COVERVTG: COVERAGE-CONSTRAINED SPARSE OBSERVATION FOR UNIVERSAL VIDEO TEMPORAL GROUNDING

Anonymous authors
Paper under double-blind review

ABSTRACT

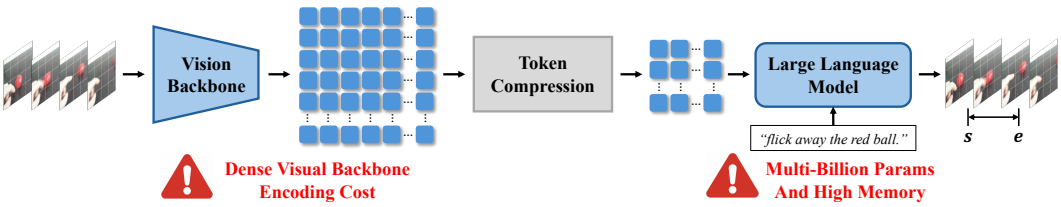
Universal Video Temporal Grounding (VTG) aims to localize diverse natural-language queries in videos spanning different domains, viewpoints, and durations. Recent approaches largely rely on multimodal large language models (MLLMs), whose large parameter scale and dense visual encoding cost make them expensive for long videos and difficult to deploy in resource-constrained scenarios. In this work, we explore a lightweight backbone-centric alternative and propose **CoverVTG**, a coverage-constrained framework that directly reduces the number of visual tokens requiring expensive backbone encoding. Since shallow patch embeddings lack reliable high-level semantics, CoverVTG formulates pre-backbone sparsification as a *structured sparse observation* problem rather than semantic token selection. Within each local spatio-temporal tube, a sparse set of tokens is selected under explicit spatial and temporal coverage constraints, so that discarded observations remain locally observable from complementary retained tokens. We formalize this property through a **coverage–observability–recoverability** connection and establish a bound on local recovery error under spatio-temporal smoothness. Lightweight tube-local recovery adapters inserted throughout a frozen visual backbone progressively exploit such complementary observations as semantic representations emerge. Extensive experiments show that CoverVTG achieves competitive grounding accuracy with substantially fewer parameters and lower inference latency than MLLM-based approaches.

1 INTRODUCTION

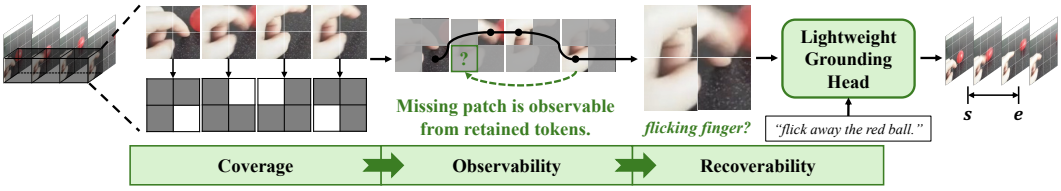
Video temporal grounding (VTG) aims to localize temporal moments in an untrimmed video according to a natural-language query. Beyond conventional dataset-specific settings (Gao et al., 2017; Zhang et al., 2020a; Lei et al., 2021; Hannan et al., 2024; Feng et al., 2025), Universal VTG requires a unified model to handle videos with diverse viewpoints, domains, and durations, together with heterogeneous free-form queries (Li et al., 2026). Recent approaches largely address this challenge with multimodal large language models (MLLMs) (Huang et al., 2024; Zeng et al., 2025; Guo et al., 2025; Liu et al., 2026), leveraging their strong vision-language understanding and generalization. These advances demonstrate the potential of MLLMs for Universal VTG, yet scalable processing remains particularly challenging for long videos containing hundreds or thousands of frames.

Despite their strong generalization ability, MLLM-based solutions remain computationally expensive and difficult to deploy in resource-constrained scenarios. Their large parameter scale incurs substantial memory and computation overhead during both training and inference. Meanwhile, existing efficiency designs (Li et al., 2026) are largely concerned with controlling the amount of visual information consumed by the downstream language model. Such strategies effectively reduce multimodal context cost, but another important source of computation remains: *dense visual backbone encoding*. Before visual representations can be compressed or consumed by the downstream model, the visual backbone must extract semantic features from a large number of spatio-temporal observations. This motivates a complementary question for Universal VTG: *can we directly reduce the number of visual observations that require expensive semantic encoding, while retaining sufficient evidence for accurate temporal grounding?*

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) A representative MLLM-based pipeline



(b) Our lightweight backbone-centric coverage-constrained framework

Figure 1: **Motivation of CoverVTG.** (a) Representative MLLM-based VTG methods densely encode visual observations before token compression and downstream language modeling, resulting in substantial visual encoding and model overhead. (b) CoverVTG instead performs coverage-constrained sparse observation before expensive semantic encoding. Spatial-temporal coverage keeps discarded observations locally observable from retained tokens, enabling subsequent semantic recovery and lightweight temporal grounding.

Reducing backbone computation requires token sparsification to occur *before* expensive semantic encoding, which introduces a fundamental difficulty. A natural strategy is to retain only semantically important or query-relevant tokens. However, before passing through a deep visual backbone, tokens are represented only by shallow patch embeddings and do not yet contain reliable high-level semantics. Estimating semantic importance or visual-text relevance at this stage would therefore require either the expensive backbone itself or an additional semantic encoder, largely offsetting the benefit of pre-backbone sparsification. We are thus faced with a key challenge: *how can we aggressively reduce backbone tokens when reliable semantic importance is not yet available?*

Our answer is to shift from semantic token selection to *structured sparse observation*. Our key observation is that natural videos exhibit strong local redundancy across both space and time (Tong et al., 2022). Within a sufficiently local spatio-temporal neighborhood, a spatial region omitted in one frame can often be complemented by another retained observation at the same time step or by the same region observed in a neighboring frame. Based on this property, we propose **CoverVTG**, a coverage-constrained pre-backbone sparsification framework for Universal VTG. Within each local spatio-temporal tube, CoverVTG retains only a small subset of patch tokens while imposing explicit spatial and temporal coverage constraints. These constraints prevent sparse observations from collapsing onto only a few frames or spatial regions.

For every discarded observation, the coverage constraints guarantee complementary retained observations along both spatial and temporal dimensions. Under local spatio-temporal smoothness, the distance from a missing observation to its retained spatial or temporal anchors yields a bound on local recovery error. This provides a principled explanation for why aggressive sparsification can remain viable even before high-level semantic information is available. To exploit this recoverability during semantic encoding, we insert lightweight tube-local spatio-temporal recovery adapters throughout a frozen visual backbone. Instead of explicitly reconstructing all discarded tokens, the adapters progressively exchange complementary information among retained observations as their representations become increasingly semantic. In this way, CoverVTG combines *structure-aware observation before semantics emerge* with *progressive local recovery as semantics emerge*.

Extensive experiments on four VTG benchmarks demonstrate that CoverVTG provides a favorable accuracy–efficiency trade-off. With substantially fewer parameters than large MLLM-based approaches, CoverVTG attains comparable zero-shot performance while reducing inference latency by more than an order of magnitude. After task-specific fine-tuning, it further achieves strong performance across both long- and short-video benchmarks, validating the effectiveness of our coverage-constrained sparse observation and progressive recovery design.

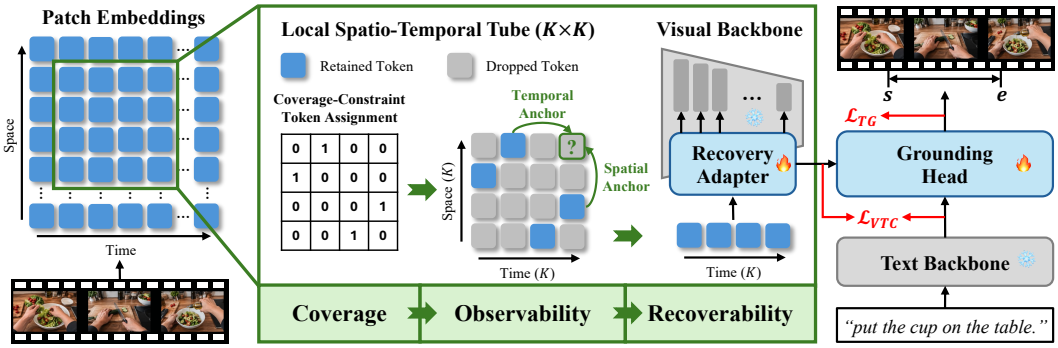


Figure 2: **The overall framework of CoverVTG**. CoverVTG partitions shallow patch embeddings into local spatio-temporal tubes and performs Coverage-Constrained Token Assignment (CCTA), retaining only a small subset of tokens while explicitly preserving complementary spatial and temporal observations. The resulting coverage makes discarded observations locally observable from retained spatial and temporal anchors. Lightweight Recovery Adapters inserted throughout the frozen visual backbone progressively exploit these observations as semantic representations emerge, providing recoverable visual features for downstream temporal grounding.

2 METHOD

2.1 FRAMEWORK OVERVIEW

Given an untrimmed video $\mathcal{V} = \{f_t\}_{t=1}^{T}$ and a natural-language query Q, video temporal grounding aims to predict the temporal segments $Y = \{(s_i, e_i)\}_{i=1}^{N_y}$ that correspond to the query. Universal VTG further requires a unified model to handle videos with substantially different domains, viewpoints, and durations.

Our goal is to reduce the computational cost of visual semantic encoding. As shown in Fig. 2, a patch embedding layer first converts the input video into shallow visual tokens $X \in \mathbb{R}^{T\times H_p\times W_p\times d}$, where $H_p \times W_p$ denotes the spatial patch grid. Let $N_{\text{tok}} = TH_pW_p$ denote the total number of patch tokens. When N_{tok} exceeds a prescribed visual-token budget N_{budget}, CoverVTG sparsifies the patch tokens *before* they enter the visual backbone. Rather than estimating semantic importance from shallow features, we construct a structured sparse observation pattern that explicitly preserves local spatial and temporal coverage. The sparse observations are subsequently processed by the visual backbone. To progressively exploit the complementary observations induced by coverage, lightweight spatio-temporal recovery adapters are inserted after every backbone block.

We next describe how the sparse observations are constructed, why coverage provides local recoverability, and how the recovery adapters exploit this property during semantic encoding.

2.2 COVERAGE-CONSTRAINED TOKEN ASSIGNMENT

A central difficulty of pre-backbone sparsification is that reliable high-level semantic importance is not yet available when the token-selection decision must be made. We therefore avoid query-dependent semantic pruning and instead formulate token reduction as a *structured sparse observation* problem.

Local spatio-temporal tubes. Consider an $l \times l$ spatial window containing $K = l^2$ spatial positions. We associate the same local window across K neighboring frames, forming a local spatio-temporal tube $\mathcal{T} = \{x_{t,p}\}_{t,p=1}^{K}$, where t denotes the local temporal index and p denotes the spatial position. The tube therefore contains K^2 candidate tokens.

Shallow token informativeness. Since token selection is performed before semantic backbone encoding, we avoid estimating semantic importance. Instead, we characterize each patch using appearance variation that can be directly computed from shallow patch embeddings.

Let $x_{t,p} \in \mathbb{R}^d$ denote the patch embedding at temporal index t and spatial position p within a local spatio-temporal tube, and $\hat{x}_{t,p}$ denote its ℓ_2-normalized representation. We then measure two complementary shallow visual cues:

$$\begin{aligned} S_{t,p}^{\text{local}} &= 1 - \cos\left(\hat{x}_{t,p}, \bar{x}_t\right), \\ S_{t,p}^{\text{temporal}} &= 1 - \cos\left(\hat{x}_{t,p}, \bar{x}_{t-1}\right), \end{aligned} \tag{1}$$

where $\bar{x}_t$ represents the average visual content of the tokens at frame t within the local spatio-temporal tube. $S_{t,p}^{\text{local}}$ measures how visually distinctive token (t,p) is relative to the mean content of the t-th temporal slice of the local tube, whereas $S_{t,p}^{\text{temporal}}$ measures its appearance variation relative to the mean content of the preceding temporal slice.

The final shallow informativeness score is

$$S_{t,p} = S_{t,p}^{\text{local}} + S_{t,p}^{\text{temporal}}. \tag{2}$$

Coverage-constrained assignment. We introduce a binary matrix $M \in \{0,1\}^{K\times K}$, where $M_{t,p} = 1$ indicates that observation (t,p) is retained.

We impose

$$\begin{aligned} \sum_{p=1}^{K} M_{t,p} &= 1, \qquad \forall t, \\ \sum_{t=1}^{K} M_{t,p} &= 1, \qquad \forall p, \end{aligned} \tag{3}$$

such that every temporal step contributes one retained observation, and every spatial position is observed once across the local temporal tube.

Among all coverage-valid assignments, we maximize the total shallow informativeness:

$$M^\star = \arg\max_{M\in\mathcal{P}_K} \sum_{t=1}^{K}\sum_{p=1}^{K} M_{t,p} S_{t,p}, \tag{4}$$

where $\mathcal{P}_K = \left\{M \in \{0,1\}^{K\times K} \mid M\mathbf{1} = \mathbf{1},\ M^\top \mathbf{1} = \mathbf{1}\right\}$.

Every feasible matrix in $\mathcal{P}_K$ is a permutation matrix. Let $\pi(t)$ denote the selected spatial location at temporal step t. Eq. 4 can equivalently be written as

$$\pi^\star = \arg\max_{\pi\in\mathfrak{S}_K} \sum_{t=1}^{K} S_{t,\pi(t)}, \tag{5}$$

where $\mathfrak{S}_K$ denotes all permutations of K elements.

The resulting sparse observation pattern distributes fine-grained spatial observations across neighboring frames instead of uniformly degrading the spatial resolution of every frame. For each input video, we adaptively choose the smallest local window size L such that the number of sparsified tokens does not exceed the visual-token budget.

$$l = \min\left\{ L \in \{1,2,3,\ldots\} \;\middle|\; \frac{N_{\text{tok}}}{L^2} \le N_{\text{budget}} \right\}. \tag{6}$$

2.3 Coverage-Induced Observability and Recoverability

The coverage constraint is designed not merely to diversify selected tokens, but to control whether missing observations remain accessible from retained ones. We next formalize this property and show how coverage induces local recoverability.

Dual-Axis Local Observability. Because $M^\star$ is a permutation matrix, every temporal position t retains exactly one spatial observation $x_{t,\pi(t)}$. Similarly, every spatial position p is retained at exactly one temporal position $\pi^{-1}(p)$.

Consider an arbitrary discarded token $x_{t,p}$, where $p \neq \pi(t)$. The coverage constraints guarantee the existence of two complementary retained observations:

$$\begin{aligned} a^s_{t,p} &= x_{t,\pi(t)}, \\ a^t_{t,p} &= x_{\pi^{-1}(p),p}. \end{aligned} \tag{7}$$

We refer to them as the *spatial anchor* and *temporal anchor*, respectively. The spatial anchor is observed at the same time step but at another spatial position, while the temporal anchor observes the same spatial position at a neighboring time step.

Local Spatio-Temporal Smoothness. Our analysis is based on the local redundancy of natural videos (Tong et al., 2022). Let $z_{t,p}$ denote the latent semantic representation associated with position (t, p). We assume that within a sufficiently local spatio-temporal neighborhood, the latent representation varies smoothly in both space and time:

$$\begin{aligned} \|z_{t,p} - z_{t,p'}\| &\leq L_s \, d_s(p, p'), \\ \|z_{t,p} - z_{t',p}\| &\leq L_t \, |t - t'|, \end{aligned} \tag{8}$$

where L_s and L_t denote local spatial and temporal smoothness constants.

This assumption does not require videos to be globally smooth. Instead, it captures the local redundancy exploited by CoverVTG: visual information omitted at one spatial-temporal location may remain observable from nearby locations or neighboring frames. The coverage-induced anchors immediately provide a simple recovery guarantee.

Proposition 2.1. *For any discarded representation $z_{t,p}$ satisfying Eqs. 8, there exists an anchor-based estimator $\hat{z}_{t,p}$ such that:*

$$\|z_{t,p} - \hat{z}_{t,p}\| \leq \min \left\{ L_s d_s(p, \pi(t)) \, , L_t \left| t - \pi^{-1}(p) \right| \right\}. \tag{9}$$

Proof sketch. From the spatial-temporal smoothness assumption and Eq. 7,

$$\begin{aligned} \|z_{t,p} - z_{t,\pi(t)}\| &\leq L_s d_s(p, \pi(t)) \, , \\ \|z_{t,p} - z_{\pi^{-1}(p),p}\| &\leq L_t \left| t - \pi^{-1}(p) \right| . \end{aligned} \tag{10}$$

Selecting the anchor with the smaller upper bound yields Eq. 9.

Eq. 9 shows that coverage turns aggressive sparsification into a locally bounded observation problem: every missing observation remains close to at least one complementary retained observation along either space or time. This establishes the central relation:

$$\boxed{\text{Coverage} \Longrightarrow \text{Observability} \Longrightarrow \text{Recoverability}}. \tag{11}$$

We emphasize that Pro. 2.1 establishes the existence of informative local anchors rather than requiring the model to explicitly reconstruct every discarded token. In practice, we exploit this recoverability by allowing retained observations to exchange complementary information through the lightweight recovery module described next.

2.4 Spatio-Temporal Recovery Adapter

Coverage-Constrained Token Assignment preserves only K observations from each local spatio-temporal tube. To progressively aggregate their complementary information during semantic encoding, we insert a lightweight **Spatio-Temporal Recovery Adapter** after every backbone block.

Let $Z^n_m \in \mathbb{R}^{K \times d}$ denote the output features of the m-th local tube from the n-th backbone block, where d is the backbone channel dimension. The recovery adapter follows a lightweight bottleneck design with tube-local attention:

$$\begin{aligned} U^n_m &= \mathrm{LN}(Z^n_m) W^n_{\mathrm{down}}, \qquad W^n_{\mathrm{down}} \in \mathbb{R}^{d \times d_a}, \\ \widetilde{U}^n_m &= \mathrm{Attn}\left(U^n_m\right), \\ H^n_m &= Z^n_m + \widetilde{U}^l_m W^n_{\mathrm{up}}, \qquad W^n_{\mathrm{up}} \in \mathbb{R}^{d_a \times d}, \end{aligned} \tag{12}$$

where $d_a \ll d$. We adopt an attention-based adapter because the number of retained observations varies with the adaptive local window size described in Eq. 6. Self-attention (Vaswani et al., 2017) naturally operates on variable-length token sets with shared parameters, allowing the same recovery module to accommodate different sparsification granularities. In contrast, convolutional adapters tied to a predefined local grid or receptive field require additional architectural handling when the local tube size changes.

2.5 Training Objectives

CoverVTG uses two complementary training objectives: a temporal grounding loss $\mathcal{L}_{\mathrm{TG}}$ for fine-grained temporal localization, and a video-text contrastive loss $\mathcal{L}_{\mathrm{VTC}}$ for learning query-discriminative visual representations.

Temporal Grounding Loss $\mathcal{L}_{\mathrm{TG}}$. We adopt HieraMamba (An & Grauman, 2026) as the temporal grounding head. Its hierarchical state-space design provides linear-time temporal modeling and is therefore well suited to long-video grounding, while preserving fine-grained temporal structure across multiple scales. We follow its standard multi-task objective, combining classification and boundary regression losses.

Video-Text Contrastive Loss $\mathcal{L}_{\mathrm{VTC}}$. Given the output features of the visual backbone, we partition the temporal sequence into a set of clips $\{v_i\}_{i=1}^{N_c}$, where each clip contains $T_{\mathrm{retrieval}}$ frames. We then employ a symmetric contrastive loss (Bain et al., 2021) to encourage strong mutual information between matched visual clips and text queries.

The overall training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{TG}} + \mathcal{L}_{\mathrm{VTC}}. \tag{13}$$

2.6 Universal VTG Inference

To accommodate videos of different durations under a fixed visual-token budget, we define the maximum number of frames that can be densely encoded as

$$T_{\mathrm{dense}} = \frac{N_{\mathrm{budget}}}{H_p W_p}. \tag{14}$$

Coarse-Grained Video Retrieval. For an input video with $T > T_{\mathrm{dense}}$, the complete video is first sparsified by Coverage-Constrained Token Assignment and subsequently encoded by the visual backbone. The resulting visual features are divided into a sequence of contiguous clips $\{\mathcal{C}_i\}_{i=1}^{N_c}$, where each clip contains $T_{\mathrm{retrieval}}$ frames. For each clip, we obtain a visual representation v_i and compute its cosine similarity with the query representation q and select the most relevant clip. We construct a temporal window centered at the selected clip and expand it to T_{dense} frames for fine-grained temporal grounding.

Fine-Grained Temporal Grounding. For an input sequence with $T \leq T_{\mathrm{dense}}$, including either a short video or a retrieved temporal window, all visual tokens are preserved without sparsification. The input is densely encoded by the visual backbone and subsequently processed by the temporal grounding head for fine-grained localization.

3 Experiments

3.1 Experimental Setup

Datasets. As summarized in Tab. 1, we pretrain CoverVTG on seven video-language datasets covering diverse video domains, durations, and temporal annotations. We evaluate the pretrained model on four temporal grounding benchmarks: Ego4D-NLQ (Grauman et al., 2022), TaCoS (Regneri et al., 2013), Charades-STA (Sigurdsson et al., 2016), and ActivityNet-Captions (Krishna et al., 2017). Ego4D-NLQ and TaCoS mainly evaluate grounding on relatively long videos, whereas Charades-STA and ActivityNet-Captions contain shorter videos.

Evaluation Metrics. We evaluate temporal grounding performance using Recall@1 under different temporal Intersection-over-Union (tIoU) thresholds. Following common evaluation protocols, we

Table 1: Statistics of the pretraining datasets and temporal grounding benchmarks.

Dataset	#Videos	#Queries	Video Len.	Seg. Len.	Domain
Pretraining Datasets					
NaQ (Ramakrishnan et al., 2023)	4.9K	1031K	413s	1.1s	Open
DiDeMo (Anne Hendricks et al., 2017)	8.4K	33.0K	29s	7.5s	Open
QuerYD (Oncescu et al., 2021)	819	5.7K	278s	13.6s	Open
HiRest (Zala et al., 2023)	526	4.0K	263s	18.9s	Open
COIN (Tang et al., 2019)	11.8K	46.4K	145s	14.9s	Open
Momentor (Qian et al., 2024)	6.7K	136.4K	403s	49.5s	Open
YouCook2 (Zhou et al., 2018)	1.2K	9.6K	316s	19.7s	Cooking
Downstream Temporal Grounding Benchmarks					
Ego4D-NLQ (Grauman et al., 2022)	1.0K	10.0K	495s	8.2s	Open
TaCoS (Regneri et al., 2013)	75	9.8K	287s	27.9s	Cooking
Charades-STA (Sigurdsson et al., 2016)	4.8K	11.2K	31s	8.1s	Activity
ActivityNet-Captions (Krishna et al., 2017)	10K	37.4K	118s	36.0s	Activity

report R@0.3 and R@0.5 on Ego4D-NLQ and TaCoS, and R@0.5 and R@0.7 on Charades-STA and ActivityNet-Captions. We evaluate both zero-shot generalization of the pretrained model and benchmark-specific fine-tuning performance.

Implementation Details. We employ Perception Encoder (Bolya et al., 2026) as the visual and text backbones, and instantiate two model variants: **CoverVTG-B** with `PE-Core-B16` and **CoverVTG-L** with `PE-Core-L14`. CoverVTG-B processes videos at a spatial resolution of 384×384, while CoverVTG-L uses 336×336. With patch sizes of 16 and 14, respectively, both variants produce the same 24×24 spatial patch grid, allowing them to share the same visual-token budget configuration. We use HieraMamba (An & Grauman, 2026) as our detection head, and keep the hyperparameters unchanged.

We implement our method with PyTorch and MMAction2 (Contributors, 2020). By default, mixed-precision training and activation checkpointing are adopted to save memory. During training, we sample videos at 2 FPS and set $N_{\text{budget}} = 147456$. The retrieval clip length is set to $T_{\text{retrieval}} = 8$. For long videos, the local window size is adaptively selected from $\{1, 2, 3, 4\}$ according to the token budget. More implementation details can be found in Appendix A.3.

3.2 Main Results

Zero-Shot Temporal Grounding. As shown in Table 2, CoverVTG consistently outperforms conventional VTG method UniVTG across all four benchmarks, demonstrating substantially stronger zero-shot generalization in the Universal VTG setting. Compared with much larger MLLM-based methods, CoverVTG also achieves competitive performance. In particular, CoverVTG-L reaches 12.51/6.39 on Ego4D-NLQ and 38.02/22.37 on TaCoS, ranking second only to UniTime among the compared methods. On short video benchmarks, it obtains 41.62/22.83 on Charades-STA and 27.98/14.14 on ActivityNet-Captions, surpassing most MLLM-based baselines.

Notably, these results are achieved with a substantially smaller model and much lower inference cost. CoverVTG-L contains only 0.7B parameters, about $10\times$ fewer than typical 7B MLLM-based methods. Compared with UniTime, it reduces inference latency by 95.66%, 96.26%, 89.19%, and 91.83% on Ego4D-NLQ, TaCoS, Charades-STA, and ActivityNet-Caps, respectively. These results demonstrate that CoverVTG provides a favorable accuracy–efficiency trade-off for Universal VTG.

Fine-Tuned Temporal Grounding. We further fine-tune CoverVTG on each downstream benchmark and compare it with task-specific grounding methods in Tab. 3. After benchmark-specific fine-tuning, CoverVTG achieves consistently strong performance across all four datasets. In particular, CoverVTG-L obtains the best results on every reported metric, reaching 27.73/18.24 on Ego4D-NLQ, 61.30/51.05 on TaCoS, 72.52/54.98 on Charades-STA, and 50.80/31.79 on ActivityNet-Captions. The consistent improvement from CoverVTG-B to CoverVTG-L further demonstrates that our sparse observation and recovery design scales well with stronger visual encoders.

Table 2: Zero-shot results on four temporal grounding benchmarks. The best and the second-best results are indicated in **bold** and with underline, respectively.

Method	Params	Long Video				Short Video			
		Ego4D-NLQ		TaCoS		Charades-STA		ActivityNet-Caps	
		R@0.3	R@0.5	R@0.3	R@0.5	R@0.5	R@0.7	R@0.5	R@0.7
MLLM-based VTG Methods									
Mr.BLIP (Boris et al., 2024)	3B	6.49	3.20	24.59	14.32	-	-	-	-
Momentor (Qian et al., 2024)	7B	-	-	-	-	26.60	11.60	23.00	12.40
VTimeLLM (Huang et al., 2024)	7B	1.67	0.77	9.30	3.90	27.50	11.40	27.80	14.30
TimeChat (Ren et al., 2024)	7B	1.67	0.79	3.77	1.60	29.73	12.53	16.38	8.36
VTG-LLM (Guo et al., 2025)	7B	1.71	0.46	6.87	2.92	34.11	15.81	12.32	6.74
ED-VTG (Pramanick et al., 2025)	7B	-	-	14.50	6.00	39.30	19.80	**33.10**	**16.00**
UniTime (Li et al., 2026)	7B	**14.67**	**7.38**	**50.06**	**31.54**	**59.09**	**31.88**	22.77	14.14
UniVTG (Lin et al., 2023)	-	6.48	3.48	5.17	1.27	25.22	10.03	11.10	4.06
CoverVTG-B	0.4B	10.57	5.60	33.76	19.09	39.72	21.39	25.41	12.33
CoverVTG-L	0.7B	12.51	6.39	38.02	22.37	41.62	22.83	27.98	14.14
Latency@A100									
UniTime (Li et al., 2026)	7B	109.22s		61.78s		17.02s		24.60s	
CoverVTG-L	0.7B	4.74s -95.66%		2.31s -96.26%		1.84s -89.19%		2.01s -91.83%	

Table 3: Fine-tuned temporal grounding results on four benchmarks. The best results are indicated in **bold**.

Method	Long Video				Short Video			
	Ego4D-NLQ		TaCoS		Charades-STA		ActivityNet-Caps	
	R@0.3	R@0.5	R@0.3	R@0.5	R@0.5	R@0.7	R@0.5	R@0.7
2D-TAN (Zhang et al., 2020b)	5.04	2.05	45.61	35.77	56.64	36.21	46.16	29.21
UniVTG (Lin et al., 2023)	11.74	7.54	56.11	43.44	60.19	38.55	42.41	21.55
UnLoc-L (Yan et al., 2023)	-	-	-	-	60.80	38.40	48.30	30.20
SnAG (Mu et al., 2024)	15.87	11.26	56.44	44.86	64.62	46.26	48.55	30.56
DeCafNet (Lu et al., 2025)	18.10	12.55	57.36	46.79	68.79	47.55	-	-
HieraMamba (An & Grauman, 2026)	18.81	13.04	59.59	48.99	-	-	-	-
CoverVTG-B	24.59	15.98	58.86	47.90	69.17	52.40	49.32	30.75
CoverVTG-L	**27.73**	**18.24**	**61.30**	**51.05**	**72.52**	**54.98**	**50.80**	**31.79**

3.3 Ablation Studies

Unless otherwise specified, all ablation studies are conducted on CoverVTG-B under the default configuration and evaluated by its zero-shot VTG performance on Ego4D-NLQ.

Effect of major components. We first investigate the contributions of Coverage-Constrained Token Assignment (CCTA), the Recovery Adapter, and coarse-to-fine inference. Tab. 4 shows that each component plays a distinct role in CoverVTG. CCTA substantially reduces inference latency from $7.34s$ to $2.14s$, although aggressive token sparsification alone leads to an accuracy drop. Introducing the Recovery Adapter compensates for the information loss caused by sparsification, improving the CCTA-based model from $5.27/2.38$ to $6.42/4.07$ R@0.3/R@0.5 while retaining low latency ($2.33s$). Finally, coarse-to-fine inference further boosts performance to $10.57/5.60$ with only a minor latency increase to $2.59s$. Overall, the full model improves both grounding accuracy and efficiency, demonstrating the complementary effects of sparse observation, feature recovery, and coarse-to-fine inference.

Effect of different token sampling strategies. We compare our coverage-constrained token assignment with several alternative sampling strategies in Tab. 5. Max, min, and average pooling aggregate local observations according to simple statistics, while Phase Shift adopts a deterministic cyclic sampling pattern that shifts the selected spatial position across consecutive frames. Although Phase Shift already benefits from more evenly distributed spatio-temporal observations and achieves $9.43/5.27$ R@0.3/R@0.5, our Coverage-Constrained strategy further improves the performance to $10.57/5.60$. These results indicate that preserving structured spatio-temporal coverage is more effective than statistical aggregation, while adaptively selecting among coverage-preserving assignments provides additional gains over a fixed sampling pattern.

Table 4: Ablation of the major components in CoverVTG.

CCTA	Recovery Adapter	Coarse-to-Fine Inference	Ego4D-NLQ R@0.3	Ego4D-NLQ R@0.5	Ego4D-NLQ Latency
×	×	×	6.13	3.50	7.34s
✓	×	×	5.27	2.38	2.14s
×	✓	×	6.54	4.14	7.51s
✓	✓	×	6.42	4.07	2.33s
✓	✓	✓	10.57	5.60	2.59s

Table 5: Sampling strategies.

Strategy	Ego4D-NLQ R@0.3	Ego4D-NLQ R@0.5
Max Pooling	8.56	4.71
Min Pooling	7.95	4.33
Avg Pooling	7.17	3.61
Phase Shift	9.43	5.27
CCTA	10.57	5.60

Table 6: Retrieval clip length.

$T_{\text{retrieval}}$	Ego4D-NLQ R@0.3	Ego4D-NLQ R@0.5
4	10.24	5.09
8	10.57	5.60
16	10.53	5.60
32	10.45	5.49
64	10.29	5.17

Table 7: Feature recoverability under sparse visual observation. Dense visual features are used as the reference.

Method	Cos. ↑
Sparse w/o Recovery	0.63
Sparse w/ Recovery	0.84

Effect of retrieval granularity. We study the influence of the retrieval clip length $T_{\text{retrieval}}$ in Tab. 6. The best performance is achieved at $T_{\text{retrieval}} = 8$, reaching 10.57/5.60 R@0.3/R@0.5. Using overly short clips provides insufficient temporal context for reliable coarse retrieval, while excessively long clips lead to coarser temporal granularity and weaker discrimination of relevant moments. The comparable results at $T_{\text{retrieval}} = 8$ and 16 also indicate that CoverVTG is relatively robust to the retrieval granularity, with $T_{\text{retrieval}} = 8$ offering a favorable balance between temporal context and localization resolution.

Can sparse observations be recovered? Our theoretical analysis suggests that coverage preserves complementary spatial and temporal anchors that provide a basis for local feature recovery. To empirically examine this property, we use densely encoded visual features produced by the same visual backbone as the reference and compare sparse representations with and without the Recovery Adapter. We report token-level cosine similarity (Cos.), where a higher value indicates greater consistency with the dense representation.

Specifically, we randomly sample 10 videos from Ego4D-NLQ and use the same videos for both settings. For each retained token, we compute the cosine similarity between its feature from sparse encoding and the corresponding feature at the same spatio-temporal location from dense encoding, and then average the similarities over all tokens and videos. As shown in Tab. 7, sparse encoding without recovery achieves an average cosine similarity of 0.63, whereas incorporating the Recovery Adapter increases it to 0.84. This improvement indicates that the Recovery Adapter effectively exploits the complementary information preserved by sparse observations, yielding token representations that are substantially more consistent with those obtained through dense encoding. These results provide empirical evidence for the observability-to-recoverability connection underlying CoverVTG.

4 Conclusion

We present CoverVTG, a lightweight backbone-centric framework for Universal Video Temporal Grounding. Instead of relying on semantic importance that is unavailable before deep visual encoding, CoverVTG performs coverage-constrained sparse observation directly at the patch-embedding stage. The resulting spatial-temporal coverage provides complementary observations for discarded tokens, establishing the *coverage* → *observability* → *recoverability* principle. Lightweight Recovery Adapters further exploit these observations throughout a frozen visual backbone, enabling efficient temporal grounding across videos of different durations. Experiments demonstrate that CoverVTG achieves competitive zero-shot generalization with substantially fewer parameters and lower inference latency than MLLM-based approaches, while also showing strong performance after task-specific fine-tuning.

AI USE STATEMENT

In this work, we used generative AI tools for language editing and assistance with manuscript preparation. We have not used generative AI tools to generate experimental data or reported results. Additionally, generative AI tools were used to suggest alternative wording for parts of the manuscript. We have reviewed all AI-assisted work. All AI-assisted content was manually inspected and revised by the authors. The technical descriptions and mathematical formulations were verified against our method and implementation, all reported results were obtained from our own experiments, and the cited literature was checked by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We have made efforts to ensure that the results in this work are reproducible. The complete formulation of CoverVTG, including Coverage-Constrained Token Assignment, the coverage–observability–recoverability analysis, the Spatio-Temporal Recovery Adapter, training objectives, and the universal VTG inference procedure, is provided in Sec. 2. The datasets, evaluation protocols, model variants, and primary implementation settings are described in Sec. 3. Additional optimization hyperparameters and detailed video-processing procedures for both training and inference, including frame sampling, visual-token budgets, adaptive window sizes, and the handling of overlength videos, are provided in Appendix A.3. Upon acceptance, we will publicly release all source code and trained model checkpoints to further facilitate reproducibility.

REFERENCES

Joungbin An and Kristen Grauman. Hieramamba: Video temporal grounding via hierarchical anchor-mamba pooling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16954–16965, 2026.

Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE international conference on computer vision*, pp. 5803–5812, 2017.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1708–1718. IEEE, 2021.

Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Bangalath, et al. Perception encoder: The best visual embeddings are not at the output of the network. *Advances in Neural Information Processing Systems*, 38:60884–60937, 2026.

Meinardus Boris, Batra Anil, Rohrbach Anna, and Rohrbach Marcus. The surprising effectiveness of multimodal large language models for video moment retrieval. *arXiv e-prints*, pp. arXiv–2406, 2024.

Zhiguo Chen, Xun Jiang, Xing Xu, Zuo Cao, Yijun Mo, and Heng Tao Shen. Joint searching and grounding: Multi-granularity video content retrieval. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 975–983, 2023.

MMAction2 Contributors. Openmmlab's next generation video understanding toolbox and benchmark. `https://github.com/open-mmlab/mmaction2`, 2020.

Yisen Feng, Haoyu Zhang, Meng Liu, Weili Guan, and Liqiang Nie. Object-shot enhanced grounding network for egocentric video. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24190–24200. IEEE, 2025.

Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5277–5285. IEEE, 2017.

Aleksandr Gordeev, Vladimir Dokholyan, Irina Tolstykh, and Maksim Kuprashevich. Saliency-guided detr for moment retrieval and highlight detection. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 907–916. IEEE, 2026.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18995–19012, 2022.

Yongxin Guo, Jingyu Liu, Mingda Li, Dingxin Cheng, Xiaoying Tang, Dianbo Sui, Qingbin Liu, Xi Chen, and Kevin Zhao. Vtg-llm: Integrating timestamp knowledge into video llms for enhanced video temporal grounding. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 3302–3310, 2025.

Tanveer Hannan, Md Mohaiminul Islam, Thomas Seidl, and Gedas Bertasius. Rgnet: A unified clip retrieval and grounding network for long videos. In *European Conference on Computer Vision*, pp. 352–369. Springer, 2024.

Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. Vtimellm: Empower llm to grasp video moments. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14271–14280. IEEE, 2024.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pp. 706–715, 2017.

Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. *Advances in Neural Information Processing Systems*, 34:11846–11858, 2021.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

Zeqian Li, Shangzhe Di, Zhonghua Zhai, Weilin Huang, Yanfeng Wang, and Weidi Xie. Universal video temporal grounding with generative multi-modal large language models. *Advances in Neural Information Processing Systems*, 38:64426–64455, 2026.

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 *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2782–2792. IEEE, 2023.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Ye Liu, Siyuan Li, Yang Wu, Chang-Wen Chen, Ying Shan, and Xiaohu Qie. Umt: Unified multi-modal transformers for joint video moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3042–3051, 2022.

Ye Liu, Jixuan He, Wanhua Li, Junsik Kim, Donglai Wei, Hanspeter Pfister, and Chang Wen Chen. R 2-tuning: Efficient image-to-video transfer learning for video temporal grounding. In *European conference on computer vision*, pp. 421–438. Springer, 2024.

Ye Liu, Kevin Qinghong Lin, Chang-Wen Chen, and Mike Zheng Shou. Videomind: A chain-of-lora agent for temporal-grounded video reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 57481–57506, 2026.

Zhijian Liu, Ligeng Zhu, Baifeng Shi, Zhuoyang Zhang, Yuming Lou, Shang Yang, Haocheng Xi, Shiyi Cao, Yuxian Gu, Dacheng Li, et al. Nvila: Efficient frontier visual language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4122–4134. IEEE, 2025.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Zijia Lu, ASM Iftekhar, Gaurav Mittal, Tianjian Meng, Xiawei Wang, Cheng Zhao, Rohith Kukkala, Ehsan Elhamifar, and Mei Chen. Decafnet: Delegate and conquer for efficient temporal grounding in long videos. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24066–24076. IEEE, 2025.

WonJun Moon, Sangeek Hyun, SangUk Park, Dongchan Park, and Jae-Pil Heo. Query-dependent video representation for moment retrieval and highlight detection. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23023–23033. IEEE, 2023.

Fangzhou Mu, Sicheng Mo, and Yin Li. Snag: Scalable and accurate video grounding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18930–18940. IEEE, 2024.

Andreea-Maria Oncescu, Joao F Henriques, Yang Liu, Andrew Zisserman, and Samuel Albanie. Queryd: A video dataset with high-quality text and audio narrations. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2265–2269. IEEE, 2021.

Shraman Pramanick, Effrosyni Mavroudi, Yale Song, Rama Chellappa, Lorenzo Torresani, and Triantafyllos Afouras. Enrich and detect: Video temporal grounding with multimodal llms. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 24297–24308. IEEE, 2025.

Long Qian, Juncheng Li, Yu Wu, Yaobo Ye, Hao Fei, Tat-Seng Chua, Yueting Zhuang, and Siliang Tang. Momentor: Advancing video large language model with fine-grained temporal reasoning. *arXiv preprint arXiv:2402.11435*, 2024.

Santhosh Kumar Ramakrishnan, Ziad Al-Halah, and Kristen Grauman. Naq: Leveraging narrations as queries to supervise episodic memory. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6694–6703. IEEE, 2023.

Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics*, 1:25–36, 2013.

Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14313–14323. IEEE, 2024.

Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European conference on computer vision*, pp. 510–526. Springer, 2016.

Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1207–1216. IEEE, 2019.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Hao Wang, Zheng-Jun Zha, Liang Li, Dong Liu, and Jiebo Luo. Structured multi-level interaction network for video moment localization via language query. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7022–7031. IEEE, 2021.

Shen Yan, Xuehan Xiong, Arsha Nagrani, Anurag Arnab, Zhonghao Wang, Weina Ge, David Ross, and Cordelia Schmid. Unloc: A unified framework for video localization tasks. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13577–13587. IEEE, 2023.

Yitian Yuan, Lin Ma, Jingwen Wang, Wei Liu, and Wenwu Zhu. Semantic conditioned dynamic modulation for temporal sentence grounding in videos. *Advances in Neural Information Processing Systems*, 32, 2019.

Abhay Zala, Jaemin Cho, Satwik Kottur, Xilun Chen, Barlas Oguz, Yashar Mehdad, and Mohit Bansal. Hierarchical video-moment retrieval and step-captioning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23056–23065. IEEE, 2023.

Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. Dense regression network for video grounding. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10284–10293. IEEE, 2020.

Xiangyu Zeng, Kunchang Li, Chenting Wang, Xinhao Li, Tianxiang Jiang, Ziang Yan, Songze Li, Yansong Shi, Zhengrong Yue, Yi Wang, et al. Timesuite: Improving mllms for long video understanding via grounded tuning. In *International Conference on Learning Representations*, volume 2025, pp. 38057–38081, 2025.

Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. Span-based localizing network for natural language video localization. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 6543–6554, 2020a.

Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. Learning 2d temporal adjacent networks for moment localization with natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 12870–12877, 2020b.

Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

A APPENDIX

A.1 RELATED WORK

Video Temporal Grounding (VTG). Conventional video temporal grounding methods typically localize query-relevant moments through proposal-based matching (Anne Hendricks et al., 2017; Gao et al., 2017; Yuan et al., 2019; Wang et al., 2021; Chen et al., 2023), span prediction (Zhang et al., 2020a; Zeng et al., 2020; Liu et al., 2022; Mu et al., 2024; Liu et al., 2024), or DETR-style direct boundary regression (Lei et al., 2021; Moon et al., 2023; Gordeev et al., 2026). UniVTG (Lin et al., 2023) further develops a unified formulation that accommodates multiple temporal grounding tasks and supervision types. Despite their effectiveness, most of them operate on pre-extracted features obtained from frozen image or video encoders. Such a design substantially reduces training cost, but also decouples visual representation learning from the grounding objective. As a result, adapting the visual representations themselves to diverse viewpoints, domains, video durations, and free-form queries is less direct, making these pipelines less naturally suited to the Universal VTG setting. In contrast, CoverVTG enables efficient task adaptation through lightweight recovery modules while directly reducing the visual observations entering the frozen backbone, providing a more flexible and scalable solution for Universal VTG.

Temporal Grounding with Multi-modal Large Language Models (MLLMs). Recent works increasingly adopt multimodal large language models (Liu et al., 2023; Li et al., 2024; Bai et al., 2025; Liu et al., 2025) to improve temporal video understanding and cross-domain generalization. VTimeLLM (Huang et al., 2024) and TimeChat (Ren et al., 2024) enhance video LLMs with temporal-aware training and representations, while VTG-LLM (Guo et al., 2025) and TimeSuite (Zeng et al., 2025) further improve long-video processing through temporal modeling and visual-token compression. UniTime (Li et al., 2026) explicitly studies Universal VTG across videos with diverse viewpoints, domains, durations, and query types. Although MLLM-based methods provide strong generalization, they are computationally expensive due to their large model scale and the cost of processing long visual sequences. Existing efficiency designs mainly control the number of visual tokens consumed by downstream multimodal modules. In contrast, CoverVTG explores a complementary backbone-centric direction by reducing the number of visual observations that require expensive semantic encoding in the first place.

A.2 LIMITATIONS

CoverVTG also has several limitations. First, our recoverability analysis relies on local spatio-temporal regularity, which may become less reliable around abrupt scene changes, rapid motion, or other highly non-smooth video regions. Second, the pre-backbone assignment is intentionally query-independent and relies only on shallow visual cues; therefore, semantically important but visually subtle observations may not always be preferentially retained. Finally, for very long videos, fine-grained grounding depends on the preceding coarse retrieval stage, so retrieval errors may propagate to the final localization result. Addressing these limitations through more robust structural priors provides promising directions for future work.

A.3 IMPLEMENTATION DETAILS

Training Configuration. Tab. 8 summarizes the optimization configurations used for pretraining and downstream fine-tuning. We use AdamW (Loshchilov & Hutter, 2017) for all stages with a weight decay of 0.05 and a cosine learning-rate schedule (Loshchilov & Hutter, 2016). During pretraining, the base learning rate is set to 1e-3. The model is pretrained for 10 epochs with one warm-up epoch. For downstream fine-tuning, the base learning rate and training epochs are adjusted for each benchmark.

Data Processing. Tab. 9 summarizes the video processing configurations during training and inference. Both stages sample videos at 2 FPS. CoverVTG-B and CoverVTG-L use spatial resolutions of 384×384 and 336×336, respectively, resulting in the same 24×24 spatial patch grid. We set $T_{\text{retrieval}} = 8$ and select the adaptive local window size from $L \in \{1, 2, 3, 4\}$ for both training and inference.

Table 8: **Training setting.**

Configuration	Pretraining	Finetuning			
		Ego4D-NLQ	TaCoS	Charades-STA	ActivityNet-Caps
Learning Rate	1e-3	1e-4	2e-4	1e-4	1e-4
Training Epochs	10	7	10	6	5
Warm-up Epochs	1	2			
Optimizer	AdamW	AdamW			
Weight Decay	0.05	0.5			
Learning Rate Schedule	Cosine Decay	Cosine Decay			

Table 9: **Data processing setting.**

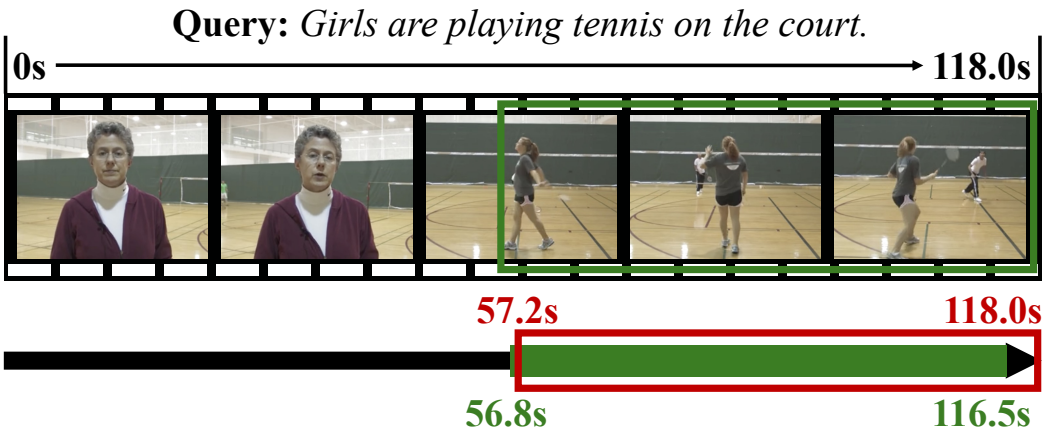
Configuration	Training	Inference
FPS	2	2
Frame size	384^2 (B) / 336^2 (L)	384^2 (B) / 336^2 (L)
Candidate L	$\{1, 2, 3, 4\}$	$\{1, 2, 3, 4\}$
$T_{\text{retrieval}}$	8	8
N_{budget}	147456	442368
Maximum supported frames	4096	12288
Maximum video duration	34 min	102 min

A key property of CoverVTG is that the visual-token budget at inference time does not need to be identical to that used during training. While we train the model with $N_{\text{budget}} = 147456$, we increase the inference budget to 442368, i.e., $3\times$ the training budget. This allows CoverVTG to exploit additional computation at test time and process substantially longer videos without retraining the model. The same candidate set $L \in \{1, 2, 3, 4\}$ and retrieval granularity $T_{\text{retrieval}} = 8$ are retained, such that the learned sparse observation and recovery mechanism can be directly applied under the enlarged inference budget.

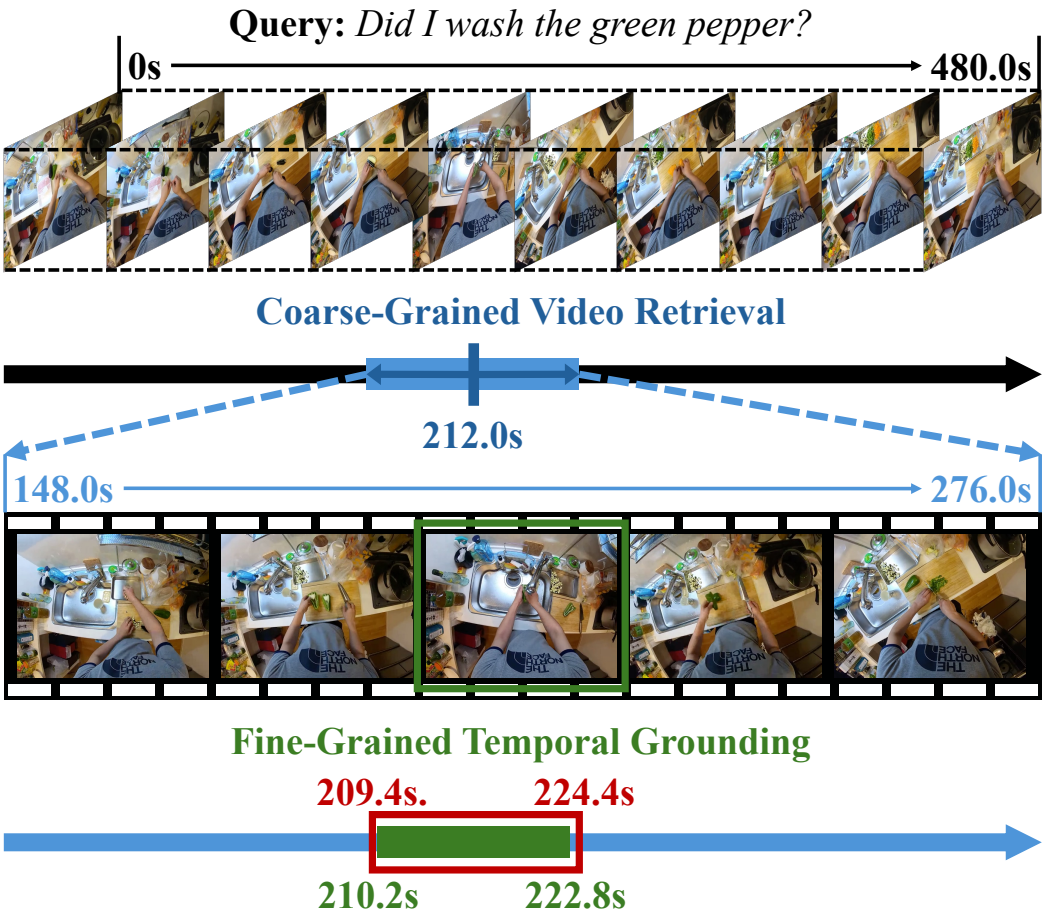
The treatment of extremely long videos also differs between training and inference. For training videos exceeding the maximum supported length, we randomly crop a segment of the maximum supported length that contains annotated ground-truth moments. During inference, videos longer than the supported length are split into temporal chunks, each containing at most the maximum supported number of frames, and processed sequentially.

A.4 Qualitative Results

Fig. 3 visualizes representative grounding results on both short- and long-video benchmarks. For the short-video example, CoverVTG directly produces temporal boundaries that closely align with the ground-truth moment, demonstrating accurate fine-grained localization without coarse retrieval. For the long-video example, coarse-grained retrieval first identifies a compact temporal region that successfully covers the target moment. Fine-grained grounding then substantially narrows this region and predicts boundaries closely matching the ground truth. These examples qualitatively demonstrate that CoverVTG can effectively handle videos of different durations, while the coarse-to-fine inference strategy provides an efficient way to localize precise moments in long videos.



(a) short-video temporal grounding on the ActivityNet-Captions benchmark.



(b) long-video temporal grounding on the Ego4D-NLQ benchmark.

Figure 3: Qualitative Results for both short- and long-video temporal grounding benchmarks. **Red** indicates the ground truth (GT), **blue** denotes the results of coarse-grained video retrieval, and **green** represents the results of fine-grained temporal grounding.