PAEM: PROGRESSION-AWARE EVENT MATCHING FOR SPATIO-TEMPORAL VIDEO GROUNDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Spatio-temporal video grounding aims to localize the spatio-temporal tube corresponding to a natural-language query in an untrimmed video. Existing methods exploit query-related semantic cues to guide spatial and temporal localization. However, semantic relevance alone may be insufficient to determine the complete extent of a queried event, as salient local cues can lead to incomplete event spans, while broader context may still fail to distinguish candidates that share substantial content but differ in event structure. We argue that event progression, encompassing how an event develops over time and connects to surrounding activities, can provide a structural basis for distinguishing semantically similar candidates. In this paper, we propose **P**rogression-**A**ware **E**vent **M**atching (PAEM), which explicitly incorporates event progression into spatio-temporal matching. Specifically, PAEM combines stage-aware context matching and state-aware process matching to jointly capture contextual stage relations and internal event progression. To capture the stage relations, we conduct stage-aware context matching by aggregating query-relevant responses before, within, and after each candidate span. Meanwhile, state-aware process matching characterizes internal state dynamics through latent responses and conditional transitions. The two matching scores jointly determine the target entity and its temporal interval. Experiments across 4 vision-language backbones show m_vIoU improvements of 3.6%–$9.9\% \uparrow$ over corresponding baselines on HCSTVG-v1/v2, demonstrating PAEM's effectiveness. When transferred to related grounding tasks in a zero-shot setting, PAEM yields consistent improvements over the corresponding baselines, with gains of up to $9.4\% \uparrow$ across the primary metrics, highlighting its generalizability.

1 INTRODUCTION

Spatio-Temporal Video Grounding (STVG) aims to localize the spatio-temporal tube corresponding to a natural-language query in an untrimmed video. This fine-grained video-language understanding supports applications such as video retrieval (Meng et al., 2025; Wang et al., 2024b), intelligent surveillance (Huang, 2026), human-computer interaction (Ahmad et al., 2025), and embodied perception (Xu et al., 2025). Unlike static visual grounding, queries in STVG typically describe events involving temporally extended actions or interactions, requiring appearance, motion, and interaction cues to identify the target entity and its event interval. Accurately grounding such queries requires not only identifying query-relevant content, but also determining how this content unfolds over time.

Existing methods exploit attribute and action cues in the query to guide local spatial and temporal localization (Gu et al., 2025a; Yang et al., 2026). However, this paradigm is susceptible to `local overreliance`: the highly salient local cues used can dominate matching and draw predictions toward only the most distinctive portion of an event, overlooking its broader temporal organization and consequently truncating its complete extent. As illustrated in Fig. 1(a), the entity description "*man in a suit*" is ambiguous, requiring local action cues to identify the target. For temporal matching, "*puts it on the table*" forms a strong semantic peak, biasing the prediction toward a span centered on this action while overlooking how the queried event unfolds beyond that moment. Consequently, the predicted span may capture the most salient action but truncate the complete event.

To incorporate broader context, other methods strengthen semantic matching through global context modeling (Jin et al., 2022) and spatio-temporal feature integration (Gu et al., 2024; Liang et al.,

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.

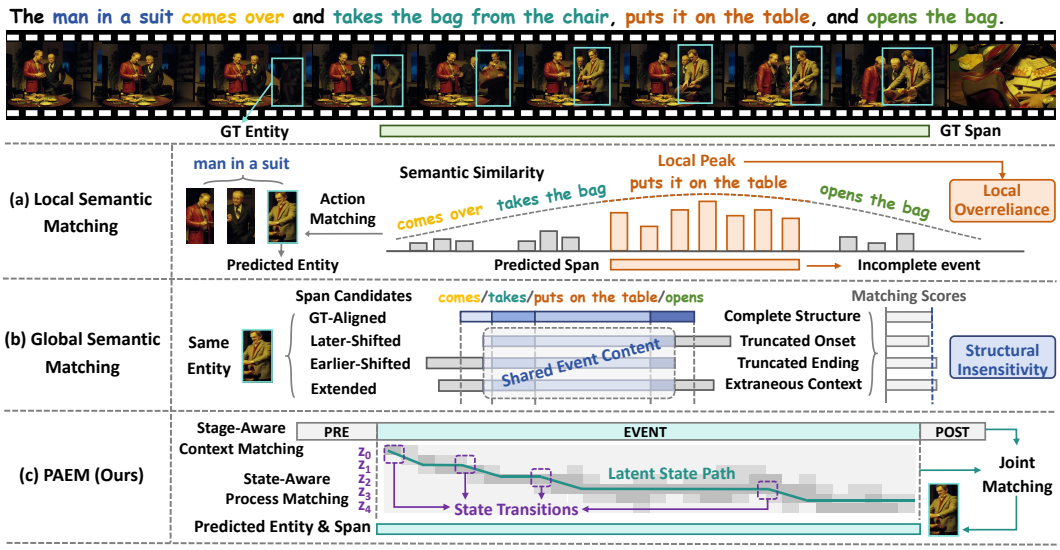


Figure 1: Comparison of matching strategies for spatio-temporal video grounding.

2024). While such context helps reduce overreliance on individual local cues, it remains insufficient to distinguish candidates that share similar semantic content but differ in temporal structure. As shown in Fig. 1(b), the GT-aligned, later-shifted, earlier-shifted, and extended candidates along the same entity trajectory share much of the content associated with "*takes the bag*", "*puts it on the table*", and "*opens the bag*", yet differ in the completeness of the progression from "*comes over*" to "*opens the bag*" and the inclusion of activities outside the queried event. Such semantically similar but structurally distinct candidates reveal the `structural insensitivity` of semantic matching to differences in event organization. Distinguishing these candidates requires examining how query-relevant content develops over time and connects to surrounding activities, rather than merely identifying shared semantic content. We term this temporal organization **event progression**.

In this paper, we propose **P**rogression-**A**ware **E**vent **M**atching (PAEM) to explicitly incorporate event progression into spatio-temporal matching. Rather than treating the queried event as a single temporal cue, PAEM characterizes it at two complementary levels: the stage structure of a candidate relative to its surrounding trajectory, and the state evolution within the candidate event itself. To support such modeling, PAEM organizes query-conditioned visual features into an entity-time lattice, preserving entity-specific temporal information across candidate spans. As illustrated in Fig. 1(c), stage-aware context matching combines entity–query relevance with PRE, EVENT, and POST responses to evaluate a candidate in relation to its surrounding activities, helping determine whether its temporal extent is structurally complete. Meanwhile, state-aware process matching captures the internal evolution of the event through latent state responses and conditional transitions, modeling how visual evidence supports state persistence and advancement over time. Together, the two matching mechanisms characterize event progression from both trajectory-level context and within-event dynamics, distinguishing candidates that share similar semantic content but differ in event structure.

Our contributions can be summarized as:

- We propose Progression-Aware Event Matching (PAEM), establishing event progression as a structural criterion for candidate selection in spatio-temporal video grounding, beyond semantic relevance alone.
- We develop a unified formulation that combines contextual stage constraints on event extent with latent path modeling of internal progression, making event structure an explicit, computable basis for candidate matching.
- Experiments across four vision-language backbones show consistent improvements over corresponding baselines, demonstrating PAEM's effectiveness in spatio-temporal video grounding. Transfer without target-task fine-tuning further supports its applicability to spatial grounding, video object segmentation, and temporal grounding.

2 Related Works

2.1 Spatio-Temporal Video Grounding

Spatio-temporal video grounding aims to localize a language-referred instance over both space and time. Early approaches such as STGRN (Zhang et al., 2020) establish video-language correspondence by modeling spatial relations and object dynamics across frames. STVGBert (Su et al., 2021) introduces a visual-linguistic Transformer for grounding, while TubeDETR (Yang et al., 2022) combines video-text encoding with joint spatio-temporal decoding. Subsequent studies place greater emphasis on consistency and interaction between spatial and temporal predictions. STCAT (Jin et al., 2022) uses a global multimodal template to maintain target consistency across frames, whereas CoSTA (Liang et al., 2024) couples spatial localization and temporal boundary prediction through an entangled decoder. More recently, SpaceVLLM (Wang et al., 2026a) introduces dedicated spatio-temporal queries and a spatial decoder to extend multimodal large language models to STVG, while STVG-o1 (Gu et al., 2025b) explores reinforcement fine-tuning for joint localization. STVG-R1 (Zhang et al., 2026) combines temporally consistent object-ID prompts with task-driven reinforcement learning for instance-level spatio-temporal grounding. Complementing these advances, our proposed PAEM makes event structure an explicit matching criterion by jointly considering stage relations along the target trajectory and latent state progression within each candidate.

2.2 Event Modeling in Video-Language Learning

Event modeling supports fine-grained video understanding by organizing visual content into semantically meaningful events and their relations. Work on VidSitu (Sadhu et al., 2021) and VideoWhisperer (Khan et al., 2022) represents events through actions, entities, and semantic roles. PDVC (Wang et al., 2021) and Vid2Seq (Yang et al., 2023b) jointly predict event intervals and descriptions, while (Zhou et al., 2024) extend event captioning to streaming inputs. For longer videos, VidChapters (Yang et al., 2023a) provides a benchmark for chapter segmentation and description, and Video ReCap (Islam et al., 2024) captures activity organization through recursive, multi-granularity captioning. Event modeling has also been incorporated into video-language alignment and grounding. EaTR (Jang et al., 2023) extracts event units to construct dynamic moment queries, while E.T. Chat (Liu et al., 2024b) combines multi-event instruction tuning with temporal embedding matching. TRACE (Guo et al., 2024) models structured event sequences, predicting timestamps, saliency scores, and descriptions conditioned on preceding events. PAEM instead focuses on latent state progression within candidate events, combining it with stage relations along the target trajectory to make event progression an explicit structural criterion for spatio-temporal matching.

3 Method

PAEM integrates stage-aware context matching and state-aware process matching over an entity-time lattice. As shown in Fig. 2, the lattice associates query-conditioned visual features with entity tubes. The two matching modules assess entity relevance and temporal stage alignment, and internal event progression through latent state paths, respectively. Their combined scores jointly determine the target entity and temporal interval.

3.1 Problem Formulation

Given an untrimmed video $V = \{v_t\}_{t=1}^{T}$ and a natural-language query Q, we extract K entity tubes $E = \{e_i\}_{i=1}^{K}$ using segmentation and tracking tools. Each tube $e_i = \{b_{i,t}\}_{t=1}^{T}$ specifies the entity's spatial region across frames. The goal is to identify the entity-span $h = (i, t_s, t_e)$ corresponding to Q, where i indexes the target tube and $1 \le t_s \le t_e \le T$ defines its temporal interval.

3.2 Entity-Time Lattice Construction

To preserve fine-grained visual information across candidate entities and temporal positions, we construct an entity-time lattice that associates query-conditioned visual features with entity tubes. For the entity axis, we directly adopt the pre-extracted entity tubes E, which provide the spatial trajectories of candidate entities. For the temporal axis, we extract query-conditioned visual context

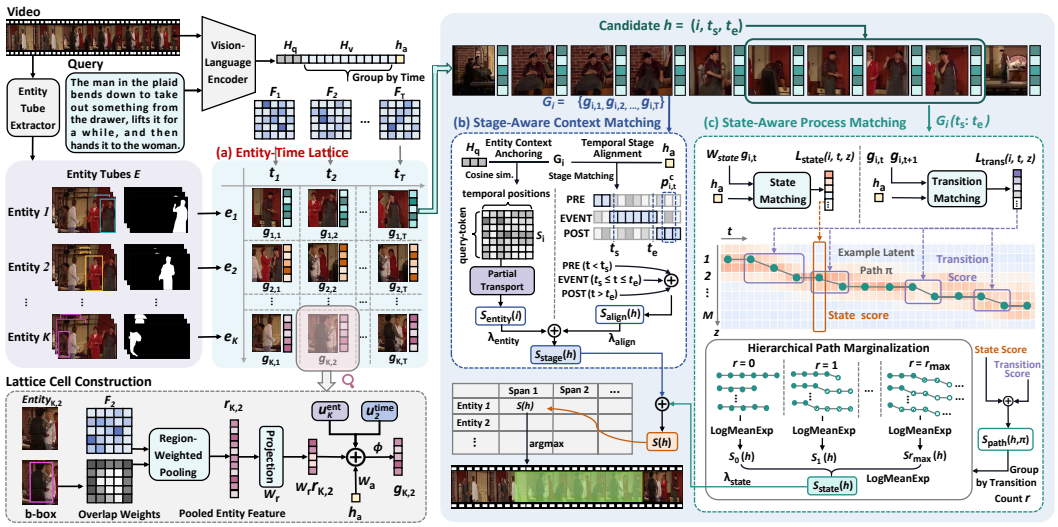


Figure 2: Overview of PAEM, including (a) the construction of the entity-time lattice, (b) the computations for stage-aware context matching, and (c) state-aware process matching. The illustrated path represents an example of latent state progression.

from the sampled video and query to obtain event-relevant representations at different temporal positions. Specifically, as shown in Fig. 2, we jointly feed V and Q into an MLLM encoder to obtain contextualized query representations H_q, visual representations H_v, and an anchor representation h_a for subsequent cell construction. At this stage, H_v retains the visual-token ordering of the MLLM sequence. We group visual tokens by sampled temporal position as $F_t = \{H_v^{(j)} \mid \pi(j) = t\}$, where $\pi(j)$ maps token j to its corresponding temporal position.

For entity i at temporal position t, we pool visual tokens in F_t according to their spatial overlap with the entity bounding box $b_{i,t}$. Each token is weighted by the fraction of its spatial support covered by the box, with weights normalized across tokens to obtain the entity-specific feature $r_{i,t}$. The resulting ROI representation is projected into a shared feature space and fused with the anchor representation h_a, entity embedding u_i^{ent}, and temporal embedding u_t^{time}:

$$g_{i,t} = \phi\big(W_r r_{i,t} + W_a h_a + u_i^{\text{ent}} + u_t^{\text{time}}\big)\,, \tag{1}$$

where W_r, W_a are learnable projections and ϕ is a nonlinear transformation. The resulting lattice $G = \{g_{i,t}\}$ represents each entity trajectory as a row, with valid contiguous sequences $G_{i,t_s:t_e} = \{g_{i,t}\}_{t=t_s}^{t_e}$ defining entity-span hypotheses $h = (i, t_s, t_e)$.

3.3 Stage-Aware Context Matching

The entity-time lattice induces competing entity-span hypotheses that may contain locally relevant evidence but differ in whether the selected entity and temporal extent are consistent with the queried event. Stage-aware context matching evaluates each candidate span against the pre-event, event, and post-event structure along its trajectory, together with the entity's overall relevance to the query.

3.3.1 Entity Context Anchoring

Stage-aware assessment first requires associating the temporal stage structure with the entity referred to by the query. As illustrated in Fig. 2(b), query tokens and valid entity-time cells are projected into a shared semantic space and matched pairwise using cosine similarity, yielding $S_i \in \mathbb{R}^{L_q \times |\mathcal{V}_i|}$. Here, L_q is the number of query tokens and $\mathcal{V}_i$ denotes the valid temporal positions of entity i. Since complete trajectories contain segments unrelated to the queried event, alignment over S_i is formulated as entropy-regularized partial transport:

$$P_i^* = \underset{P \in \Pi_\rho(\mu,\nu_i)}{\arg\max}\ [\langle P, S_i \rangle + \varepsilon \mathcal{H}(P)]\,, \tag{2}$$

where μ and ν_i denote the mass distributions over query tokens and valid temporal positions, respectively, Π_ρ is the partial transport feasible set with transported mass ratio ρ, and $\mathcal{H}(P)$ is the entropy regularizer. The entity relevance score $S_{\text{entity}}(i)$ is the weighted average of similarities in S_i, using the transport weights P_i^* and normalizing by their total mass. This score anchors temporal stage assessment to the query-relevant entity.

3.3.2 Temporal Stage Alignment

Distinguishing candidate spans along the same entity trajectory requires identifying the relative stage of each temporal position. We adopt three stages, PRE, EVENT, and POST, corresponding to before, during, and after the event.

To capture local temporal dynamics, we concatenate entity-time features within a neighborhood of radius r and project them into a shared space. Each stage representation combines a learnable stage embedding with the query anchor. Scaled dot-product matching between the projected local features and stage representations, followed by softmax over the three stages, yields stage probabilities $p_{i,t}^c$.

For each hypothesis $h = (i, t_s, t_e)$, the temporal alignment score $S_{\text{align}}(h)$ sums the mean PRE, EVENT, and POST probabilities before, within, and after the candidate interval, respectively. We combine temporal alignment with entity relevance to obtain the stage-aware context matching score:

$$S_{\text{stage}}(h) = \lambda_{\text{entity}} S_{\text{entity}}(i) + \lambda_{\text{align}} S_{\text{align}}(h), \tag{3}$$

where the two weights balance entity relevance and temporal stage alignment.

3.4 State-Aware Process Matching

Stage-aware context matching captures the relevance of the candidate entity and its temporal extent within the surrounding stage structure, while leaving the event's internal organization unresolved. We decompose each candidate interval into a sequence of latent states to characterize changes in entity actions and interactions throughout the event.

3.4.1 Latent State Response

A target event usually consists of multiple internal stages, reflected in changes in entity actions and interactions with the environment or other entities. We introduce M latent states to represent these internal variations and compute the response of each temporal position to each state, capturing its role within the event progression. Specifically, the learnable embedding u_z^{state} of state z is combined with the anchor representation h_a to obtain a query-conditioned state representation $q_z = W_q[h_a; u_z^{\text{state}}]$, where $z = 1, \ldots, M$. For each position $t \in [t_s, t_e]$ of hypothesis $h = (i, t_s, t_e)$, its compatibility with latent state z is computed as:

$$L_{\text{state}}(i, t, z) = \frac{(W_{\text{state}} g_{i,t})^\top q_z}{\sqrt{d}}, \tag{4}$$

where W_q and W_{state} are learnable projections and d is the matching dimension. Each position thus obtains M query-conditioned state responses, indicating which latent states are supported by its visual content and providing local scores for subsequent transition modeling.

3.4.2 Conditional State Transition

State responses capture only the local compatibility between temporal positions and latent states, and are insufficient to describe the continuous evolution of internal event states. We further model transitions between adjacent latent states, conditioned on local visual changes and query semantics. As shown in Fig. 2(c), we concatenate adjacent entity-time features $g_{i,t}$ and $g_{i,t+1}$, their difference $g_{i,t+1} - g_{i,t}$, and the query anchor h_a. Projection by W_d yields the transition representation $d_{i,t}$, which scores advancement from state z to $z + 1$:

$$L_{\text{trans}}(i, t, z) = \frac{\left(W_{\text{trans}}[u_z^{\text{state}}; u_{z+1}^{\text{state}}]\right)^\top d_{i,t}}{\sqrt{d}}, \qquad z = 1, \ldots, M-1, \tag{5}$$

where W_d and W_{trans} are learnable projection matrices. $L_{\text{trans}}(i, t, z)$ measures how strongly the query-conditioned visual change between adjacent positions supports the transition from state z to $z + 1$. These transition scores complement state responses for evaluating candidate latent paths.

3.4.3 HIERARCHICAL PATH MARGINALIZATION

The state and transition scores define a latent event map with M states at each temporal position. Each admissible path $\pi = (z_{t_s}, \ldots, z_{t_e}) \in \Pi(h)$, as exemplified in Fig. 2(c), selects a state $z_t \in \{1, \ldots, M\}$ at each position. Consecutive states satisfy $z_{t+1} \in \{z_t, z_t + 1\}$, preserving ordered progression through state persistence or forward advancement. The path score sums the selected state responses and transition scores at actual state advances:

$$S_{\text{path}}(h, \pi) = \sum_{t=t_s}^{t_e} L_{\text{state}}(i, t, z_t) + \sum_{t=t_s}^{t_e-1} \mathbb{I}[z_{t+1} = z_t + 1] L_{\text{trans}}(i, t, z_t). \tag{6}$$

Direct marginalization over all paths favors transition counts with more admissible paths. We group paths with exactly r forward transitions into $\Pi_r(h)$ and apply log-mean-exp first within each group, then across the valid transition counts $\mathcal{R}(h)$:

$$S_r(h) = \log \frac{\sum_{\pi \in \Pi_r(h)} e^{S_{\text{path}}(h,\pi)}}{|\Pi_r(h)|}, \qquad S_{\text{state}}(h) = \log \frac{\sum_{r \in \mathcal{R}(h)} e^{S_r(h)}}{|\mathcal{R}(h)|}. \tag{7}$$

The first level aggregates paths with different transition locations and state durations under the same transition count, normalizing for the number of admissible paths. The second level combines groups with different transition counts. The resulting $S_{\text{state}}(h)$ captures alternative internal event progressions within the candidate interval.

3.5 UNIFIED INFERENCE

We combine stage-aware context matching and state-aware process matching:

$$S(h) = S_{\text{stage}}(h) + \lambda_{\text{state}} S_{\text{state}}(h), \tag{8}$$

where λ_{state} weights internal event progression. We select the highest-scoring entity-span from the entity-time lattice, jointly determining the target entity tube and its temporal interval.

We normalize $S(h)$ over entities and temporal intervals, respectively, to obtain conditional distributions for supervision. We combine these losses with expected vIoU regularization:

$$L_{\text{loc}} = L_{\text{entity}} + L_{\text{time}} - \eta R_{\text{vIoU}}, \tag{9}$$

where L_{entity} supervises entity selection and L_{time} supervises entity-conditioned temporal localization. R_{vIoU} denotes the expected candidate vIoU, and η controls its contribution. This objective jointly trains stage-aware context matching and state-aware process matching through their combined scores. Detailed loss formulations are provided in Appendix A.2.2.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

Benchmarks and Metrics. Our evaluation covers spatio-temporal video grounding and transfer to related video grounding tasks, including spatial grounding, temporal grounding, and referring video object segmentation. The main results on spatio-temporal video grounding are reported on HCSTVG-v1/v2 (Tang et al., 2021). For transfer evaluation, ST-Align (Li et al., 2025) assesses spatial and spatio-temporal localization of humans and objects, MeViS (Ding et al., 2023) evaluates referring video object segmentation, and Charades-STA (Gao et al., 2017) and TVGBench (Wang et al., 2026b) assess temporal grounding. We report m_tIoU, m_vIoU, and vIoU@$\{0.3, 0.5\}$ for spatio-temporal grounding, tIoU@$\{0.3, 0.5\}$ for temporal grounding, and $\mathcal{J}$, $\mathcal{F}$, $\mathcal{J}\&\mathcal{F}$ for MeViS.

Implementation Details. We implement PAEM on four vision-language backbones: InternVL-3.5-8B (Wang et al., 2025), VideoLLaMA3-7B (Zhang et al., 2025), Qwen2.5-VL-7B (Bai et al., 2025b), and Qwen3-VL-8B (Bai et al., 2025a). We freeze the original backbone parameters, including the visual encoder, and jointly optimize all PAEM module parameters and LoRA parameters. The baseline uses the same LoRA configuration without PAEM and is trained solely with the language modeling loss. Videos are sampled at 2 FPS and processed in 24-frame training windows

Table 1: Performance comparison on HCSTVG-v1, HCSTVG-v2, and ST-Align (%). † denotes backbones fine-tuned using only the language modeling (LM) loss, without PAEM.

Models	HCSTVG-v1		HCSTVG-v2		ST-Align		
	STVG		STVG		STVG		SVG
	m_tIoU	m_vIoU	m_tIoU	m_vIoU	m_tIoU	m_vIoU	m_vIoU
TubeDETR (Yang et al., 2022)	–	32.4	53.9	36.4	–	–	–
STVGFormer (Lin et al., 2023)	–	36.9	58.1	38.7	–	–	–
CG-STVG (Gu et al., 2024)	52.8	38.4	60.0	39.5	–	–	–
TA-STVG (Gu et al., 2025a)	53.0	39.1	60.4	40.2	–	–	–
GroundingGPT-7B (Li et al., 2024)	22.2	16.7	19.6	14.7	12.2	9.2	17.9
VTimeLLM-7B (Huang et al., 2024)	25.8	18.4	28.6	20.1	15.5	10.8	22.7
Grounded-VideoLLM-7B (Wang et al., 2024a)	31.6	22.7	34.8	24.3	33.0	18.6	27.4
LLaVA-ST-7B (Li et al., 2025)	42.1	28.9	45.7	30.6	43.8	22.8	32.5
SpaceVLLM-7B (Wang et al., 2026a)	56.9	39.3	58.0	34.0	41.7	21.9	31.6
STVG-R1 (Zhang et al., 2026)	56.9	39.1	61.3	40.8	45.1	23.4	48.6
InternVL-3.5-8B†	34.4	26.8	48.9	29.0	32.7	19.4	29.9
+*PAEM*	48.2 ↑+13.8	31.5 ↑+4.7	54.2 ↑+5.3	35.6 ↑+6.6	36.1 ↑+3.4	22.7 ↑+3.3	34.7 ↑+4.8
VideoLLaMA3-7B†	51.2	30.7	55.5	34.7	31.9	21.1	39.4
+*PAEM*	55.6 ↑+4.4	37.1 ↑+6.4	59.1 ↑+3.6	38.3 ↑+3.6	33.4 ↑+1.5	22.9 ↑+1.8	42.6 ↑+3.2
Qwen2.5-VL-7B†	52.4	33.6	57.9	34.5	36.3	20.1	42.2
+*PAEM*	58.7 ↑+6.3	41.6 ↑+8.0	64.0 ↑+6.1	42.0 ↑+7.5	45.7 ↑+9.4	**25.1** ↑+5.0	**49.9** ↑+7.7
Qwen3-VL-8B†	53.2	33.3	57.3	34.9	39.9	18.4	41.4
+*PAEM*	**61.4** ↑+8.2	**43.2** ↑+9.9	**66.2** ↑+8.9	**42.9** ↑+8.0	**47.2** ↑+7.3	24.9 ↑+6.5	48.7 ↑+7.3

with a stride of 12 frames. We merge and deduplicate the HCSTVG-v1/v2 training sets, excluding overlaps with validation or test data. VidSTG (Zhang et al., 2020) is additionally included to cover non-human targets and interactions. Entity tubes are precomputed using Grounding DINO (Liu et al., 2024a) and SAM2.1-Small (Ravi et al., 2025). Further details are provided in the appendix.

4.2 Results on Spatio-Temporal Video Grounding

Table 1 compares PAEM with existing methods on HCSTVG-v1/v2 and evaluates its zero-shot transfer to ST-Align across four vision-language backbones. On HCSTVG-v1/v2, PAEM with Qwen3-VL-8B achieves 43.2 and 42.9 m_vIoU, exceeding the strongest competing results in the table by 3.9% ↑ over SpaceVLLM on v1 and 2.1% ↑ over STVG-R1 on v2. To distinguish PAEM's gains from the capabilities of the underlying backbones, we also compare it with baselines that use the same LoRA configuration without PAEM modules, marked by † in the table. PAEM improves both m_tIoU and m_vIoU across all four backbones, with m_vIoU gains of 4.7%-9.9% on HCSTVG-v1 and 3.6%-8.0% on HCSTVG-v2. These consistent improvements indicate that progression-aware event matching benefits backbones with different initial grounding capabilities.

Without target-dataset fine-tuning, PAEM also improves both spatio-temporal and spatial grounding on ST-Align across all four backbones. With Qwen2.5-VL-7B, PAEM achieves 25.1 and 49.9 m_vIoU for STVG and SVG, respectively, surpassing STVG-R1, the strongest competing method in the table for both tasks, by 1.7% ↑ and 1.3% ↑. These results support the transferability of PAEM to both joint spatio-temporal localization and spatial grounding.

Table 2: Performance comparison on MeViS (%). The results of TrackGPT are reported by VISA. †: fine-tuned with LM loss only, without PAEM.

Models	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$
URVOS (Seo et al., 2020)	25.7	29.9	27.8
MTTR (Botach et al., 2022)	28.8	31.2	30.0
ReferFormer (Wu et al., 2022)	29.8	32.2	31.0
LMPM (Ding et al., 2023)	34.2	40.2	37.2
LISA (Lai et al., 2024)	35.1	39.4	37.2
TrackGPT (Zhu et al., 2023)	37.6	42.6	40.1
VISA (Yan et al., 2024)	40.7	46.3	43.5
VideoGLaMM (Munasinghe et al., 2025)	42.1	48.2	45.2
STVG-R1 (Zhang et al., 2026)	44.7	50.0	47.3
InternVL-3.5-8B†	31.3	36.4	33.9
+*PAEM*	36.2 ↑+4.9	41.2 ↑+4.8	38.7 ↑+4.8
VideoLLaMA3-7B†	27.9	34.2	31.1
+*PAEM*	35.6 ↑+7.7	42.8 ↑+8.6	39.2 ↑+8.1
Qwen2.5-VL-7B†	38.4	43.7	41.1
+*PAEM*	**46.2** ↑+7.8	**50.5** ↑+6.8	**48.4** ↑+7.3
Qwen3-VL-8B†	36.4	44.2	40.3
+*PAEM*	45.7 ↑+9.3	49.8 ↑+5.6	47.8 ↑+7.5

4.3 Cross-Task Generalization

To further assess transferability, we extend evaluation to referring video object segmentation and temporal grounding on MeViS, Charades-STA, and TVGBench without target-dataset fine-tuning. MeViS extends localization from bounding boxes to masks across frames. As shown in Table 2, PAEM consistently improves

segmentation across all four backbones, with $\mathcal{J}\&\mathcal{F}$ gains of 4.8–8.2% ↑. These results suggest that event structure provides useful discriminative information for pixel-level localization, extending beyond tube prediction in the source task. With Qwen2.5-VL-7B, PAEM achieves 48.4 $\mathcal{J}\&\mathcal{F}$ and surpasses STVG-R1 across all three metrics.

The temporal grounding results in Table 3 show similarly consistent gains, with tIoU@0.5 improving by 4.6–6.1% ↑ on Charades-STA and 2.8–6.0% ↑ on TVGBench. With Qwen3-VL-8B, PAEM achieves the best results among the compared methods on both TVGBench metrics and tIoU@0.3 on Charades-STA, while remaining competitive at the stricter tIoU@0.5 threshold on Charades-STA. Together, the spatial, segmentation, and temporal results support the transferability of progression-aware event matching to both target identification and event interval prediction.

Table 3: Performance comparison on Charades-STA and TVGBench (%) under the zero-shot setting. † denotes backbones fine-tuned with LM loss only, without PAEM.

Models	Charades-STA		TVGBench	
	tIoU@0.3	tIoU@0.5	tIoU@0.3	tIoU@0.5
TimeSuite (Zeng et al., 2025)	69.9	48.7	31.1	18.0
TRACE (Guo et al., 2024)	–	40.3	37.0	25.5
LLaVA-ST (Li et al., 2025)	63.1	44.8	–	–
STVG-R1 (Zhang et al., 2026)	<u>73.2</u>	**52.5**	<u>42.5</u>	<u>27.4</u>
InternVL-3.5-8B†	52.4	31.8	29.6	15.7
+PAEM	58.1 ↑+5.7	36.4 ↑+4.6	32.8 ↑+3.2	18.5 ↑+2.8
VideoLLaMA3-7B†	63.7	40.9	36.1	21.2
+PAEM	66.9 ↑+3.2	46.2 ↑+5.3	41.3 ↑+5.2	27.2 ↑+6.0
Qwen2.5-VL-7B†	63.2	43.7	35.8	21.6
+PAEM	71.5 ↑+8.3	49.1 ↑+5.4	40.2 ↑+4.4	25.1 ↑+3.5
Qwen3-VL-8B†	64.7	45.9	39.2	24.4
+PAEM	**73.4** ↑+8.7	<u>52.0</u> ↑+6.1	**43.8** ↑+4.6	**29.0** ↑+4.6

Table 4: Ablation results on HCSTVG-v1/v2 with Qwen2.5-VL-7B as the backbone.

Configuration	HCSTVG-v1				HCSTVG-v2			
	m_tIoU	m_vIoU	vIoU@0.3	vIoU@0.5	m_tIoU	m_vIoU	vIoU@0.3	vIoU@0.5
(a) **Full PAEM**	58.7	41.6	70.0	41.2	64.0	42.0	71.7	41.0
(b) w/o Entity-Time Lattice	54.9 ↓3.8	37.8 ↓3.8	63.7 ↓6.3	35.9 ↓5.3	60.7 ↓3.3	38.6 ↓3.4	65.8 ↓5.9	36.2 ↓4.8
(c) **w/o Stage-Aware Context Matching**	56.5 ↓2.2	36.9 ↓4.7	62.8 ↓7.2	34.4 ↓6.8	61.2 ↓2.8	37.4 ↓4.6	63.9 ↓7.8	35.1 ↓5.9
(d) w/o Entity Context Anchoring	58.4 ↓0.3	38.1 ↓3.5	64.9 ↓5.1	36.7 ↓4.5	64.1 ↑0.1	38.7 ↓3.3	66.2 ↓5.5	37.0 ↓4.0
(e) w/o Temporal Stage Alignment	56.9 ↓1.8	40.0 ↓1.6	67.4 ↓2.6	39.1 ↓2.1	61.8 ↓2.2	40.9 ↓1.1	69.5 ↓2.2	39.4 ↓1.6
(f) **w/o State-Aware Process Matching**	52.6 ↓6.1	38.8 ↓2.8	65.6 ↓4.4	37.5 ↓3.7	57.2 ↓6.8	38.7 ↓3.3	66.8 ↓4.9	36.9 ↓4.1
(g) w/o Latent State Response	55.8 ↓2.9	39.5 ↓2.1	66.9 ↓3.1	38.2 ↓3.0	60.6 ↓3.4	40.5 ↓1.5	69.1 ↓2.6	38.4 ↓2.6
(h) w/o Conditional State Transition	54.4 ↓4.3	40.2 ↓1.4	68.1 ↓1.9	39.0 ↓2.2	59.1 ↓4.9	40.2 ↓1.8	68.7 ↓3.0	38.6 ↓2.4
(i) w/o Hierarchical Path Marginalization	57.3 ↓1.4	41.0 ↓0.6	68.5 ↓1.5	40.5 ↓0.7	62.1 ↓1.9	41.5 ↓0.5	70.6 ↓1.1	40.2 ↓0.8

4.4 Ablation Analysis

Tab. 4 evaluates PAEM's components with Qwen2.5-VL-7B under identical training settings. Each variant independently modifies the full model (a). Removing the Entity-Time Lattice (b) reduces m_vIoU by 3.8% ↓ and 3.4% ↓ on HCSTVG-v1/v2, highlighting its foundational role in organizing entity-specific temporal information for structured matching. On this representation, Stage-Aware Contextual Matching provides key constraints on target identity and event extent, with its removal (c) lowering m_vIoU by 4.7% ↓ and 4.6% ↓. Within this branch, removing Entity Context Anchoring (d) mainly reduces m_vIoU, while removing Temporal Stage Alignment (e) lowers m_tIoU, revealing their complementary roles in target discrimination and event extent estimation. Beyond these constraints, State-Aware Progression Matching contributes more strongly to temporal localization, with its removal (f) lowering m_tIoU by 6.1% ↓ and 6.8% ↓. Removing Latent State Response (g) degrades both metrics, while removing Conditional State Transition (h) causes larger temporal drops, highlighting the importance of state compatibility and advancement for internal event progression. Finally, performance drops without Hierarchical Path Marginalization (i) support the value of aggregating alternative latent paths for event matching.

4.5 Discussion on Matching Behavior

To understand how PAEM's matching decisions contribute to localization gains, we analyze PAEM's entity selection and temporal localization, score complementarity, and relate changes in entity selection to final spatio-temporal gains. As shown in Fig. 3(a), PAEM achieves higher entity selection accuracy across candidate counts, demonstrating stronger target discrimination. This advantage extends to temporal localization in (b). Although boundary errors increase for both models at higher event coverage ratios, PAEM maintains lower errors, indicating more accurate event extent estima-

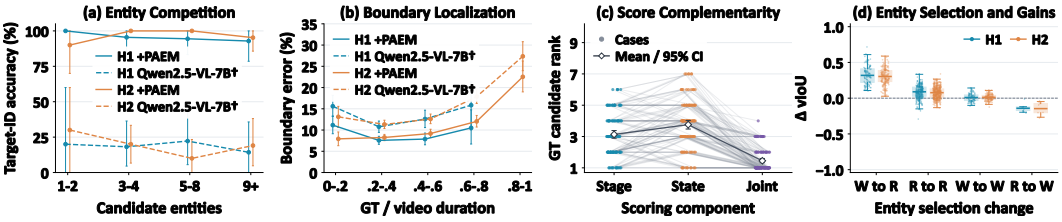


Figure 3: Analysis of matching behavior. H1/H2 denote HCSTVG-v1/v2, respectively.

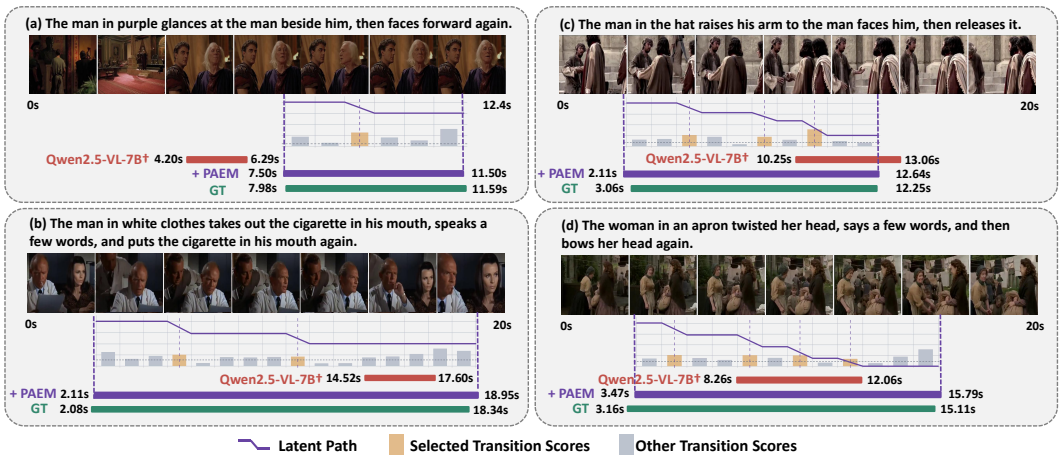


Figure 4: Temporal localization comparison of Qwen2.5-VL-7B† with and without PAEM.

tion. In (c), joint scoring improves the average GT candidate rank, highlighting the complementarity of stage context and internal state progression in reducing candidate ambiguity. Finally, (d) shows the largest vIoU gains when entity selection is corrected, with positive gains remaining when both models select the correct target. These results indicate that PAEM benefits both target discrimination and event extent estimation, supporting more accurate spatio-temporal localization.

4.6 Case Study

To illustrate the relationship between internal state progression and temporal localization, Fig. 4 presents PAEM's temporal predictions and the corresponding highest-scoring latent paths. The number of visited states is automatically determined by the model based on candidate content. We select four cases whose latent paths contain one to four state transitions, respectively, for visualization in (a)–(d). In (a), the baseline predicts an interval before the man glances at the person beside him and faces forward again. In (b)–(d), its predictions miss the beginning or ending of the described action sequences. By jointly assessing latent state responses and conditional transition scores, PAEM identifies the queried event in (a) and extends temporal coverage to include "takes out the cigarette" in (b) and "raises his arm" in (c), which the baseline misses. In (d), PAEM covers both "twisted her head" and "bows her head again", recovering the beginning and ending of the queried sequence beyond the baseline's middle interval. These results illustrate how PAEM captures event structure to improve the completeness and accuracy of temporal localization.

5 Conclusion

We proposed Progression-Aware Event Matching (PAEM), which evaluates both semantic relevance and event structure for spatio-temporal video grounding. By coupling stage relations along entity trajectories with latent state progression within candidate spans, PAEM distinguishes complete events from candidates that share relevant content but differ in temporal coverage and internal progression. Improvements across four vision-language backbones and transfer to related grounding tasks support event progression as an effective basis for target identification and complete event localization.

AI USE STATEMENT

We used generative AI tools for Chinese–English translation and feedback on research methodology and experimental analysis, including the description of PAEM's matching formulation and the interpretation of ablation results and grounding cases. Additional uses included drafting and editing manuscript text, organizing the paper and appendix, searching for relevant literature, and assisting with figure layouts and captions. The authors checked AI-assisted text and visualizations against the actual implementation, recorded predictions, experimental results, and cited sources, and take full responsibility for the final manuscript. Our implementation is available for anonymous review at `https://anonymous.4open.science/r/paem-4B85/`.

REPRODUCIBILITY STATEMENT

We document PAEM in detail to support reproducibility, covering both its formulation and the implementation choices needed to reproduce the experiments. Sec. 3.2 and App. A.2 specify the matching procedure and training objectives, down to the normalization over valid candidates and the construction of supervision targets. Sec. 4.1 and App. A.1 provide the experimental settings, including the exact sample counts after overlap removal, the segmentation checkpoint and configuration, ground-truth association and proposal reuse, and training hyperparameters such as the random seed. We also explain how predictions are converted for each transfer task, making the evaluation setup explicit.

REFERENCES

Ghazi Shazan Ahmad, Ahmed Heakl, Hanan Gani, Abdelrahman Shaker, Zhiqiang Shen, Fahad Shahbaz Khan, and Salman Khan. Videomolmo: Spatio-temporal grounding meets pointing. *arXiv preprint arXiv:2506.05336*, 2025.

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*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. corr abs/2502.13923 (2025). *arXiv preprint arXiv:2502.13923*, 2025b.

Adam Botach, Evgenii Zheltonozhskii, and Chaim Baskin. End-to-end referring video object segmentation with multimodal transformers. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4975–4985. IEEE, 2022.

Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, and Chen Change Loy. Mevis: A large-scale benchmark for video segmentation with motion expressions. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2694–2703. IEEE, 2023.

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.

Xin Gu, Heng Fan, Yan Huang, Tiejian Luo, and Libo Zhang. Context-guided spatio-temporal video grounding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18330–18339. IEEE, 2024.

Xin Gu, Yaojie Shen, Chenxi Luo, Tiejian Luo, Yan Huang, Yuewei Lin, Heng Fan, and Libo Zhang. Knowing your target: Target-aware transformer makes better spatio-temporal video grounding. In *International Conference on Learning Representations*, volume 2025, pp. 62122–62142, 2025a.

Xin Gu, Haoji Zhang, Qihang Fan, Jingxuan Niu, Zhipeng Zhang, Libo Zhang, Guang Chen, Fan Chen, Longyin Wen, and Sijie Zhu. Thinking with bounding boxes: Enhancing spatio-temporal video grounding via reinforcement fine-tuning. *arXiv preprint arXiv:2511.21375*, 2025b.

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

Jiantang Huang. Two-pass zero-shot temporal-spatial grounding of rare traffic events in surveillance video. *arXiv preprint arXiv:2605.01512*, 2026.

Md Mohaiminul Islam, Ngan Ho, Xitong Yang, Tushar Nagarajan, Lorenzo Torresani, and Gedas Bertasius. Video recap: Recursive captioning of hour-long videos. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18198–18208. IEEE, 2024.

Jinhyun Jang, Jungin Park, Jin Kim, Hyeongjun Kwon, and Kwanghoon Sohn. Knowing where to focus: Event-aware transformer for video grounding. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13800–13810. IEEE, 2023.

Yang Jin, Zehuan Yuan, Yadong Mu, et al. Embracing consistency: A one-stage approach for spatio-temporal video grounding. *Advances in Neural Information Processing Systems*, 35:29192–29204, 2022.

Zeeshan Khan, CV Jawahar, and Makarand Tapaswi. Grounded video situation recognition. *Advances in Neural Information Processing Systems*, 35:8199–8210, 2022.

Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9579–9589. IEEE, 2024.

Hongyu Li, Jinyu Chen, Ziyu Wei, Shaofei Huang, Tianrui Hui, Jialin Gao, Xiaoming Wei, and Si Liu. Llava-st: A multimodal large language model for fine-grained spatial-temporal understanding. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8592–8603. IEEE, 2025.

Zhaowei Li, Qi Xu, Dong Zhang, Hang Song, Yiqing Cai, Qi Qi, Ran Zhou, Junting Pan, Zefeng Li, Vu Tu, et al. Groundinggpt: Language enhanced multi-modal grounding model. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6657–6678, 2024.

Yaoyuan Liang, Xiao Liang, Yansong Tang, Zhao Yang, Ziran Li, Jingang Wang, Wenbo Ding, and Shao-Lun Huang. Costa: End-to-end comprehensive space-time entanglement for spatio-temporal video grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 3324–3332, 2024.

Zihang Lin, Chaolei Tan, Jian-Fang Hu, Zhi Jin, Tiancai Ye, and Wei-Shi Zheng. Collaborative static and dynamic vision-language streams for spatio-temporal video grounding. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23100–23109. IEEE, 2023.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024a.

Ye Liu, Zongyang Ma, Zhongang Qi, Yang Wu, Ying Shan, and Chang W Chen. Et bench: Towards open-ended event-level video-language understanding. *Advances in Neural Information Processing Systems*, 37:32076–32110, 2024b.

Jiahao Meng, Xiangtai Li, Haochen Wang, Yue Tan, Tao Zhang, Lingdong Kong, Yunhai Tong, Anran Wang, Zhiyang Teng, Yujing Wang, et al. Open-o3-video: Grounded video reasoning with explicit spatio-temporal evidence. *arXiv preprint arXiv:2510.20579*, 2025.

Shehan Munasinghe, Hanan Gani, Wenqi Zhu, Jiale Cao, Eric Xing, Fahad Shahbaz Khan, and Salman Khan. Videoglamm: A large multimodal model for pixel-level visual grounding in videos. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19036–19046. IEEE, 2025.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations*, volume 2025, pp. 28085–28128, 2025.

Arka Sadhu, Tanmay Gupta, Mark Yatskar, Ram Nevatia, and Aniruddha Kembhavi. Visual semantic role labeling for video understanding. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5585–5596. IEEE, 2021.

Seonguk Seo, Joon-Young Lee, and Bohyung Han. Urvos: Unified referring video object segmentation network with a large-scale benchmark. In *European conference on computer vision*, pp. 208–223. Springer, 2020.

Rui Su, Qian Yu, and Dong Xu. Stvgbert: A visual-linguistic transformer based framework for spatio-temporal video grounding. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1513–1522. IEEE, 2021.

Zongheng Tang, Yue Liao, Si Liu, Guanbin Li, Xiaojie Jin, Hongxu Jiang, Qian Yu, and Dong Xu. Human-centric spatio-temporal video grounding with visual transformers. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(12):8238–8249, 2021.

Haibo Wang, Zhiyang Xu, Yu Cheng, Shizhe Diao, Yufan Zhou, Yixin Cao, Qifan Wang, Weifeng Ge, and Lifu Huang. Grounded-videollm: Sharpening fine-grained temporal grounding in video large language models. *arXiv preprint arXiv:2410.03290*, 2024a.

Jiankang Wang, Zhihan Zhang, Zhihang Liu, Yang Li, Jiannan Ge, Hongtao Xie, and Yongdong Zhang. Spacevllm: Endowing multimodal large language model with spatio-temporal video grounding capability. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 9912–9920, 2026a.

Teng Wang, Ruimao Zhang, Zhichao Lu, Feng Zheng, Ran Cheng, and Ping Luo. End-to-end dense video captioning with parallel decoding. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6827–6837. IEEE, 2021.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Ye Wang, Ziheng Wang, Boshen Xu, Yang Du, Kejun Lin, Zihan Xiao, Zihao Yue, Jianzhong Ju, Liang Zhang, Dingyi Yang, et al. Time-r1: Post-training large vision language model for temporal video grounding. *Advances in Neural Information Processing Systems*, 38:83330–83364, 2026b.

Yuxuan Wang, Yueqian Wang, Pengfei Wu, Jianxin Liang, Dongyan Zhao, and Zilong Zheng. Lstp: Language-guided spatial-temporal prompt learning for long-form video-text understanding. *arXiv preprint arXiv:2402.16050*, 2, 2024b.

Jiannan Wu, Yi Jiang, Peize Sun, Zehuan Yuan, and Ping Luo. Language as queries for referring video object segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4964–4974. IEEE, 2022.

Qi'ao Xu, Tianwen Qian, Yuqian Fu, Kailing Li, Yang Jiao, Jiacheng Zhang, Xiaoling Wang, and Liang He. Tog-bench: Task-oriented spatio-temporal grounding in egocentric videos. *arXiv preprint arXiv:2512.03666*, 2025.

Cilin Yan, Haochen Wang, Shilin Yan, Xiaolong Jiang, Yao Hu, Guoliang Kang, Weidi Xie, and Efstratios Gavves. Visa: Reasoning video object segmentation via large language models. In *European Conference on Computer Vision*, pp. 98–115. Springer, 2024.

Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Tubedetr: Spatio-temporal video grounding with transformers. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16421–16432. IEEE, 2022.

Antoine Yang, Arsha Nagrani, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vidchapters-7m: Video chapters at scale. *Advances in Neural Information Processing Systems*, 36:49428–49444, 2023a.

Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10714–10726. IEEE, 2023b.

Zaiquan Yang, Yuhao Liu, Gerhard Hancke, and Rynson Lau. Unleashing the potential of multimodal llms for zero-shot spatio-temporal video grounding. *Advances in Neural Information Processing Systems*, 38:22636–22670, 2026.

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.

Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. Videollama 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025.

Xiaowen Zhang, Zhi Gao, Licheng Jiao, Lingling Li, and Qing Li. Stvg-r1: Incentivizing instance-level reasoning and grounding in videos via reinforcement learning. *arXiv preprint arXiv:2602.11730*, 2026.

Zhu Zhang, Zhou Zhao, Yang Zhao, Qi Wang, Huasheng Liu, and Lianli Gao. Where does it exist: Spatio-temporal video grounding for multi-form sentences. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10665–10674. IEEE, 2020.

Xingyi Zhou, Anurag Arnab, Shyamal Buch, Shen Yan, Austin Myers, Xuehan Xiong, Arsha Nagrani, and Cordelia Schmid. Streaming dense video captioning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18243–18252. IEEE, 2024.

Jiawen Zhu, Zhi-Qi Cheng, Jun-Yan He, Chenyang Li, Bin Luo, Huchuan Lu, Yifeng Geng, and Xuansong Xie. Tracking with human-intent reasoning. *arXiv preprint arXiv:2312.17448*, 2023.

A Appendix

A.1 Datasets and Implementation Details

A.1.1 Training Data Preparation

HCSTVG overlap removal. To avoid cross-dataset leakage when jointly training on HCSTVG-v1 and HCSTVG-v2, we remove training samples that overlap with either evaluation split. Specifically, we exclude HCSTVG-v1 training samples overlapping with the HCSTVG-v2 validation set, and HCSTVG-v2 training samples overlapping with the HCSTVG-v1 test set. This leaves 2,748 and 9,047 training samples, respectively, yielding a merged and deduplicated set of 11,795 samples.

VidSTG sampling. We additionally incorporate VidSTG to broaden the coverage of non-human entities and entity interactions. VidSTG includes both declarative captions and interrogative questions. To balance these query types while maintaining comparable contributions from VidSTG and HCSTVG, we sample 5,898 training instances of each type. The resulting 11,796 VidSTG instances approximately match the size of the cleaned HCSTVG set. We combine the two sources and keep the resulting training set fixed across all backbones and their corresponding baselines.

A.1.2 Entity Tube Extraction

Detection and temporal propagation. We construct entity tubes offline using Grounding DINO and SAM2.1-Small. Grounding DINO detects candidate bounding boxes on anchor frames, followed by non-maximum suppression to remove redundant detections. The retained boxes serve as spatial prompts for SAM2.1-Small, using the sam2.1_hiera_small.pt checkpoint and the sam2.1_hiera_s.yaml configuration. The predicted instance masks are propagated across video frames while maintaining their instance identifiers, forming candidate entity trajectories. These trajectories provide spatial candidates; the queried event interval is determined separately by the grounding model.

Tube representation and temporal alignment. Each tube is represented by an instance identifier and its frame-indexed spatial regions. For box-based grounding, each valid instance mask is converted into an axis-aligned bounding box. We preserve the correspondence between proposal frame indices and the source-video timeline to align the trajectories with the sampled visual features. Proposal frame indices are therefore distinguished from sampled-token positions and aligned through the source-frame mapping. The aligned boxes define the spatial support of entity-time lattice cells, while the instance identifiers preserve entity correspondence across time.

Training association and proposal reuse. For entity-selection supervision, we associate each annotated target with the retained proposal having the highest spatio-temporal overlap with its ground-truth tube. The matched proposal provides the target entity identifier, while the annotated event interval provides temporal supervision. This association assigns training labels to existing proposals without replacing their predicted trajectories with ground-truth regions. We reuse the precomputed proposals and their identifier mappings across all backbones and their corresponding baselines, ensuring a consistent spatial candidate set for comparison.

A.1.3 Training Configuration

We freeze the pretrained backbone parameters and optimize the LoRA adapters together with the PAEM modules. For each backbone, its corresponding baseline uses the same training data and LoRA configuration but is optimized solely with the language modeling loss. Table 5 summarizes the training hyperparameters.

A.1.4 Cross-Task Adaptation

Spatial and spatio-temporal grounding. For spatio-temporal grounding on HCSTVG and ST-Align, we return the selected entity's bounding-box trajectory within the predicted event interval. For spatial grounding on ST-Align, we retain the entity prediction and evaluate its bounding boxes at the frames specified by the evaluation protocol. The spatial output is obtained directly from the selected proposal rather than from an additional box-regression head.

Table 5: Training configuration.

Hyperparameter	Value
Video sampling rate	2 FPS
Training window length	24 sampled frames
Window stride	12 sampled frames
Maximum cached visual tokens per window	3,072
LoRA rank	128
LoRA scaling factor α	256
LoRA dropout	0.05
PAEM hidden dimension	1,024
PAEM dropout	0.1
LoRA learning rate	1×10^{-5}
PAEM learning rate	1×10^{-5}
Weight decay	0.01
Per-device batch size	1
Effective batch size	4
Learning rate schedule	Linear decay
Warmup steps	0
Training precision	BF16
Gradient checkpointing	Enabled
Random seed	42

Temporal grounding. For Charades-STA and TVGBench, entity tubes provide intermediate spatial support for evaluating candidate events, while the final output consists only of the predicted start and end times. Entity identity is treated as an internal inference variable and is not evaluated as a spatial prediction. Placeholder entity identifiers in the data interface are not interpreted as ground-truth entity labels.

Referring video object segmentation. For MeViS, the selected entity is mapped back to its associated SAM2 mask trajectory. This preserves the correspondence between the entity used for matching and the pixel-level prediction. PAEM performs query-conditioned entity selection, while SAM2 provides the instance masks; no additional segmentation decoder is introduced. Under the zero-shot protocol, the source-trained parameters are kept fixed across these target tasks, and adaptation is restricted to input formatting and task-specific output conversion.

A.2 Additional Method Details

A.2.1 Entity-Time Lattice Construction

Spatial and temporal alignment. We associate each visual token with its sampled temporal position and spatial support in the input frame. Entity boxes are mapped into the same coordinate system as the visual-token grid. Pooling is performed separately at each temporal position, preserving the correspondence between an entity's spatial trajectory and its query-conditioned visual features.

Region-weighted pooling. For entity i at position t, let $a_{i,t,j}$ denote the fraction of visual token j's spatial support covered by the entity box $b_{i,t}$. We normalize these overlap weights and compute the pooled feature as

$$w_{i,t,j} = \frac{a_{i,t,j}}{\sum_k a_{i,t,k}}, \qquad r_{i,t} = \sum_j w_{i,t,j} H_v^{(j)}, \tag{10}$$

where both sums are restricted to visual tokens at position t with nonzero total overlap. The weights depend on proposal geometry, while $H_v^{(j)}$ contains query-conditioned visual information. The pooled feature $r_{i,t}$ is then projected and fused with the query anchor, entity embedding, and temporal embedding following the lattice construction in the main text. For batched computation, padding masks exclude padded entities and temporal positions from candidate scoring and normalization.

A.2.2 TRAINING OBJECTIVES

Candidate probabilities. We derive entity and temporal supervision from the joint candidate score $S(h)$ defined in the main text. Let k index a candidate temporal interval and write its score for entity i as $S(i,k)$. Applying softmax over entities at a fixed interval gives $p(i \mid k)$, while applying softmax over intervals for a fixed entity gives $p(k \mid i)$. Both distributions are normalized over valid candidates only.

Temporal supervision. Let $i^\star$ denote the proposal associated with the annotated target, and let u_k be the temporal IoU between candidate interval k and the annotated interval. We construct the temporal targets as

$$q_k = \frac{\exp(u_k/\tau)}{\sum_j \exp(u_j/\tau)}, \qquad y_k = (1-\beta)q_k + \beta b_k, \tag{11}$$

where the sums cover valid intervals of entity $i^\star$. Here, τ controls target softness, and b_k assigns equal probability to the intervals attaining the highest temporal IoU and zero to the others. The coefficient β balances soft overlap-based supervision and emphasis on the best-overlap intervals. Using temporal IoU rather than vIoU separates temporal alignment supervision from spatial proposal quality.

Entity and temporal losses. The two loss terms introduced in the main text are

$$L_{\text{entity}} = -\sum_k q_k \log p(i^\star \mid k), \qquad L_{\text{time}} = -\sum_k y_k \log p(k \mid i^\star). \tag{12}$$

Entity supervision compares entities at a fixed temporal interval and weights each comparison by the interval's temporal relevance. Temporal supervision compares intervals along the matched target trajectory. Both terms optimize the same joint scores, jointly supervising stage-aware context matching and state-aware process matching.

Expected vIoU regularization. We additionally apply softmax to $S(h)$ over all valid entity-span candidates to obtain $p(h)$. The expected candidate vIoU is

$$R_{\text{vIoU}} = \sum_h p(h)\,\text{vIoU}(h, h^\star), \tag{13}$$

where $h^\star$ denotes the ground-truth spatio-temporal target. Including $-\eta R_{\text{vIoU}}$ in the localization loss encourages higher probability for candidates with accurate spatial trajectories and temporal extents. The resulting objective follows the main text:

$$L_{\text{loc}} = L_{\text{entity}} + L_{\text{time}} - \eta R_{\text{vIoU}}. \tag{14}$$

No additional annotations of latent states or transition locations are required.

A.3 ADDITIONAL EXPERIMENTAL RESULTS AND ANALYSIS

Table 6 reports additional vIoU@0.3 and vIoU@0.5 results on HCSTVG-v1/v2. PAEM improves both metrics across all four backbones, indicating that more predictions achieve sufficient joint spatial and temporal overlap with the ground truth. The consistent gains at the stricter 0.5 threshold show that the improvement is not limited to coarse localization, complementing the average IoU results reported in the main text. Unlike mean IoU, these threshold-based results establish that the gains include an increased proportion of successful predictions, rather than only improved overlap within the same success categories.

Table 7 further examines threshold-based performance under zero-shot transfer. For STVG, gains in tIoU@0.5 show that more predicted intervals meet the temporal overlap criterion, while improvements in vIoU@0.5 confirm that more predictions also satisfy the joint localization criterion. In particular, a temporally accurate interval alone does not ensure correct tube localization; improvements in both metrics therefore provide a more complete assessment of transfer performance. For SVG, PAEM improves both vIoU@0.3 and vIoU@0.5 across all backbones, demonstrating that its spatial grounding gains persist under stricter overlap requirements. These results support transfer to both temporal and spatial localization beyond improvements in mean overlap.

Table 6: Detailed results on the HCSTVG-v1 test set and HCSTVG-v2 validation set (%). † denotes LM-only fine-tuning without PAEM.

	HCSTVG-v1				HCSTVG-v2			
Model	mtIoU	mvIoU	vIoU@0.3	vIoU@0.5	mtIoU	mvIoU	vIoU@0.3	vIoU@0.5
STVG-R1	56.9	39.1	66.7	38.6	61.3	40.8	67.9	38.8
InternVL-3.5-8B†	34.4	26.8	42.7	17.9	48.9	29.0	47.3	20.6
+PAEM	48.2	31.5	51.8	24.1	54.2	35.6	59.6	30.4
VideoLLaMA3-7B†	51.2	30.7	50.1	22.8	55.5	34.7	57.8	28.7
+PAEM	55.6	37.1	61.9	33.7	59.1	38.3	64.3	34.8
Qwen2.5-VL-7B†	52.4	33.6	55.9	27.2	57.9	34.5	57.6	28.9
+PAEM	58.7	41.6	70.0	41.2	64.0	42.0	71.7	41.0
Qwen3-VL-8B†	53.2	33.3	55.4	26.8	57.3	34.9	58.1	29.6
+PAEM	61.4	43.2	72.1	44.3	66.2	42.9	73.0	42.6

Table 7: Detailed zero-shot results on ST-Align (%). † denotes LM-only fine-tuning without PAEM.

	Spatio-Temporal Video Grounding (STVG)				Video Spatial Grounding (SVG)		
Model	tIoU@0.5	mtIoU	vIoU@0.5	mvIoU	vIoU@0.3	vIoU@0.5	mvIoU
STVG-R1	43.6	45.1	25.9	23.4	60.3	53.9	48.6
InternVL-3.5-8B†	28.8	32.7	13.6	19.4	44.8	30.7	29.9
+PAEM	33.5	36.1	19.4	22.7	51.9	37.2	34.7
VideoLLaMA3-7B†	27.6	31.9	17.8	21.1	54.3	43.8	39.4
+PAEM	29.8	33.4	21.6	22.9	58.6	47.1	42.6
Qwen2.5-VL-7B†	33.1	36.3	15.9	20.1	55.8	46.5	42.2
+PAEM	44.8	45.7	28.6	25.1	62.7	56.1	49.9
Qwen3-VL-8B†	37.6	39.9	13.8	18.4	54.9	45.6	41.4
+PAEM	47.1	47.2	27.3	24.9	61.5	54.2	48.7

A.4 Additional Qualitative Results

We present additional qualitative comparisons between Qwen2.5-VL-7B† and +PAEM across datasets with different target categories, query types, and event durations. The examples examine target discrimination, event coverage, and transfer to related grounding tasks. Throughout the figures, red and purple denote predictions from Qwen2.5-VL-7B† and +PAEM, respectively, while green denotes ground-truth annotations where shown.

A.4.1 Spatio-Temporal Grounding Cases

We first present cases from HCSTVG and ST-Align, covering crowded scenes, similar-looking entities, person interactions, and relational descriptions of human and non-human targets. These cases examine whether the model identifies both the queried entity and its corresponding event interval.

Entity disambiguation. Fig. 5 illustrates that a plausible temporal interval does not necessarily imply correct target selection. Nearby people may share similar appearances or participate in the same activity, making local visual cues insufficient for distinguishing them. PAEM more consistently selects the entity performing the described behavior, reducing confusion with nearby candidates. Temporal boundaries remain imperfect in some cases, but the spatial comparisons reveal improvements that interval overlap alone cannot capture.

Person interactions. Fig. 6 presents interactions and associated object actions that extend over time. The baseline predictions sometimes cover only part of the described process or select a separate activity before or after the annotated event. PAEM more closely covers the queried sequences, linking their constituent actions within a common interval. These comparisons highlight the importance of identifying the specific interaction described by the query and distinguishing it from surrounding activity.

ST-Align. Fig. 7 extends the comparison to human and non-human targets described through spatial relations and interactions. Such queries require distinguishing the referred entity from other participants mentioned in the same description. PAEM better identifies the entity occupying the queried

role and produces temporal intervals closer to the annotations. The examples therefore examine both relational target discrimination and the temporal coverage of the relevant activity.

A.4.2 Cross-Task Cases

We further examine transfer to non-human entity selection and temporal grounding without additional training on the target datasets. The following examples cover changing spatial configurations, sustained activities, and sequences of related actions, illustrating different demands on target identity and event extent estimation.

MeViS. Fig. 8 presents non-human targets among visually similar instances, with descriptions involving appearance, relative position, and motion. As the instances move, their positions and poses change, so the target cannot always be identified from a spatial cue in a single frame. PAEM maintains more consistent correspondence with the referred instance across the displayed frames, reducing confusion with neighboring candidates. Since the selected entity determines the mask trajectory used for segmentation, these visualizations illustrate the target-selection component of transfer to MeViS, with bounding boxes shown instead of masks for visual clarity.

TVGBench. Fig. 9 contains extended activities and sequences of related actions that share similar objects and visual context. This makes it necessary to distinguish the queried process from adjacent activities involving the same person. The baseline intervals sometimes include preceding content or end before the described process is complete. PAEM more closely captures the full event interval, reducing temporal shifts and incomplete coverage. The comparisons highlight the importance of assessing action continuity over the candidate span.

Charades examples. Fig. 10 presents indoor daily activities with varied durations and interactions with household objects. The cases include both brief object manipulations and sustained behaviors, requiring different temporal extents despite relatively stable scene content. The baseline predictions exhibit delayed starts, premature endings, or unnecessary temporal extension. PAEM produces intervals closer to the annotations, recovering omitted event portions while excluding more of the surrounding activity. These examples show that improved coverage involves adjusting both event boundaries, rather than simply favoring longer intervals.

A.5 Limitations

PAEM relies on pre-extracted entity tubes, making its predictions dependent on detection and tracking quality. If the target is missed or its trajectory contains identity switches, PAEM may lack a suitable candidate for matching. Since spatial outputs come from the selected tube, PAEM cannot directly correct errors in its bounding boxes. In addition, evaluating entity-span candidates and aggregating latent paths introduces computational overhead. Longer videos and larger candidate sets increase this cost, making efficient candidate selection and temporal processing important directions for future work.

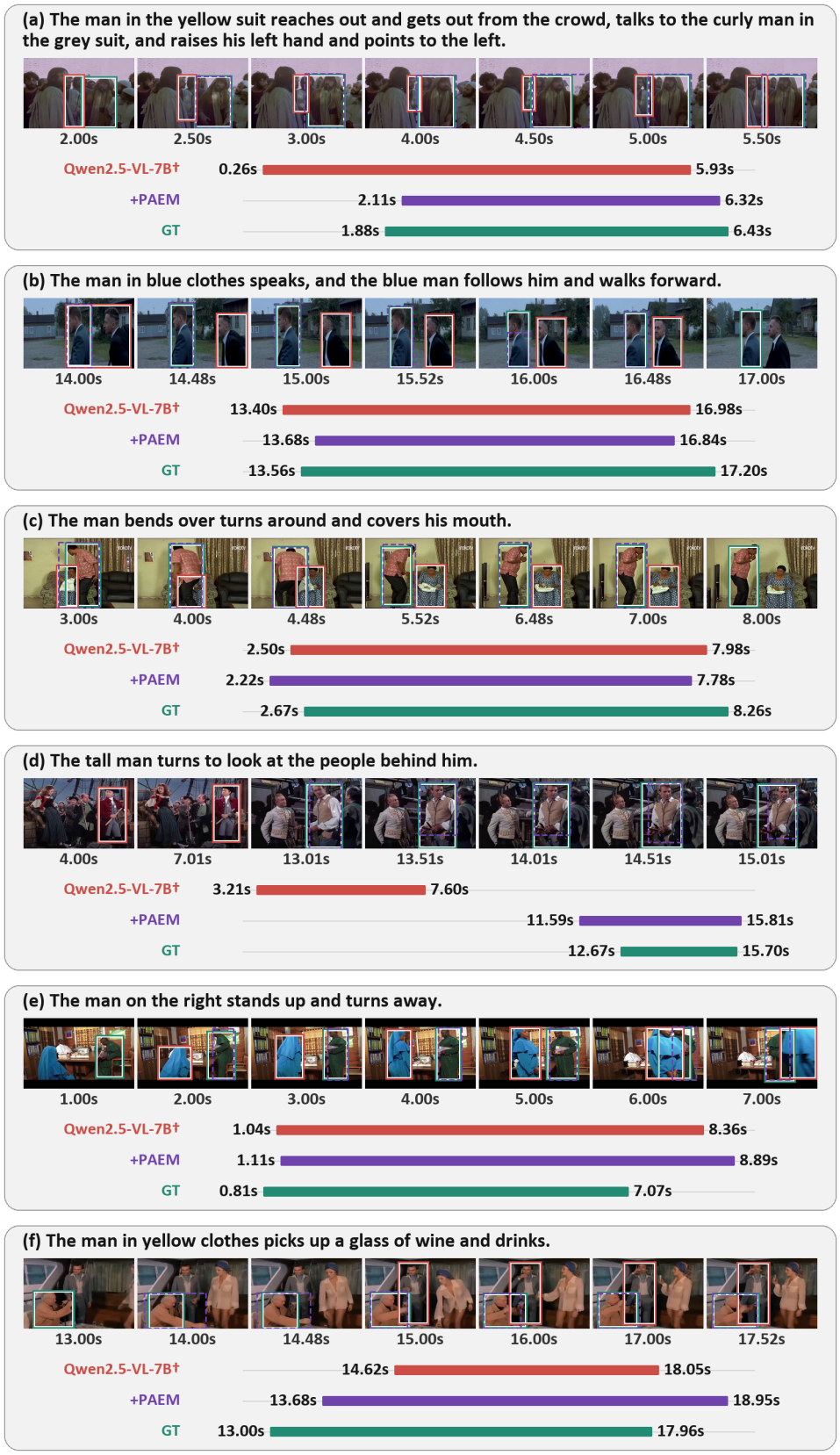


Figure 5: Entity disambiguation cases on HCSTVG.

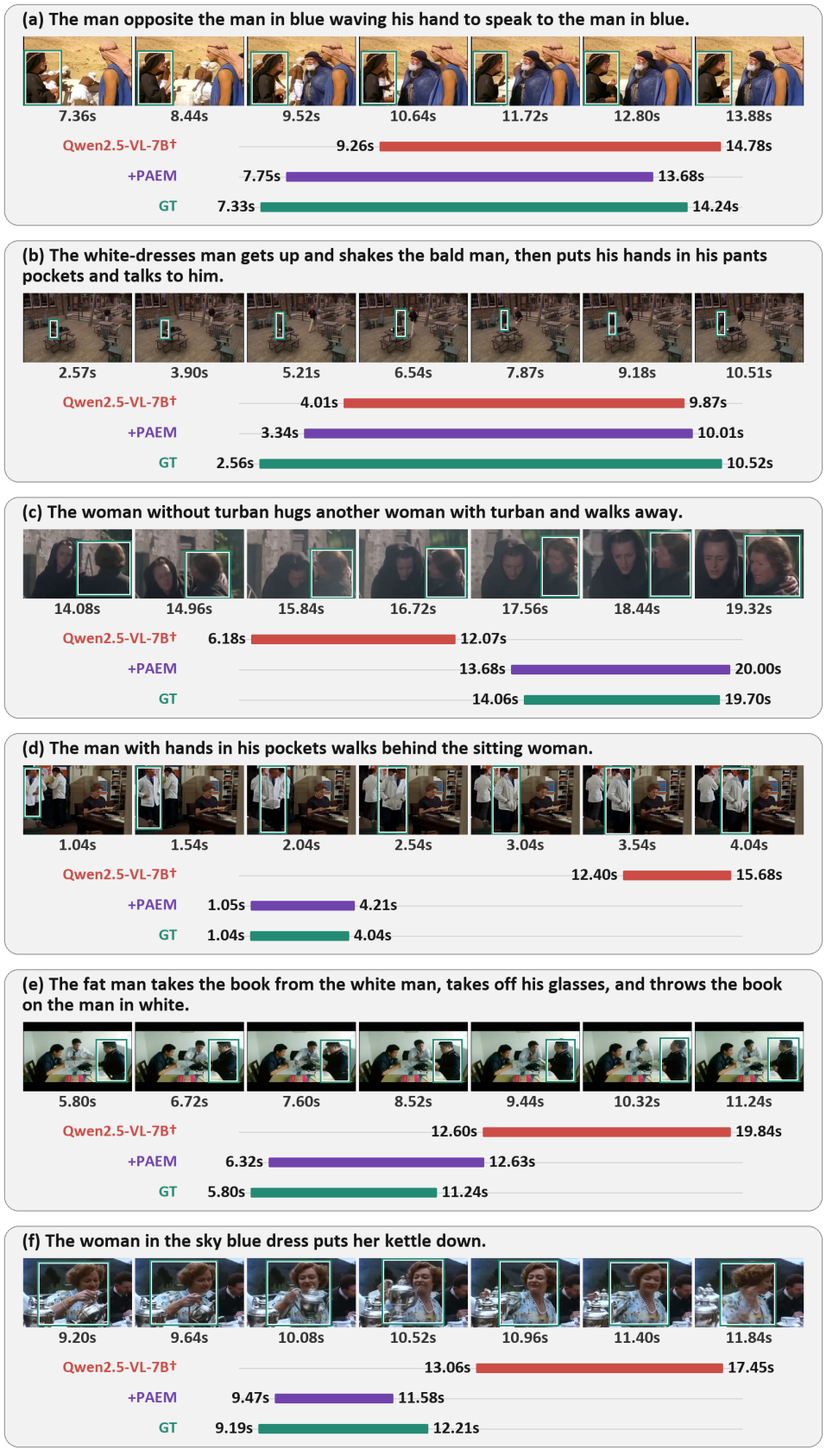


Figure 6: Interaction and action localization cases on HCSTVG.

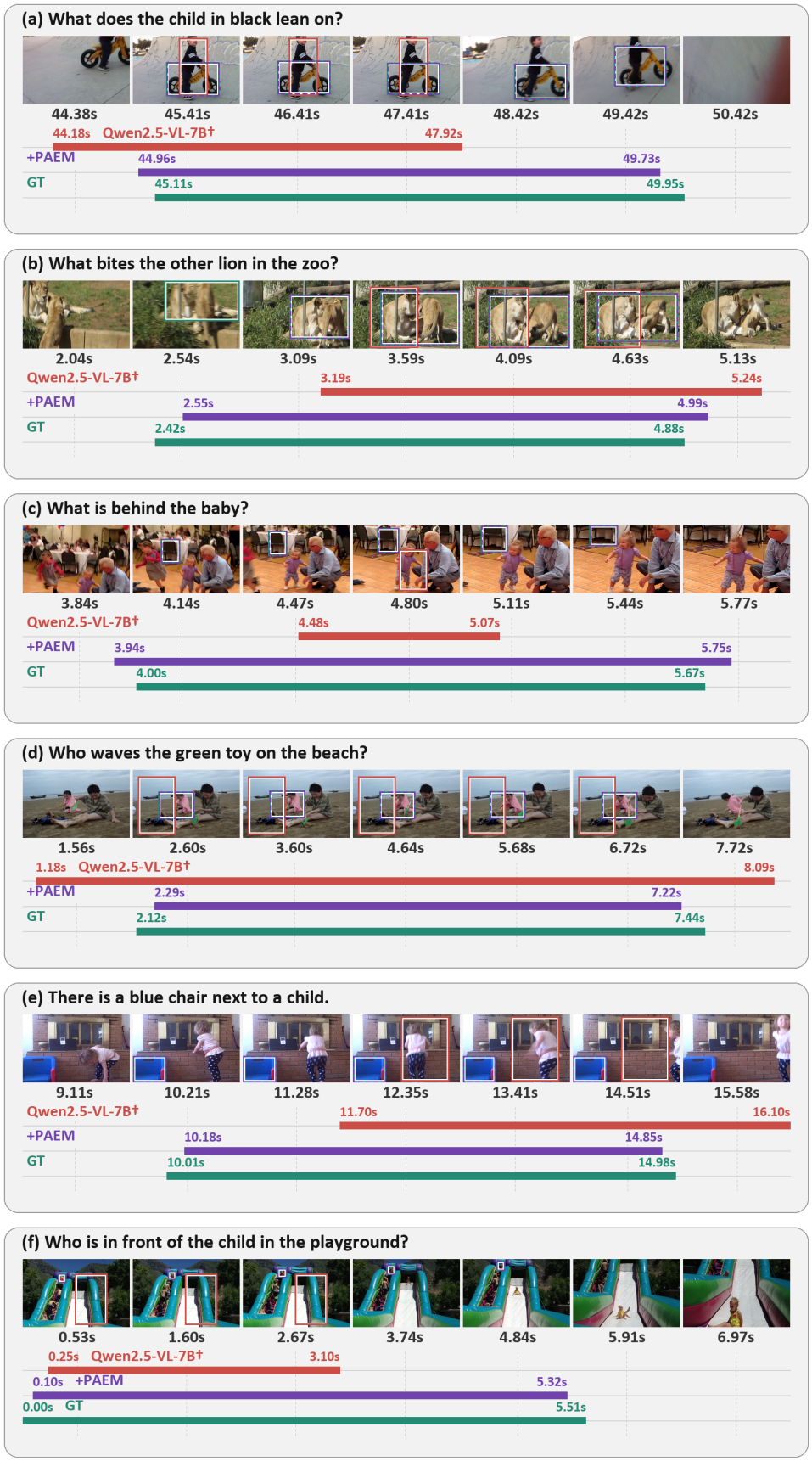


Figure 7: Spatio-temporal grounding cases on ST-Align.



Figure 8: Entity selection cases on MeViS.

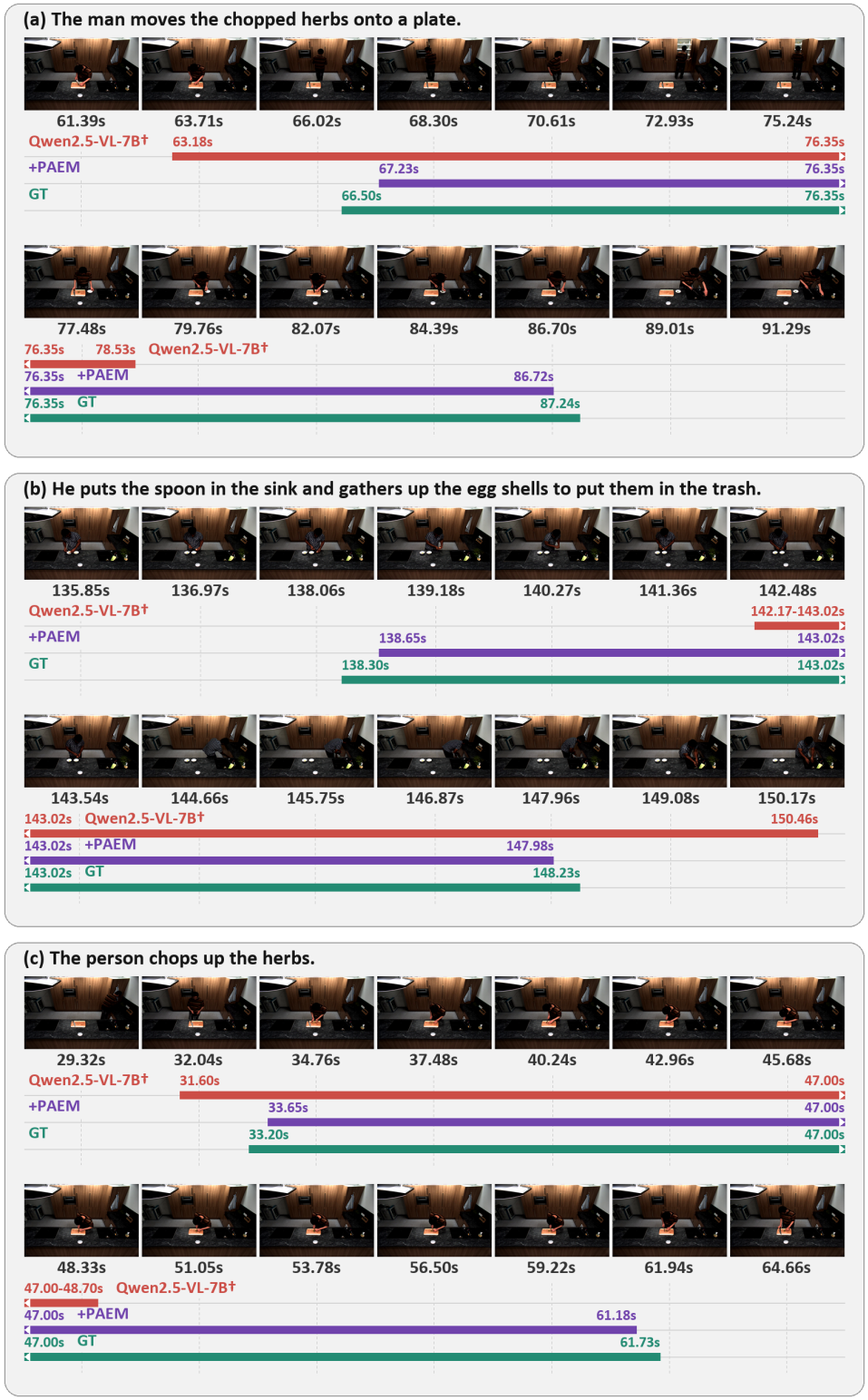


Figure 9: Temporal grounding cases on TVGBench.

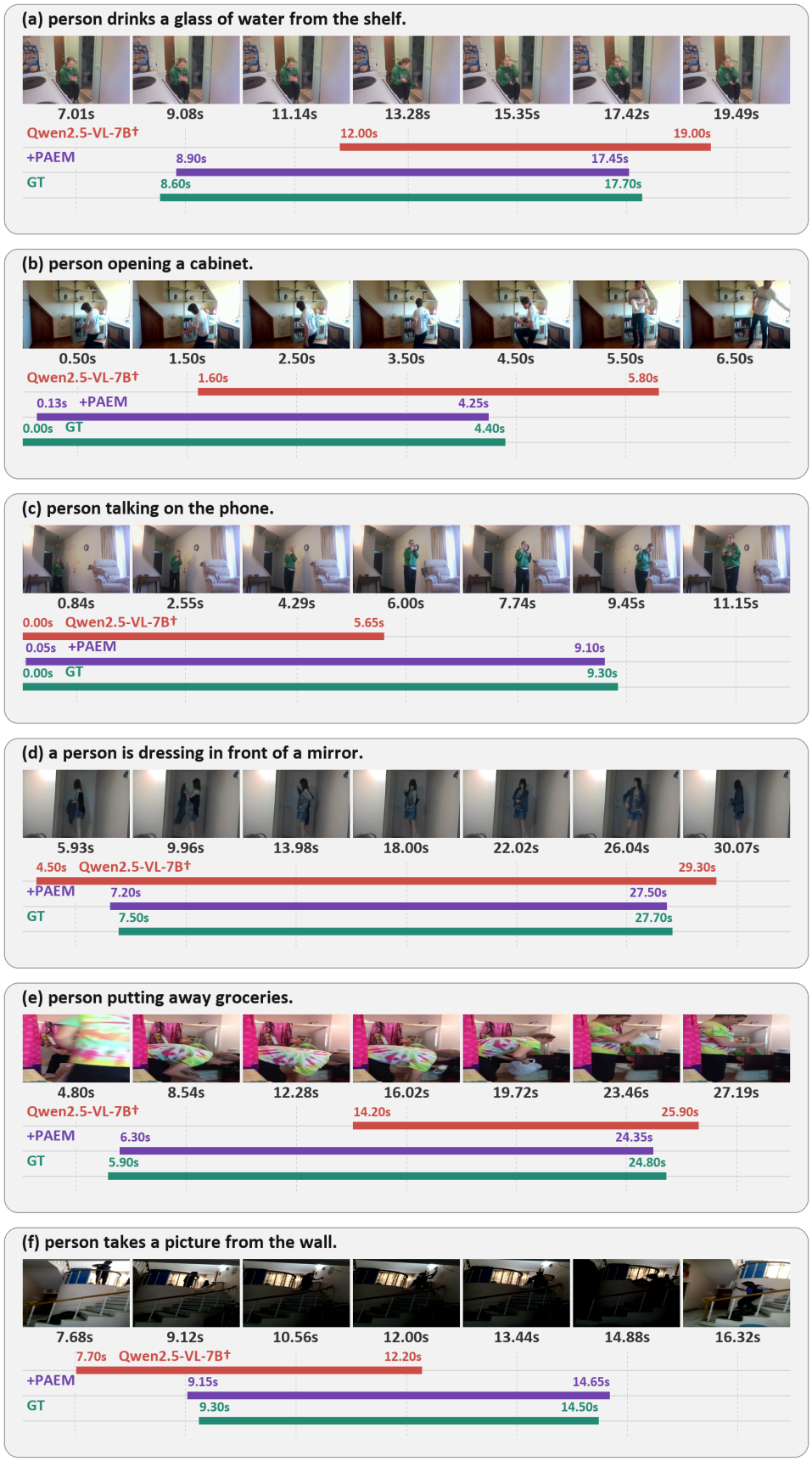


Figure 10: Temporal grounding cases from the Charades subset of TVGBench.