DYNASENSE: LEARNING TO PERCEIVE DYNAMIC CHANGES IN VIDEOS FOR TEMPORAL GROUNDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Current video Multi-modal Large Language Models (Video-MLLMs) for Video Temporal Grounding (VTG) rely on RGB-centric static visual representations, which capture visual content but offer limited modeling of visual state evolution over time. Consequently, these approaches fail to perceive dynamic visual changes at the temporal boundaries of events, where fine-grained temporal transitions are critical for accurate localization. To address this limitation, we propose DynaSense, a Video-MLLM that improves temporal grounding by explicitly modeling dynamic representations of temporal changes within videos. DynaSense models continuously evolving videos by representing visual state changes at each temporal step relative to the preceding step. It first aligns visual features at the current timestamp with the most relevant references from the preceding timestamp within a position-free embedding space. Subsequently, it encodes the resulting temporal change as the difference between the corresponding visual features after incorporating 2D spatial positional encodings. This dynamic representation captures both appearance differences and spatial displacements, providing temporally discriminative evidence for localizing event boundaries. Furthermore, we introduce video state change recognition dataset (VideoSCR), a video state-change recognition dataset constructed via an automated annotation pipeline, to enhance the perception and recognition of video dynamics. DynaSense consists of two training stages. The first performs state-change question answering on VideoSCR to facilitate the learning of dynamic video features, and the second introduces video temporal grounding data to strengthen the capability for temporal localization. Experiments across multiple VTG benchmarks demonstrate that DynaSense consistently improves temporal grounding accuracy and can be readily integrated into other Video-MLLMs to enhance their temporal grounding capabilities.

1 INTRODUCTION

Video-MLLMs can recognize visible content and describe ongoing events at a coarse level, yet their fine-grained temporal perception remains limited, which constrains performance on video temporal grounding. Recent studies have improved temporal understanding and basic localization through time-sensitive visual encoding and boundary-aware training (Huang et al., 2024a; Ren et al., 2024; Wang et al., 2024; Zeng et al., 2025a), timestamp representations (Chen et al., 2024; Guo et al., 2026; 2025b; Li et al., 2026b), systematic optimization of grounding data and training recipes (Zhang et al., 2026), and complementary advances in query enrichment, semantic supervision, and temporal output modeling (Intelligent Editing Team, 2025; Pang et al., 2026; Qian et al., 2024; Wang et al., 2025a; Zeng et al., 2025b). However, a central question remains: Can Video-MLLMs precisely localize event boundaries? We argue that video temporal grounding requires two capabilities. The first is semantic alignment between the event query and the corresponding video segment, and the second is perception of second-level dynamic changes across the video. Current Video-MLLMs have reached a reasonable level on the former, while the latter remains limited.

Most Video-MLLMs uniformly sample RGB frames and encode each frame with an image-oriented backbone, thus providing mainly static appearance evidence for recognition and vision-language alignment (Fan et al., 2026; Hou et al., 2026; Nie et al., 2024; Wang et al., 2026; Zhao et al., 2025). With such appearance-only inputs, temporal dynamics must be inferred indirectly from static feature sequences, and event-level evaluations consistently report persistent difficulties in fine-grained

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.

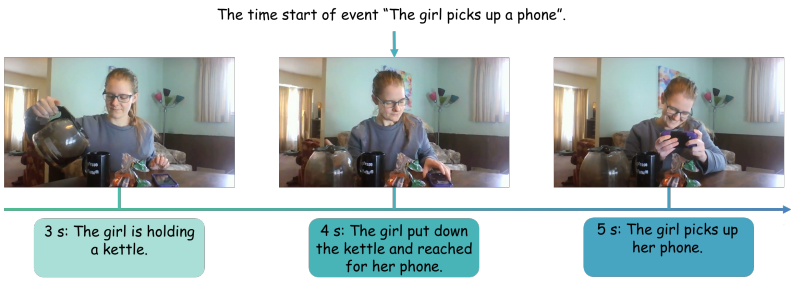


Figure 1: **An illustration of the dynamic change at an event boundary.** Adjacent frames share most scene content and appearance, while event-related states change across the transition. The sequence shows a girl holding a kettle, putting it down and reaching for her phone, and then picking up the phone. The arrow marks the onset of the phone-picking event.

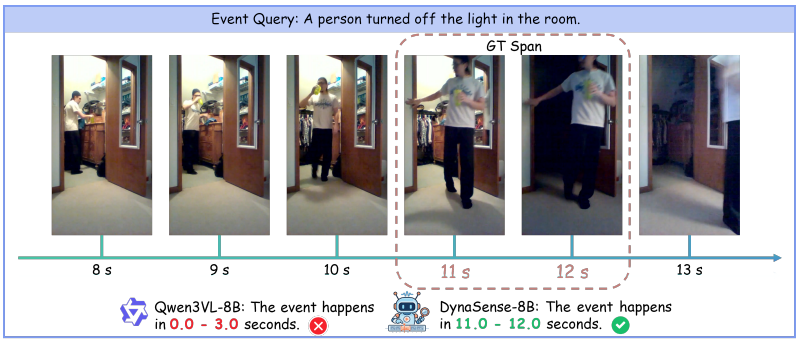


Figure 2: **Qualitative comparison of temporal grounding results.** Given the same event query, the appearance-only Qwen3VL produces a coarse and temporally misaligned prediction, whereas DynaSense accurately localizes the short interval corresponding to the actual state transition.

localization (Guo et al., 2025b; Liu et al., 2024b; Wang et al., 2025a), despite advances in frame selection and dense sampling (Fan et al., 2026; Nie et al., 2024; Wang et al., 2026). As shown in Fig. 1, an event boundary is typically defined by a dynamic state transition between neighboring moments, formalized by Kinetics-GEB+ as before/after states (Wang et al., 2022). For example, the event picks up a phone occurs as the girl transitions from reaching for the phone to holding it. Fine-grained temporal localization therefore requires explicit modeling of dynamic changes between neighboring temporal observations.

These limitations have motivated complementary temporal modeling approaches. ST-LLM models spatial-temporal token sequences within the LLM (Liu et al., 2024a). ReMoRa refines codec motion vectors with a dedicated motion module pretrained on dense optical-flow targets and further models temporal dependencies via a hierarchical state-space module (Yashima et al., 2026). Its motion-processing components remain separate from the appearance image encoder. Other approaches exploit compressed or predictive representations. Video-LaVIT learns separate visual and motion tokenizers, while EMA fuses appearance and motion within GOPs. AdaCodec, CoPE-VideoLM, and LLaVA-OneVision-2 introduce motion and residual signals in visual interfaces (Glint Lab et al., 2026; Hou et al., 2026; Jin et al., 2024; Sarkar et al., 2026; Zhao et al., 2025). However, they all require dedicated encoders with pretraining for dynamic features (Hou et al., 2026; Sarkar et al., 2026; Yashima et al., 2026). This adds pretraining and alignment costs and complicates the architecture, requiring both the original ViT for static RGB features and separate encoders for dynamic video features. These observations motivate a natural question: *Can video dynamics be implicitly encoded in the ViT visual feature space, rather than represented in RGB or motion-vector space?*

We present **DynaSense**, which derives dynamic visual features directly from RGB appearance features. For each consecutive frame pair, it establishes local patch correspondences in a position-free space and computes differences after incorporating spatial positions to encode content residuals and displacement cues. Both dynamic and RGB static features are encoded by the native ViT without

dynamics-features-specific encoder. Appearance and dynamic features are projected by independent mergers, with dynamic mergers initialized from pretrained appearance mergers and fine-tuned separately, requiring only approximately **0.1B** additional parameters and fine-tuning on a small amount of VQA data. Inspired by Kinetics-GEB+, we construct **VideoSCR**, a video state-change recognition dataset for interpreting dynamic representations (Wang et al., 2022), and design a two-stage fine-tuning framework grounding temporal localization in state-change recognition, first describing states before and after boundaries and then transferring to query-conditioned grounding. Fig. 2 compares the inference results of Qwen3VL-8B and our DynaSense on a sample containing a one-second event segment. Qwen3VL-8B produces a coarse prediction that is also semantically misaligned with the queried event, whereas DynaSense accurately localizes the actual event interval. Our contributions can be summarized as follows:

- We propose DynaSense, a Video-MLLM that extracts dynamic features from native patch embeddings via position-free local matching and position-enriched differencing. These features capture content changes and spatial displacement cues without an additional motion encoder and reach the LLM through a dual-stream interface reusing the original visual backbone.
- We construct VideoSCR, a video state-change recognition dataset combining boundary-centered annotations with synthesized state transitions. It supports a two-stage curriculum from before/after state description to query-conditioned temporal grounding.
- Extensive experiments show that DynaSense achieves competitive performance across multiple video temporal grounding benchmarks and significantly outperforms the appearance-only baseline. These results demonstrate that explicitly modeling dynamic changes benefits fine-grained temporal localization.

2 Related Work

Video temporal grounding with Video-MLLMs. Video temporal grounding localizes the temporal interval described by a natural-language query. Early DETR-style methods formulate moment retrieval as set prediction over pre-extracted video and text features (Lei et al., 2021), while recent Video-MLLMs support generative localization via instruction-following interfaces. TimeChat develops time-sensitive video representations (Ren et al., 2024), VTimeLLM adopts boundary-aware staged training (Huang et al., 2024a), and HawkEye explores time-aware training with coarse temporal representations (Wang et al., 2024). TimeLens systematically studies grounding data quality, timestamp encoding, and optimization recipes (Zhang et al., 2026), while UniVTG unifies related grounding tasks via shared formulations (Lin et al., 2023). Within Video-MLLMs, recent work improves localization through fine-grained temporal reasoning and query enrichment (Pramanick et al., 2025; Qian et al., 2024; Wang et al., 2025a), semantic supervision and causal event modeling (Guo et al., 2025a; Pang et al., 2026), dedicated temporal output representations (Wu et al., 2025; Zeng et al., 2025b), and expert-guided or predictive temporal reasoning (Yang et al., 2025; Zheng et al., 2026b). Query-guided token selection and instruction-conditioned sampling further address frame retention (Fan et al., 2026; Liu et al., 2024b; Wang et al., 2026). LITA combines time tokens with SlowFast visual tokens (Huang et al., 2024b), while Seq2Time transfers sequential knowledge to temporal grounding (Deng et al., 2025). VideoChat-R1 and Time-R1 explore reinforcement post-training (Li et al., 2025; Wang et al., 2025c), VideoChat-Flash compresses long video contexts hierarchically (Li et al., 2026a), and OmniVTG couples open-world data with self-correction training (Zheng et al., 2026a). VTimeCoT instead uses training-free visual tools for temporal reasoning (Zhang et al., 2025a).

Video dynamic representations in Video-MLLMs. Foundational approaches combine appearance with optical flow, rank pooling, or space-time attention (Bertasius et al., 2021; Bilen et al., 2016; Simonyan & Zisserman, 2014). Explicit dynamics have a long history spanning feature-level temporal differences, state-change descriptions, multi-scale differencing, correlation-based motion features, correspondence distillation, and compressed-domain boundary detection (Kwon et al., 2020; Li et al., 2022; 2020; Liu et al., 2025; Wang et al., 2021; 2022). Recent Video-MLLMs incorporate such evidence through spatial-temporal token modeling, dense or high-frame-rate sampling, and codec motion vectors refined with optical-flow supervision (Liu et al., 2024a; Nie et al., 2024; Yashima et al., 2026; Zhao et al., 2026). Another family exploits compressed or predictive representations through decoupled tokenization, motion-aware GOP encoding, and codec-stream interfaces

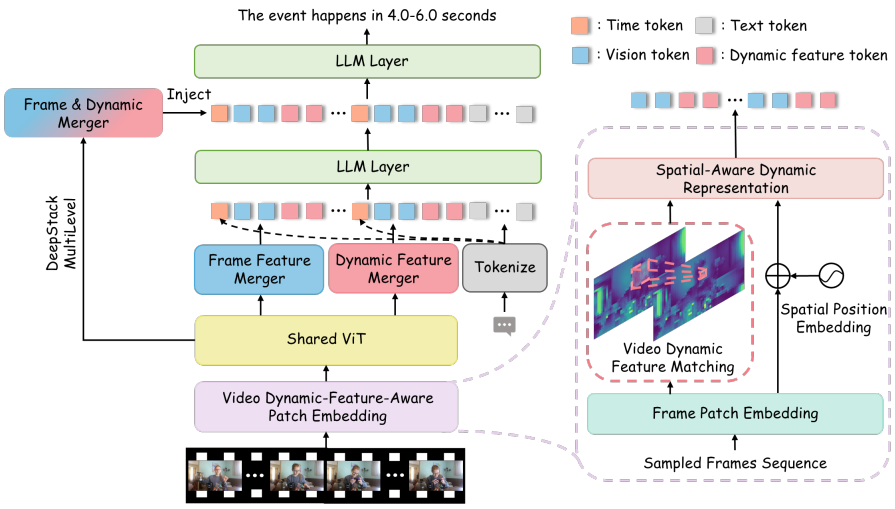


Figure 3: **The architecture of DynaSense.** Position-free patch embeddings establish local correspondences across consecutive frames. Position-enriched matched differences form video dynamic features. A shared ViT processes appearance and dynamic streams with separate attention segments, followed by two independent mergers that map both streams into the language model.

(Glint Lab et al., 2026; Hou et al., 2026; Jin et al., 2024; Sarkar et al., 2026; Zhao et al., 2025). However, they all require dedicated tokenizers or encoders with pretraining for dynamic features (Hou et al., 2026; Sarkar et al., 2026; Yashima et al., 2026). These designs demonstrate the value of explicit temporal evidence but typically require codec-specific preprocessing, dedicated tokenizers or encoders, and additional alignment training.

3 METHODS

3.1 PRELIMINARIES

We formulate video temporal grounding as the task of identifying the video segment semantically aligned with a natural-language description and accurately predicting the temporal boundaries of that segment. We decompose the video representation into static features F_{s} and dynamic features F_{d}. The static features provide RGB-based visual evidence for associating the event described by the query with its corresponding video content, whereas the dynamic features capture changes between adjacent temporal observations and thereby support precise localization of event boundaries. Given a video of duration L and a natural-language query q, our model DynaSense Φ predicts the temporal interval as

$$\hat{s} = \Phi(F_{\mathrm{s}}, F_{\mathrm{d}}, q), \qquad \hat{s} = [\hat{t}_s, \hat{t}_e], \qquad 0 \le \hat{t}_s \le \hat{t}_e \le L, \tag{1}$$

where $\hat{t}_s$ and $\hat{t}_e$ denote the predicted start and end timestamps of the grounded event.

Accordingly, video temporal grounding hinges on two central requirements. The first is semantic alignment between the video segment and the language description, and the second is precise localization of the fine-grained temporal boundaries of the corresponding event. The former is primarily supported by static features F_{s}, which associate query elements such as actors, objects, and actions with the corresponding visual content, while the latter depends on dynamic features F_{d}, which expose local changes across neighboring temporal units. Boundary localization is particularly challenging because adjacent frames often resemble event frames, limiting static cues for identifying the exact transition moment.

3.2 DYNASENSE

Fig. 3 overviews DynaSense. Given a sampled video and language query, DynaSense establishes local patch correspondences between consecutive frames in a position-free embedding space and

constructs dynamic features by differencing matched features. Appearance and dynamic inputs are processed by a shared ViT and projected into the language space through dual-stream mergers. The LLM jointly reasons over static semantics and dynamic changes to predict the temporal interval.

Video Dynamic Feature Matching. Given an input video, we uniformly sample a frame sequence and patchify each frame via frame patch embedding, yielding patch embeddings $E_{t,i,j}$ where t indexes the timestamp of frame and (i, j) indexes the 2D spatial location. For a current patch (i, j), let $\Omega_w(i, j)$ be the valid grid locations in a square window of odd width w centered at (i, j) in the preceding frame, where (u, v) denotes the 2D spatial location of a candidate patch in the preceding frame. The matched location is represented as

$$(u^*_{t,i,j}, v^*_{t,i,j}) = \underset{(u,v)\in\Omega_w(i,j)}{\arg\min} \left\| E_{t,i,j} - E_{t-1,u,v} \right\|_1, \qquad t \geq 1, \tag{2}$$

Equal minimum distances are resolved in favor of the nearest spatial candidate, with any remaining tie resolved by the fixed candidate order. We argue that patch matching should be governed primarily by the semantic features of the patches themselves rather than by spatial positions, with spatial proximity considered only when semantic features are similar. We therefore perform patch feature matching after frame patch embedding and before 2D spatial position encoding. Each current patch retains a match, and correspondences need not be one-to-one.

Correspondence is established locally, where each patch refers only to candidates within a window centered at the same location in the preceding frame. This design bounds the search cost while avoiding spurious matches from distant regions, reflecting that event-defining changes are typically local across adjacent frames. The resulting correspondences then serve as references for constructing spatial-aware dynamic representations below.

Spatial-Aware Dynamic Representation. Let $P_{i,j}$ be the interpolated two-dimensional absolute position embedding at grid location (i, j), and let $Z_{t,i,j} = E_{t,i,j} + P_{i,j}$ denote the static appearance feature enriched with positional information. We define the video dynamic feature as

$$D_{0,i,j} = \mathbf{0}, \qquad D_{t,i,j} = Z_{t,i,j} - Z_{t-1,u^*_{t,i,j},v^*_{t,i,j}}, \quad t \geq 1. \tag{3}$$

Expanding this expression gives

$$D_{t,i,j} = \underbrace{E_{t,i,j} - E_{t-1,u^*_{t,i,j},v^*_{t,i,j}}}_{\text{matched content residual}} + \underbrace{P_{i,j} - P_{u^*_{t,i,j},v^*_{t,i,j}}}_{\text{spatial displacement cue}}. \tag{4}$$

Equation 4 explains the ordering of the operations. Matching uses visual similarity without positional information, whereas the subsequent difference preserves information about a change in matched location. The second term implicitly characterizes the displacement of dynamic regions between consecutive frames through the difference of position embeddings. For same-location matches, the term vanishes, whereas locally moved similar content can retain a nonzero position term despite a small appearance residual. The first dynamic input is zero because no preceding frame exists.

Unified Visual Encoding. Appearance and dynamic inputs have the same shape and pass through a shared vision transformer f_θ. For each visual level ℓ, write

$$H^{a,\ell} = f^\ell_\theta(Z), \qquad H^{d,\ell} = f^\ell_\theta(D). \tag{5}$$

In implementation, the inputs are packed together for a single forward call, with duplicated spatial rotary embeddings. Attention remains confined to the token sequence within each frame. Fusion across streams and time is handled by the language model.

For the final visual level and each of the three deepstack levels, we use distinct appearance and dynamic mergers to project the corresponding features as follows:

$$A^\ell_t = M^\ell_a(H^{a,\ell}_t), \qquad U^\ell_t = M^\ell_d(H^{d,\ell}_t). \tag{6}$$

The native spatial merge factor is two along each axis. Each merger therefore produces $n = H_g W_g/4$ tokens per frame, with the language model's embedding dimension. All four appearance mergers and all four dynamic mergers have independent parameter objects. Both families are initialized from the corresponding pretrained Qwen3-VL mergers. Copying initializes compatible interfaces but does not tie their weights during optimization.

Temporal Packaging. For each timestamp, the final token sequence contains a timestamped appearance block, a textual difference label, and a separately wrapped dynamic block:

$$\mathcal{X}_t = [\,\text{text}(\tau_t),\ \langle v\rangle A_t \langle/v\rangle,\ \text{text}(\tau_{t-1}, \tau_t),\ \langle v\rangle U_t \langle/v\rangle\,]. \tag{7}$$

The difference label states "Difference of τ_{t-1}s frame and τ_ts frame". And the first label uses τ_0 for both endpoints. Equation 7 uses shorthand wrappers for the native vision-start and vision-end tokens. Visual tokens are ordered appearance first, dynamic second, at every frame and every deepstack level. Intermediate features are injected at corresponding visual positions via inherited deepstack connections. Dynamic tokens reuse the multimodal rotary position coordinates of the appearance tokens at the same block and spatial location. This preserves spatial alignment between the two streams while identifying the pair of times represented by the dynamic input.

3.3 Video State Change Recognition Dataset Synthesis

We construct VideoSCR by combining original Kinetics-GEB+ training annotations (Wang et al., 2022) with synthetic state-change annotations in the same boundary-centered format, totaling approximately 48K entries including approximately 20K synthetic entries for Stage 1 training. The synthesis pipeline samples short clips, annotates visible state transitions, validates structured responses, and exports subject-conditioned before and after descriptions. Each annotator receives only a clip and its duration without source annotations or hinted boundaries and identifies up to three visually supported changes, each specified by a clip-relative timestamp, subject, states before and after the transition, and change category. Annotation instructions exclude camera effects and inferred intentions, while automatic checks enforce response schema, valid timestamps, explicit subjects, and distinct before and after descriptions. Appendix B provides the detailed procedure.

3.4 Training Recipe

Stage 1: state-change recognition. Stage 1 uses the VideoSCR constructed from Section 3.3. Each example contains a video, a target boundary b, a subject description c, and descriptions r^- and r^+ of the subject's state immediately before and after b.The supplied timestamp indicates where to inspect rather than a target boundary predicted in this stage, while the objective connects local change evidence to language describing its semantic content.

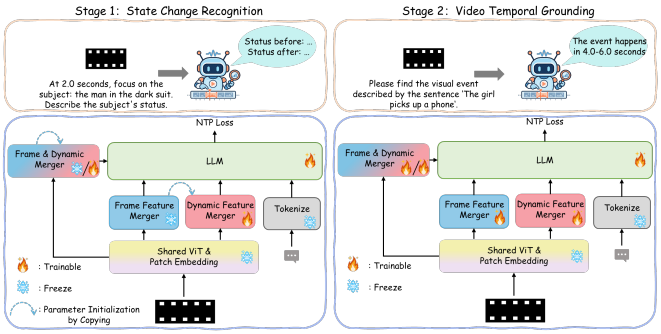


Figure 4: **Two-stage training.** Stage 1 trains the dynamic mergers and language model on VideoSCR to learn boundary-centered state descriptions, while Stage 2 fine-tunes the model on TimeLens-100k for video temporal grounding.

We freeze the shared ViT parameters θ and appearance mergers M_a^ℓ, and optimize all dynamic mergers M_d^ℓ together with the language model and output head, where each M_d^ℓ is initialized from the corresponding pretrained frame-feature merger. Keeping the pretrained appearance mapping fixed gives the dynamic interface a stable complementary stream while the language model learns the state-description task. Stage 1 uses the standard teacher-forced next-token prediction loss over the supervised response positions $\mathcal{A}$:

$$\mathcal{L}_{\text{NTP}} = -\frac{1}{|\mathcal{A}|} \sum\nolimits_{k\in\mathcal{A}} \log p_{\phi, M_a, M_d}(y_k \mid \mathcal{X}(V), q_s, y_{<k}), \tag{8}$$

where q_s is the stage-specific instruction. User instructions and visual placeholders are excluded from the response loss.

Stage 2: temporal grounding. We initialize the entire model from Stage 1 and train on TimeLens-100k (Zhang et al., 2026). Given V and q, the model generates "The event happens in t_s – t_e

Table 1: **Comparison with state-of-the-art methods on TimeLens-Bench.** We report R1@0.3, R1@0.5, R1@0.7, and mIoU (%) on Charades-TimeLens and ActivityNet-TimeLens. Per metric, red, blue, and green denote the first, second, and third best results, respectively.

Method	Source	Charades-TimeLens				ActivityNet-TimeLens			
		R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU
Closed-source models									
GPT-5 (OpenAI, 2025)	-	59.3	42.0	22.0	40.5	57.4	44.9	30.4	42.9
Gemini-2.5-Pro (Comanici et al., 2025)	-	74.1	61.1	34.0	52.8	72.3	64.2	47.1	58.1
Open-source models									
Qwen3-VL-2B (Bai et al., 2025)	arXiv'25	59.6	42.9	22.1	41.2	51.2	39.5	24.2	37.6
InternVL3.5-4B (Wang et al., 2025b)	arXiv'25	25.1	13.4	5.4	16.6	22.3	15.4	11.6	17.6
Qwen3-VL-4B (Bai et al., 2025)	arXiv'25	70.4	54.6	28.4	48.8	64.7	53.5	35.9	48.6
InternVL3.5-8B (Wang et al., 2025b)	arXiv'25	32.2	17.9	7.0	21.2	26.2	17.5	12.4	20.1
Qwen3-VL-8B (Bai et al., 2025)	arXiv'25	69.2	53.6	27.6	48.3	62.2	51.3	34.5	46.8
Grounding specialists									
TRACE-7B (Guo et al., 2025a)	ICLR'25	37.2	21.8	9.6	27.1	43.4	33.9	22.0	32.7
Grounded-VideoLLM (Wang et al., 2025a)	EMNLP'25	43.3	28.7	13.5	30.0	39.2	29.6	19.5	30.0
VideoChat-R1-7B (Li et al., 2025)	arXiv'25	51.9	30.8	11.7	33.7	35.0	23.9	11.3	25.0
Time-R1-7B (Wang et al., 2025c)	NeurIPS'25	57.9	32.0	16.9	36.6	44.8	31.0	19.0	33.1
UniTime-7B (Li et al., 2026b)	NeurIPS'25	63.1	36.8	17.3	41.3	35.6	25.2	15.8	26.8
VideoChat-Flash-7B (Li et al., 2026a)	ICLR'26	60.2	37.9	17.8	39.7	35.5	21.8	10.5	24.8
TimeLens-7B (Zhang et al., 2026)	CVPR'26	70.5	55.6	28.4	48.8	62.8	51.0	32.6	46.2
TimeLens-8B (Zhang et al., 2026)	CVPR'26	76.6	63.0	35.2	55.2	68.9	58.4	40.6	53.2
DynaSense-2B (ours)	-	69.8	56.1	28.8	49.0	57.8	48.3	31.8	44.6
DynaSense-8B (ours)	-	75.1	62.0	32.6	53.1	63.7	55.0	37.0	49.9

seconds." Training annotations with multiple intervals are serialized as a list of intervals, while the primary evaluation described below scores the first predicted interval. We unfreeze the appearance mergers while keeping the visual backbone frozen, optimizing $\{M_a^\ell, M_d^\ell, \phi\}$. Stage 2 uses the same NTP loss in Eq. 8 for query-conditioned interval generation.

4 Experiments

4.1 Experimental Setups

Training data. Stage 1 uses VideoSCR described in Section 3.3, where each retained boundary provides a subject-conditioned before/after example to teach the model to recognize fine-grained state changes from dynamic visual evidence. Stage 2 uses TimeLens-100k to transfer this change-aware representation to query-conditioned temporal grounding, enabling the model to associate dynamic changes with natural-language queries and corresponding temporal intervals.

Evaluation data and metrics. We evaluate on TimeLens-Bench, comprising Charades-TimeLens, ActivityNet-TimeLens, and QVHighlights-TimeLens (Zhang et al., 2026). For each query, temporal IoU is the intersection length divided by the union length between ground-truth interval s and first parsed predicted interval $\hat{s}$. We report R@1 at IoU thresholds of 0.3, 0.5, and 0.7 with mean IoU in percent, assigning zero overlap to missing or invalid predictions. We further evaluate zero-shot generalization on five temporal grounding tasks in E.T.Bench, namely TVG, EPM, TAL, EVS, and VHD, measured by F1 scores with $\mathrm{F1}_{\mathrm{gnd}}$ as their mean.

Implementation. We build DynaSense upon Qwen3-VL-2B-Instruct and Qwen3-VL-8B-Instruct (Bai et al., 2025) to enhance their temporal grounding capabilities. We use Qwen3-VL-2B and Qwen3-VL-8B as baselines, with ablation studies conducted on the 2B scale. Local patch matching uses a 3×3 window. Videos are sampled at 2 fps. Both training stages use a learning rate of 10^{-5} for one epoch with a global batch size of 128, optimized with AdamW (Loshchilov & Hutter, 2019) under a cosine schedule with a 0.03 warmup ratio and a weight decay of 0.1. Training uses DeepSpeed ZeRO-3 (Rajbhandari et al., 2020). All experiments are conducted on 8 NVIDIA H100 GPUs. See more details in appendix A.

4.2 Comparison with Video-MLLMs on Video Temporal Grounding

DynaSense for TimeLens-Bench. As detailed in Table 1, DynaSense outperforms the corresponding Qwen3-VL baseline on all eight metrics at both scales. At 2B, DynaSense gains 7.8% mIoU to 49.0% on Charades-TimeLens and 7.0% to 44.6% on ActivityNet-TimeLens, with R1@0.7 gains of 6.7% to 28.8% and 7.6% to 31.8% on the two subsets. At 8B, it gains 4.8% mIoU to 53.1%

Table 2: **Zero-shot generalization on E.T.Bench.** We report F1 scores (%) on five temporal grounding tasks, with $\mathrm{F1}_{\mathrm{gnd}}$ denoting their mean. The E.T.Instruct column indicates whether a model is fine-tuned on E.T.Instruct. Per metric, red, blue, and green denote the first, second, and third best results across all listed models, respectively.

Method	Source	Fine-tuned on E.T.Instruct	Grounding					
			$\mathrm{F1}_{\mathrm{TVG}}$	$\mathrm{F1}_{\mathrm{EPM}}$	$\mathrm{F1}_{\mathrm{TAL}}$	$\mathrm{F1}_{\mathrm{EVS}}$	$\mathrm{F1}_{\mathrm{VHD}}$	$\mathrm{F1}_{\mathrm{gnd}}$
VTimeLLM (Huang et al., 2024a)	CVPR'24	×	7.6	1.9	18.2	15.9	28.9	14.5
VTG-LLM (Guo et al., 2025b)	AAAI'25	×	15.9	3.7	14.4	26.8	48.2	21.8
TimeChat (Ren et al., 2024)	CVPR'24	×	26.2	3.9	10.1	29.1	40.5	22.0
TRACE (Guo et al., 2025a)	ICLR'25	×	44.3	11.1	19.1	27.4	65.9	33.6
E.T.Chat-7B (Liu et al., 2024b)	NeurIPS'24	✓	38.6	10.2	30.8	25.4	62.5	33.5
MeCo-7B (Pang et al., 2026)	ICLR'26	✓	59.0	17.5	34.2	35.5	67.9	42.8
Qwen3-VL-8B (Bai et al., 2025)	arXiv'25	×	55.0	8.8	38.3	13.0	63.2	35.7
TimeLens-8B (Zhang et al., 2026)	CVPR'26	×	56.2	23.7	38.8	11.1	53.1	36.6
DynaSense-8B (ours)	–	×	51.8	23.8	42.8	19.6	58.1	39.2

and 3.1% to 49.9%, with R1@0.7 gains of 5.0% to 32.6% and 2.5% to 37.0%, validating dynamic visual representations for fine-grained boundary localization. Among reported open-source models, DynaSense-8B ranks second on all four metrics of each subset, while TimeLens-8B leads by 2.1% and 3.3% mIoU on Charades-TimeLens and ActivityNet-TimeLens. This comparison involves different training paradigms. TimeLens emphasizes high-quality data curation and thinking-free RLVR with a tailored recipe (Zhang et al., 2026), whereas we use TimeLens-100K with two-stage SFT, first learning visual state changes and then query-conditioned localization. Our objective is dynamic visual representation and modeling for finer boundary perception, with gains over underlying Video-MLLMs as the central criterion.

DynaSense for zero-shot E.T.Bench grounding. As detailed in Tab. 2, DynaSense-8B achieves the highest mean grounding F1 among the reported models without fine-tuning on E.T.Instruct, reaching 39.2% compared with 35.7% for Qwen3-VL-8B and 36.6% for TimeLens-8B. The gains over Qwen3-VL-8B are concentrated on EPM, TAL, and EVS, where scores increase from 8.8% to 23.8%, from 38.3% to 42.8%, and from 13.0% to 19.6%, respectively. Relative to TimeLens-8B, DynaSense-8B improves TAL by 4.0 percentage points and EVS by 8.5 percentage points, while EPM is nearly unchanged (23.8% versus 23.7%). Across all listed models, including those fine-tuned on E.T.Instruct, DynaSense-8B ranks first on EPM and TAL and second in mean grounding F1. These results support transfer to several temporal grounding tasks beyond the training setting, with the largest baseline improvement on EPM.

4.3 Ablation Studies

In this section, all ablations use DynaSense-2B trained on VideoSCR-28K and TimeLens-20K, reporting R1 and mIoU on Charades-TimeLens.

Effect of our method. Table 3 disentangles state-change supervision from the dynamic representation. VideoSCR alone does not surpass the baseline, while the combination of VideoSCR with DynaSense improves R1@0.5 by 2.5% and mIoU by 1.7%, indicating that the gains arise from the joint effect of VideoSCR and dynamic representation method rather than from the data alone.

Effect of matching window size. Table 4 varies the local matching window $w \in 3, 5, 7, 9$ while keeping all remaining settings fixed. Performance peaks at $w = 3$ and declines with larger windows, with R1@0.3 dropping by 1.7% and mIoU dropping by 0.8% at $w = 7$, indicating that a compact neighborhood captures sufficient displacement, while larger ones introduce spurious matches.

Effect of matching and differencing order. Table 5 examines the effects of performing feature matching after position encoding and feature differencing before position encoding. The proposed order outperforms matching after PE by 0.4% in R1@0.3 and outperforms differencing before PE by 0.9% in R1@0.3 and 0.9% in R1@0.5, which supports governing matching by semantic features and incorporating positional information at the differencing step.

Effect of dual-merger design. Table 6 compares independent appearance and dynamic mergers with a shared merger. Independent mergers outperform the shared variant by 2.0% in R1@0.3, 2.7% in R1@0.5, and 1.3% in mIoU, which indicates that the dynamic stream requires a dedicated projection into the language space.

Table 3: **Ablation of our method.** DynaSense denotes our proposed video dynamic representation method, which consists of Video Dynamic Feature Matching, Spatial-Aware Dynamic Representation, and Unified Visual Encoding.

VideoSCR	DynaSense	Charades-TimeLens			
		R1@0.3	R1@0.5	R1@0.7	mIoU
×	×	66.2	51.7	26.2	45.9
✓	×	66.0	50.5	25.6	45.7
✓	✓	68.1	54.2	27.5	47.6

Table 4: **Ablation of matching window size.** The matching window size defines the spatial neighborhood in the previous frame centered at the position corresponding to each patch in the current frame.

Window w	Charades-TimeLens			
	R1@0.3	R1@0.5	R1@0.7	mIoU
3	68.1	54.2	27.5	47.6
5	66.9	52.5	27.0	46.9
7	66.4	52.4	27.3	46.8
9	66.8	53.0	27.1	47.0

Table 5: **Ablation of matching and differencing order.** * denotes Qwen3VL-2B-based models trained on VideoSCR-28K and TimeLens-20K. PE denotes positional encoding.

Method	Charades-TimeLens			
	R1@0.3	R1@0.5	R1@0.7	mIoU
DynaSense*	68.1	54.2	27.5	47.6
Matching after PE	67.7	54.1	27.8	47.6
Differencing before PE	67.2	53.3	27.9	47.6

Table 6: **Ablation of merger design.** Shared merger indicates that appearance and dynamic features are processed by the same merger module.

Shared merger	Charades-TimeLens			
	R1@0.3	R1@0.5	R1@0.7	mIoU
×	68.1	54.2	27.5	47.6
✓	66.1	51.5	26.5	46.3



Figure 5: **Local patch-matching regions across sampled frames.** From left to right, the overlays show a person changing posture and extending an arm. Red marks current patches whose best local match in the preceding temporal block has a different spatial grid index, while blue marks patches matched at the same index. The colors indicate this correspondence-based classification, not annotated motion or localization predictions.

4.4 Qualitative Analysis and Visualization

Fig. 5 visualizes the patch correspondences used to construct dynamic features in a sequence of sampled video frames. For each patch in the current temporal block, we find its best local match in the preceding block. A patch is colored red on the RGB frames when the selected reference has a different spatial grid index and blue when the indices agree. As the person changes posture and extends an arm, many red patches follow the changing body region, while much of the surrounding scene remains blue. Scattered red patches also occur outside the person, and some patches on the person remain blue. The overlay illustrates where local correspondences shift spatially, and it is neither a ground-truth motion mask nor a feature-importance map nor evidence of improved temporal grounding accuracy. Appendix C presents prediction results from Qwen3VL and DynaSense.

5 Conclusion

We present DynaSense, a Video-MLLM for video temporal grounding that explicitly models dynamic visual changes by establishing local patch correspondences between consecutive frames in a position-free embedding space and constructing dynamic features through position-enriched differencing, capturing both content variations and spatial displacement cues. To enhance fine-grained temporal perception, we construct VideoSCR and adopt a two-stage training strategy based on state-change recognition; extensive experiments across multiple VTG benchmarks show that DynaSense consistently improves temporal grounding over the underlying Video-MLLM baselines and generalizes effectively to unseen grounding tasks. Despite these gains, the current design relies on local correspondence between consecutive frames and may be less effective for long-range temporal dependencies or large displacements, which motivates more flexible dynamic modeling in future work.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, et al. Qwen3-VL Technical Report, 2025. URL https://arxiv.org/abs/2511.21631.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is Space-Time Attention All You Need for Video Understanding? In *International Conference on Machine Learning*, pp. 813–824, 2021. URL https://proceedings.mlr.press/v139/bertasius21a.html.

Hakan Bilen, Basura Fernando, Efstratios Gavves, Andrea Vedaldi, and Stephen Gould. Dynamic Image Networks for Action Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3034–3042, 2016. URL https://openaccess.thecvf.com/content_cvpr_2016/html/Bilen_Dynamic_Image_Networks_CVPR_2016_paper.html.

Shimin Chen, Xiaohan Lan, Yitian Yuan, et al. TimeMarker: A Versatile Video-LLM for Long and Short Video Understanding with Superior Temporal Localization Ability, 2024. URL https://arxiv.org/abs/2411.18211.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities. *arXiv preprint arXiv:2507.06261*, 2025. doi: 10.48550/arXiv.2507.06261. URL https://arxiv.org/abs/2507.06261.

Andong Deng, Zhongpai Gao, Anwesa Choudhuri, Benjamin Planche, Meng Zheng, Bin Wang, Terrence Chen, Chen Chen, and Ziyan Wu. Seq2Time: Sequential Knowledge Transfer for Video LLM Temporal Grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13766–13775, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Deng_Seq2Time_Sequential_Knowledge_Transfer_for_Video_LLM_Temporal_Grounding_CVPR_2025_paper.html.

Rong Fan, Kaiyan Xiao, Minghao Zhu, et al. GroundVTS: Visual Token Sampling in Multimodal Large Language Models for Video Temporal Grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10408–10418, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Fan_GroundVTS_Visual_Token_Sampling_in_Multimodal_Large_Language_Models_for_CVPR_2026_paper.html.

Glint Lab, AIM for Health Lab, and MVP Lab. LLaVA-OneVision-2: Towards Next-Generation Perceptual Intelligence, 2026. URL https://arxiv.org/abs/2605.25979.

Google DeepMind. Gemini 3.1 Flash-Lite Model Card, 2026. URL https://deepmind.google/models/model-cards/gemini-3-1-flash-lite/. Model card, March 3, 2026.

Chaohong Guo, Yihan He, Yongwei Nie, et al. T2SGrid: Temporal-to-Spatial Gridification for Video Temporal Grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3443–3454, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Guo_T2SGrid_Temporal-to-Spatial_Gridification_for_Video_Temporal_Grounding_CVPR_2026_paper.html.

Yongxin Guo, Jingyu Liu, Mingda Li, et al. TRACE: Temporal Grounding Video LLM via Causal Event Modeling. In *International Conference on Learning Representations*, pp. 89594–89618, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/df027cf11469e746ef94d583f9f5537f-Abstract-Conference.html.

Yongxin Guo, Jingyu Liu, Mingda Li, et al. VTG-LLM: Integrating Timestamp Knowledge into Video LLMs for Enhanced Video Temporal Grounding. In *Proceedings of the AAAI conference on artificial intelligence*, pp. 3302–3310, 2025b. doi: 10.1609/aaai.v39i3.32341. URL https://ojs.aaai.org/index.php/AAAI/article/view/32341.

Haowen Hou, Zhen Huang, Zheming Liang, et al. AdaCodec: A Predictive Visual Code for Video MLLMs, 2026. URL https://arxiv.org/abs/2606.02569.

Bin Huang, Xin Wang, Hong Chen, et al. VTimeLLM: Empower LLM to Grasp Video Moments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14271–14280, 2024a. doi: 10.1109/CVPR52733.2024.01353. URL https://openaccess.thecvf.com/content/CVPR2024/html/Huang_VTimeLLM_Empower_LLM_to_Grasp_Video_Moments_CVPR_2024_paper.html.

De-An Huang, Shijia Liao, Subhashree Radhakrishnan, Hongxu Yin, Pavlo Molchanov, Zhiding Yu, and Jan Kautz. LITA: Language Instructed Temporal-Localization Assistant. In *European Conference on Computer Vision*, pp. 202–218, 2024b. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/8090_ECCV_2024_paper.php.

Intelligent Editing Team. Vidi: Large Multimodal Models for Video Understanding and Editing, 2025. URL https://arxiv.org/abs/2504.15681.

Yang Jin, Zhicheng Sun, Kun Xu, et al. Video-LaVIT: Unified Video-Language Pre-training with Decoupled Visual-Motional Tokenization. In *International Conference on Machine Learning*, pp. 22185–22209, 2024. URL https://proceedings.mlr.press/v235/jin24f.html.

Heeseung Kwon, Manjin Kim, Suha Kwak, and Minsu Cho. MotionSqueeze: Neural Motion Feature Learning for Video Understanding. In *European Conference on Computer Vision*, 2020. doi: 10.48550/arXiv.2007.09933. URL https://arxiv.org/abs/2007.09933.

Jie Lei, Tamara L. Berg, and Mohit Bansal. QVHighlights: Detecting Moments and Highlights in Videos via Natural Language Queries. In *Advances in Neural Information Processing Systems*, pp. 11846–11858, 2021. doi: 10.48550/ARXIV.2107.09609. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/62e0973455fd26eb03e91d5741a4a3bb-Paper.pdf.

Congcong Li, Xinyao Wang, Longyin Wen, Dexiang Hong, Tiejian Luo, and Libo Zhang. End-to-End Compressed Video Representation Learning for Generic Event Boundary Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13967–13976, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Li_End-to-End_Compressed_Video_Representation_Learning_for_Generic_Event_Boundary_Detection_CVPR_2022_paper.html.

Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yinan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. VideoChat-R1: Enhancing Spatio-Temporal Perception via Reinforcement Fine-Tuning. *arXiv preprint arXiv:2504.06958*, 2025. doi: 10.48550/arXiv.2504.06958. URL https://arxiv.org/abs/2504.06958.

Xinhao Li, Yi Wang, Jiashuo Yu, et al. VideoChat-Flash: Hierarchical Compression for Long-Context Video Modeling. In *International Conference on Learning Representations*, volume 2026, pp. 109089–109117, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/b14d7175755b180dc2163e15e3110cb6-Abstract-Conference.html.

Yan Li, Bin Ji, Xintian Shi, et al. TEA: Temporal Excitation and Aggregation for Action Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 906–915, 2020. doi: 10.1109/CVPR42600.2020.00099. URL https://openaccess.thecvf.com/content_CVPR_2020/supplemental/Li_TEA_Temporal_Excitation_CVPR_2020_supplemental.pdf.

Zeqian Li, Shangzhe Di, Zhonghua Zhai, et al. Universal Video Temporal Grounding with Generative Multi-modal Large Language Models. In *Advances in Neural Information Processing Systems*, pp. 64426–64455, 2026b. doi: 10.52202/085713-2159. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/5d2e24df9cfaad3189833b819c40b392-Abstract-Conference.html.

Kevin Qinghong Lin, Pengchuan Zhang, Joya Chen, et al. UniVTG: Towards Unified Video-Language Temporal Grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2782–2792, 2023. doi: 10.1109/iccv51070.2023.00262. URL https://openaccess.thecvf.com/content/ICCV2023/html/Lin_UniVTG_Towards_Unified_Video-Language_Temporal_Grounding_ICCV_2023_paper.html.

Ruyang Liu, Chen Li, Haoran Tang, et al. ST-LLM: Large Language Models Are Effective Temporal Learners. In *European Conference on Computer Vision*, pp. 1–18, 2024a. doi: 10.1007/978-3-031-72998-0_1. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/papers/07364.pdf.

Yang Liu, Qianqian Xu, Peisong Wen, Siran Dai, and Qingming Huang. When the Future Becomes the Past: Taming Temporal Correspondence for Self-supervised Video Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24033–24044, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Liu_When_the_Future_Becomes_the_Past_Taming_Temporal_Correspondence_for_CVPR_2025_paper.html.

Ye Liu, Zongyang Ma, Zhongang Qi, et al. E.T. Bench: Towards Open-Ended Event-Level Video-Language Understanding. In *Advances in Neural Information Processing Systems*, pp. 32076–32110, 2024b. doi: 10.52202/079017-1009. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/38ace89a980ead30c6be2b46afc1a5bd-Abstract-Datasets_and_Benchmarks_Track.html.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Ming Nie, Dan Ding, Chunwei Wang, et al. SlowFocus: Enhancing Fine-grained Temporal Understanding in Video LLM. In *Advances in Neural Information Processing Systems*, 2024. doi: 10.52202/079017-2599. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/94ef721705ea95d6981632be62bb66e2-Abstract-Conference.html.

OpenAI. Introducing GPT-5, 2025. URL https://openai.com/index/introducing-gpt-5/. Available from OpenAI announcement, August 7, 2025.

Zongshang Pang, Mayu Otani, and Yuta Nakashima. Measure Twice, Cut Once: A Semantic-Oriented Approach to Video Temporal Localization with Video LLMs. In *International Conference on Learning Representations*, volume 2026, pp. 125897–125917, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/cc00f5428c68244edde498d94b7a7ccb-Abstract-Conference.html.

Shraman Pramanick, Effrosyni Mavroudi, Yale Song, et al. Enrich and Detect: Video Temporal Grounding with Multimodal LLMs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 24297–24308, 2025. doi: 10.1109/iccv51701.2025.02252. URL https://openaccess.thecvf.com/content/ICCV2025/html/Pramanick_Enrich_and_Detect_Video_Temporal_Grounding_with_Multimodal_LLMs_ICCV_2025_paper.html.

Viorica Pătrăucean, Lucas Smaira, Ankush Gupta, et al. Perception Test: A Diagnostic Benchmark for Multimodal Video Models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 42748–42761, 2023. doi: 10.52202/075280-1852. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/8540fba4abdc7f9f7a7b1cc6cd60e409-Abstract.html.

Long Qian, Juncheng Li, Yu Wu, et al. Momentor: Advancing Video Large Language Model with Fine-Grained Temporal Reasoning. In *International Conference on Machine Learning*, pp. 41340–41356, 2024. doi: 10.5555/3692070.3693750. URL https://proceedings.mlr.press/v235/qian24a.html.

Zihan Qiu, Zekun Wang, Xiao Li, et al. On the Design of Qwen3.8-Next Architecture: Evaluation, Efficiency, and Training Stability. *arXiv preprint arXiv:2608.30320*, 2026. doi: 10.48550/arXiv.2608.30320. URL https://arxiv.org/abs/2608.30320.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. ZeRO: Memory Optimizations Toward Training Trillion Parameter Models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1–16, 2020. doi: 10.1109/SC41405.2020.00024. URL https://doi.org/10.1109/SC41405.2020.00024.

Shuhuai Ren, Linli Yao, Shicheng Li, et al. TimeChat: A Time-sensitive Multimodal Large Language Model for Long Video Understanding. In *booktitle = Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,*, pp. 14313–14323, 2024. doi: 10.1109/cvpr52733.2024.01357. URL https://openaccess.thecvf.com/content/CVPR2024/html/Ren_TimeChat_A_Time-sensitive_Multimodal_Large_Language_Model_for_Long_Video_CVPR_2024_paper.html.

Sayan Deb Sarkar, Rémi Pautrat, Ondrej Miksik, et al. CoPE-VideoLM: Leveraging Codec Primitives For Efficient Video Language Modeling, 2026. URL https://arxiv.org/abs/2602.13191.

Karen Simonyan and Andrew Zisserman. Two-Stream Convolutional Networks for Action Recognition in Videos. In *Advances in Neural Information Processing Systems*, volume 27, 2014. URL https://proceedings.neurips.cc/paper_files/paper/2014/hash/ca007296a63f7d1721a2399d56363022-Abstract.html.

Haibo Wang, Zhiyang Xu, Yu Cheng, et al. Grounded-VideoLLM: Sharpening Fine-grained Temporal Grounding in Video Large Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, 2025a. doi: 10.18653/v1/2025.findings-emnlp.50. URL https://aclanthology.org/2025.findings-emnlp.50.pdf.

Limin Wang, Zhan Tong, Bin Ji, and Gangshan Wu. TDN: Temporal Difference Networks for Efficient Action Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1895–1904, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Wang_TDN_Temporal_Difference_Networks_for_Efficient_Action_Recognition_CVPR_2021_paper.html.

Shihao Wang, Guo Chen, De-An Huang, et al. VideoITG: Multimodal Video Understanding with Instructed Temporal Grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24640–24650, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Wang_VideoITG_Multimodal_Video_Understanding_with_Instructed_Temporal_Grounding_CVPR_2026_paper.html.

Weiyun Wang, Zhangwei Gao, Lixin Gu, et al. InternVL3.5: Advancing Open-Source Multimodal Models in Versatility, Reasoning, and Efficiency. *arXiv preprint arXiv:2508.18265*, 2025b. doi: 10.48550/arXiv.2508.18265. URL https://arxiv.org/abs/2508.18265.

Ye Wang, Ziheng Wang, Boshen Xu, et al. Time-R1: Post-Training Large Vision Language Model for Temporal Video Grounding. In *Advances in Neural Information Processing Systems*, volume 38, pp. 83330–83364, 2025c. doi: 10.52202/085713-2793. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/7801b29c93b599b8d0c44138596bdeed-Abstract-Conference.html.

Yueqian Wang, Xiaojun Meng, Jianxin Liang, et al. HawkEye: Training Video-Text LLMs for Grounding Text in Videos, 2024. URL https://arxiv.org/abs/2403.10228.

Yuxuan Wang, Difei Gao, Licheng Yu, et al. GEB+: A Benchmark for Generic Event Boundary Captioning, Grounding and Retrieval. In *European Conference on Computer Vision*, pp. 709–725, 2022. doi: 10.1007/978-3-031-19833-5_41. URL https://www.ecva.net/papers/eccv_2022/papers_ECCV/papers/136950703.pdf.

Yongliang Wu, Xinting Hu, Yuyang Sun, et al. Number it: Temporal Grounding Videos like Flipping Manga. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13754–13765, 2025. doi: 10.1109/cvpr52734.2025.01284. URL https://arxiv.org/abs/2411.10332.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. NExT-QA: Next Phase of Question-Answering to Explaining Temporal Actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9777–9786, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Xiao_NExT-QA_Next_Phase_of_Question-Answering_to_Explaining_Temporal_Actions_CVPR_2021_paper.html.

Zuhao Yang, Yingchen Yu, Yunqing Zhao, Shijian Lu, and Song Bai. TimeExpert: An Expert-Guided Video LLM for Video Temporal Grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 24286–24296, 2025. doi: 10.1109/ICCV51701.2025.02251.

Daichi Yashima, Shuhei Kurita, Yusuke Oda, and Komei Sugiura. Remora: Multimodal large language model based on refined motion representation for long-video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31845–31855, June 2026.

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 *The Thirteenth International Conference on Learning Representations*, 2025a.

Yingsen Zeng, Zepeng Huang, Yujie Zhong, Chengjian Feng, Jie Hu, Lin Ma, and Yang Liu. Distime: Distribution-based time representation for video large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21961–21971, October 2025b.

Jinglei Zhang, Yuanfan Guo, Rolandos Alexandros Potamias, Jiankang Deng, Hang Xu, and Chao Ma. VTimeCoT: Thinking by Drawing for Video Temporal Grounding and Reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 24203–24213, 2025a. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhang_VTimeCoT_Thinking_by_Drawing_for_Video_Temporal_Grounding_and_Reasoning_ICCV_2025_paper.html.

Jun Zhang, Teng Wang, Yuying Ge, et al. TimeLens: Rethinking Video Temporal Grounding with Multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10419–10429, 2026. doi: 10.48550/ARXIV.2512.14698. URL https://openaccess.thecvf.com/content/CVPR2026/html/Zhang_TimeLens_Rethinking_Video_Temporal_Grounding_with_Multimodal_LLMs_CVPR_2026_paper.html.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. LLaVA-Video: Video Instruction Tuning With Synthetic Data. *Transactions on Machine Learning Research*, 2025b. doi: 10.48550/arXiv.2410.02713. URL https://openreview.net/forum?id=EElFGvt39K.

Henghao Zhao, Ge-Peng Ji, Rui Yan, et al. VideoExpert: Augmented LLM for Temporal-Sensitive Video Understanding. *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 5893–5908, 2026. doi: 10.1109/tcsvt.2026.3653742. URL https://ieeexplore.ieee.org/document/11348931.

Zijia Zhao, Yuqi Huo, Tongtian Yue, et al. Efficient Motion-Aware Video MLLM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24159–24168, 2025. doi: 10.1109/cvpr52734.2025.02250. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zhao_Efficient_Motion-Aware_Video_MLLM_CVPR_2025_paper.html.

Minghang Zheng, Zihao Yin, Yi Yang, Yuxin Peng, and Yang Liu. OmniVTG: A Large-Scale Dataset and Training Paradigm for Open-World Video Temporal Grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24620–24629, 2026a. URL https://openaccess.thecvf.com/content/CVPR2026/html/Zheng_OmniVTG_A_Large-Scale_Dataset_and_Training_Paradigm_for_Open-World_Video_CVPR_2026_paper.html.

Zelin Zheng, Xinyan Liu, Ruixin Li, et al. Foresee-to-Ground: From Predictive Temporal Perception to Evidence-Driven Reasoning for Video Temporal Grounding. In *International Conference on Machine Learning*, 2026b. doi: 10.48550/ARXIV.2605.21973. URL https://arxiv.org/abs/2605.21973.

APPENDIX

A TRAINING AND MODEL HYPERPARAMETERS

Table 7 summarizes the default configuration of the two-stage training recipe for DynaSense-8B. Stage 1 learns state-change descriptions, and Stage 2 initializes from the Stage 1 checkpoint for temporal grounding. The shared visual backbone remains frozen throughout both stages.

Table 7: **Training and model hyperparameters for DynaSense-8B.** Values correspond to the default 8B training script. Shared settings apply to both stages.

Hyperparameter	Stage 1	Stage 2
Model and visual input		
Base model	Qwen3-VL-8B-Instruct	
Local matching window	3×3	
Matching chunk size	1,024 patches	
Video sampling rate	2 fps	
Maximum sampled frames	448	
Minimum visual token setting	64	
Total visual token budget	28,672	
Maximum sequence length	51,200 tokens	
Optimization		
Training data	VideoSCR	TimeLens-100K
Training epochs	1	1
Optimizer	AdamW	
LLM and merger learning rate	10^{-5}	10^{-5}
Learning-rate schedule	Cosine	
Warmup ratio	0.03	
Weight decay	0.1	
Global batch size	128	
Batch size per device	1	
Number of devices	8	
Gradient accumulation steps	16	
Random seed	42	
Trainable modules and numerical settings		
Shared visual backbone	Frozen	Frozen
Appearance mergers	Frozen	Trainable
Dynamic mergers	Trainable	Trainable
Language model and output head	Trainable	Trainable
LoRA	Disabled	
Training precision	BF16	
Distributed optimization	DeepSpeed ZeRO-3	
Gradient checkpointing	Enabled	

The frame cap is computed as $28{,}672/64 = 448$ in the default configuration. The global batch size is $8 \times 1 \times 16 = 128$. The visual token budget is specified jointly for the appearance and dynamic branches, while the maximum sequence length also accommodates the textual input and target tokens.

B State-Change Data Synthesis

This appendix details the synthesis pipeline introduced in Section 3.3. Fig. 6 presents the overall workflow. The implementation details below describe the current synthesis configuration, where sampling settings are configuration parameters rather than statistics of the completed collection.

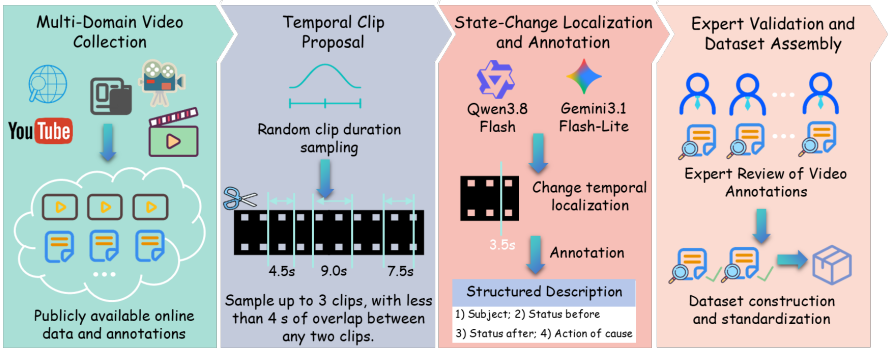


Figure 6: **State-change data synthesis.** Source videos are sampled into short clips, annotated for visible state transitions, checked, and assembled into a Kinetics-GEB+-compatible collection (Wang et al., 2022). The diagram includes two annotator options and an expert-review stage. The current executable configuration uses Qwen3.8-Flash and automatic validation, while Gemini usage and completed expert review require separate run records.

B.1 Source Videos and Clip Sampling

The source configuration indexes video collections from Perception (Pătrăucean et al., 2023), NExT-QA (Xiao et al., 2021), and LLaVA-Video (Zhang et al., 2025b). For each source video, the sampler derives a deterministic random seed from the global seed (42) and the video identifier. It samples up to three clips, drawing durations from a truncated Gaussian and start times uniformly over the valid interval of the source video. The current configuration uses a mean duration of 8 seconds, a standard deviation of 2 seconds, and duration bounds of 4–12 seconds, further constrained by the source-video duration. A candidate clip is retained only when its overlap with each previously selected clip is below 4 seconds. Sampling is limited to 200 attempts per video.

B.2 Visual Annotation and Output Format

The annotation prompt supplies only the clip and its duration, without source captions, source boundary annotations, or a proposed transition time. The current configuration uses Qwen3.8-Flash with video input sampled at 2 fps, a temperature of 0.1, and a maximum output length of 768 tokens. The model first decides whether the clip contains a clear visible change and then returns at most three transitions in a structured JSON response. For architectural context, the Qwen3.8-Next report describes Qwen3.8-Flash-Next (Qiu et al., 2026).

Each transition contains a timestamp strictly inside the clip, a specific subject description, the subject's visible states immediately before and after the boundary, and one of five Kinetics-GEB+ categories: change of action, change of color, change of object being interacted with, change of subject, or multiple changes. An optional causal-action field is populated only when the action is directly visible. All descriptions are requested in concise English. The prompt excludes changes caused solely by camera cuts, subtitles, blur, zoom, or viewpoint, and disallows inferred identities, intentions, emotions, and unseen causes. Clips without a reliable transition receive an empty change list and a rejection reason.

B.3 Automatic Validation and Review Protocol

Automatic validation first parses the response as a single JSON object and checks the required fields, types, allowed labels, and maximum number of changes. Retained clips must contain at least one

transition and no rejection reason, while discarded clips must provide a rejection reason and an empty change list. Timestamps must lie within the clip, and duplicate timestamps are rejected after rounding to millisecond precision.

Text checks reject bare-pronoun subjects, refusal or uncertainty phrases, Markdown markup, and unwanted explanatory prefixes. Before/after descriptions must differ after case, punctuation, and whitespace normalization. Accepted text is normalized and transitions are sorted by timestamp. These checks enforce structural and textual consistency, while visual correctness is addressed in the expert-review stage depicted in Fig. 6.

The review protocol calls for two independent reviewers to assess transition observability, temporal alignment, subject specificity, and consistency of the before/after descriptions, with disagreements adjudicated by a third reviewer. This describes the review procedure shown in the diagram, rather than a claim of completed review: the current exporter records no human-review results. Likewise, the executable configuration enables Qwen3.8-Flash, while the Gemini-3.1-Flash-Lite option (Google DeepMind, 2026) shown in the diagram requires separate usage records.

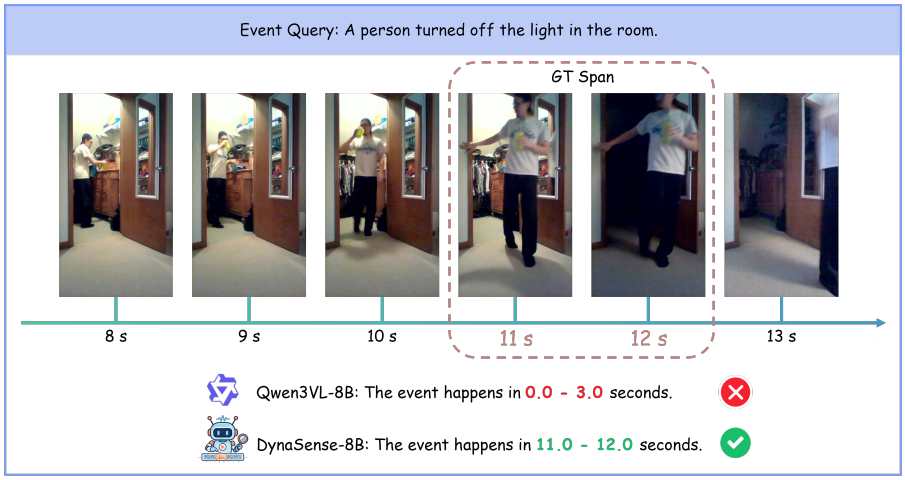
B.4 Export and Training Examples

Accepted transitions are exported in the Kinetics-GEB+ boundary-centered format. Each boundary record contains its timestamp, neighboring boundary timestamps (or clip endpoints), change label, subject, before/after descriptions, and optional causal action. The exporter also retains provenance linking the boundary to its source video, clip interval, annotation model, and prompt version. Synthetic annotations can then be merged with the original Kinetics-GEB+ training annotations for the state-change recognition stage.

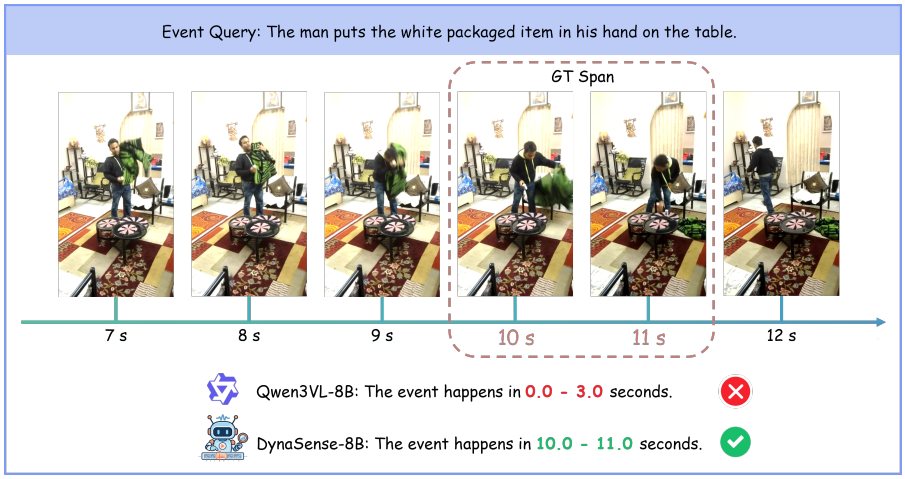
Each training example conditions on the video, a supplied boundary timestamp, and the subject, and supervises the corresponding before/after state descriptions, as defined in Section 3.4. Accepted clips and annotated boundaries are counted separately because one clip may contribute multiple boundaries. Cross-dataset video deduplication is disabled in the current synthesis configuration. Overlap checks against evaluation videos must therefore be established from separate collection records before disjointness can be asserted.

C Qualitative Inference Results

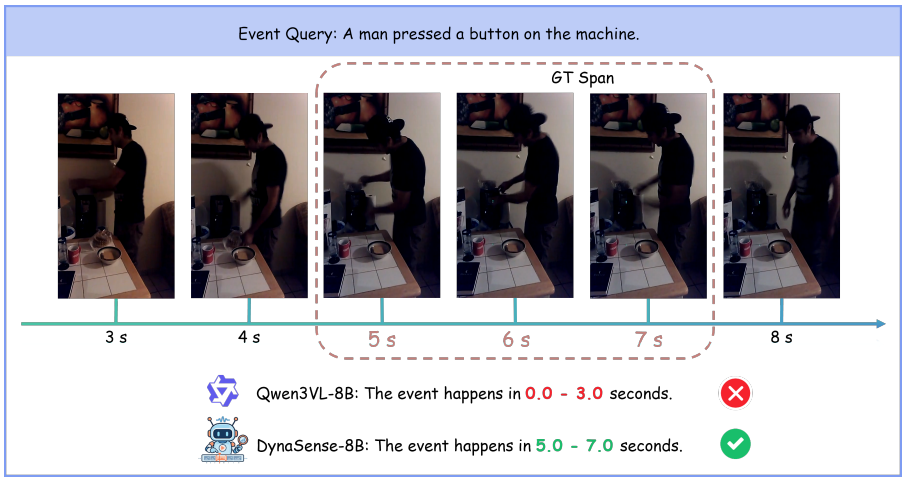
Figure 7 presents representative qualitative examples of temporal grounding by DynaSense. Each example visualizes sampled frames with the event query and the predicted intervals of DynaSense and Qwen3VL-8B. The annotated events are extremely short: the events in Figure 7a and Figure 7b span only one second, from 11.0 to 12.0 seconds and from 10.0 to 11.0 seconds, while the event in Figure 7c spans only two seconds, from 5.0 to 7.0 seconds. Such short transitions are defined by subtle state changes under nearly identical appearance, including illumination change, object placement, and a brief manual action. The appearance-based Qwen3VL-8B predicts 0.0 to 3.0 seconds in all three cases, which has no overlap with the ground truth. In contrast, DynaSense precisely recovers the annotated intervals through modeling of video dynamic representations, which illustrates the benefit of explicit dynamic evidence for fine-grained boundary localization.



(a) Turning off the light in the room.



(b) Placing a white packaged item on the table.



(c) Pressing a button on the machine.

Figure 7: **Qualitative temporal grounding results.** Representative examples of DynaSense and Qwen3VL-8B on diverse state-changing events.