Ji, Yujiu Yang, and Xiu Li. Bridging the gap: A unified video comprehension framework for moment retrieval and highlight detection. In *CVPR*, 2024. 2

[49] Jiaqi Xu, Cuiling Lan, Wenxuan Xie, Xuejin Chen, and Yan Lu. Slot-vlm: Object-event slots for video-language modeling. In *NIPS*, 2024. 2, 3

[50] Jin Yang, Ping Wei, Huan Li, and Ziyang Ren. Task-driven exploration: Decoupling and inter-task feedback for joint moment retrieval and highlight detection. In *CVPR*, 2024. 2

[51] Taojiannan Yang, Yi Zhu, Yusheng Xie, Aston Zhang, Chen Chen, and Mu Li. Aim: Adapting image models for efficient video understanding. In *ICLR*, 2023. 5

[52] Xun Yang, Fuli Feng, Wei Ji, Meng Wang, and Tat-Seng Chua. Deconfounded video moment retrieval with causal intervention. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 1–10, 2021. 3

[53] Yitian Yuan, Xiaohan Lan, Xin Wang, Long Chen, Zhi Wang, and Wenwu Zhu. A closer look at temporal sentence grounding in videos: Dataset and metric. In *Proceedings of the 2nd international workshop on human-centric multimedia analysis*, pages 13–21, 2021. 3

[54] Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. Dense regression network for video grounding. In *CVPR*, 2020. 2

[55] Xiangyu Zeng, Kunchang Li, Chenting Wang, Xinhao Li, Tianxiang Jiang, Ziang Yan, Songze Li, Yansong Shi, Zhengrong Yue, Yi Wang, Yali Wang, Yu Qiao, and Limin Wang. Timesuite: Improving MLLMs for long video understanding via grounded tuning. In *ICLR*, 2025. 1, 2, 6, 7

[56] Jun Zhang, Teng Wang, Yuying Ge, Yixiao Ge, Xinhao Li, Ying Shan, and Limin Wang. Timelens: Rethinking video temporal grounding with multimodal llms. *arXiv preprint arXiv:2512.14698*, 2025. 4

[57] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. Learning 2d temporal adjacent networks for moment localization with natural language. In *AAAI*, 2020. 2