TAR: Temporal Anchor-Constrained Reasoning for Video Temporal Grounding

Chaohong Guo[1]*, Xun Mo[1]*, Yongwei Nie[1]†,
Fei Ma[2]†, Xuemiao Xu[1], and Chengjiang Long[3]‡

[1] South China University of Technology
[2] Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ)
[3] Bytedance Inc.
cguo5104@gmail.com nieyongwei@scut.edu.cn

Abstract. Video Temporal Grounding (VTG) aims to localize specific video segments corresponding to natural language queries. While recent Large Vision-Language Models (LVLMs) employ Reinforcement Learning to generate Chains-of-Thought (CoT), they typically rely solely on outcome-based supervision. Consequently, this often leads to hallucinations, where the reasoning process becomes disconnected from the visual content and the final prediction. Existing attempts to mitigate this by relying on external supervision from larger models or separate reward models are computationally expensive and prone to rigid patterns. To address these challenges, we propose **TAR** (**T**emporal **A**nchor-Constrained **R**easoning), a framework that introduces the temporal anchor (T-anchor) as a transparent and auditable checkpoint mechanism. T-anchor enforces progressive refinement within the CoT, compelling the model to continuously ground its intermediate thoughts in visual evidence and iteratively calibrate temporal predictions, thereby significantly enhancing the faithfulness and autonomy of the reasoning process and final accuracy. Furthermore, we introduce a bootstrapping paradigm that automatically harvests high-quality CoT data using only a standard 7B model, eliminating the dependency on ultra-large models. Extensive experiments demonstrate that TAR achieves state-of-the-art performance and generates faithful, autonomous, and progressively refined reasoning traces.

Keywords: Video Temporal Grounding · Reinforcement Learning

1 Introduction

Video Temporal Grounding (VTG) is a fundamental task in video understanding that requires models to localize specific segments within untrimmed videos corresponding to a given natural language description. Envisioning VTG technology empowering AI assistants to pinpoint specific moments (such as "a child

* Equal contribution.
† Corresponding author.
‡ This work was done when Dr. Chengjiang Long was with Meta.

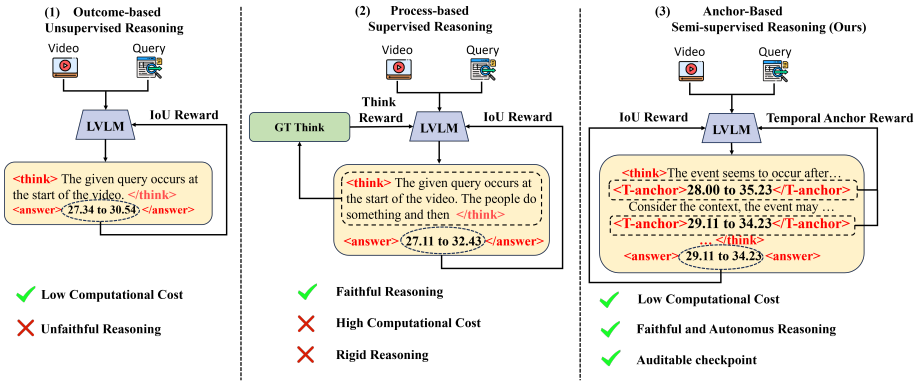

Fig. 1: Comparison of reasoning paradigms. (1) **Outcome-based Unsupervised Reasoning** relies solely on final outcome supervision, offering low computational cost but leading to unfaithful reasoning. (2) **Process-based Supervised Reasoning** depends on step-by-step supervision (GT Think), resulting in faithful reasoning but suffering from high computational costs and rigid patterns. (3) **Anchor-Based Semi-supervised Reasoning (Ours)** introduces auditable temporal anchors (`<T-anchor>`) as checkpoints. By utilizing a Temporal Anchor Reward, our method achieves faithful and autonomous reasoning while maintaining a low computational cost.

entering the kitchen") from hours of footage, this capability is critical for real-world applications including video surveillance, intelligent video retrieval, and human-computer interaction.

Existing VTG methods generally fall into three categories. First, Vision-Language Pre-training (VLP) methods extract features using pre-trained encoders and employ specific modules for fusion and localization. However, the disjoint training of feature extraction and localization often leads to error accumulation. Second, methods based on Supervised Fine-Tuning (SFT) of Large Vision-Language Models (LVLMs) typically rely on next-token prediction (cross-entropy loss). This objective aligns poorly with the Intersection-over-Union (IoU) metric, and since these models usually output direct answers without reasoning chains, they lack interpretability. Third, recent approaches utilize Reinforcement Learning (RL) to encourage models to generate a Chain-of-Thought (CoT) denoted as `<think>...</think>` before producing the final `<answer>`.

A core challenge remains: ensuring the reasoning process is tightly grounded in video content to effectively enhance localization accuracy. As illustrated in Figure 1 (1), methods [17, 36] like Time-R1 [32] rely solely on final prediction constraints driven by an IoU Reward. We term this paradigm *Outcome-based Unsupervised Reasoning*. While it boasts low computational cost, these methods often result in the "right answer for the wrong reason" phenomenon, where the generated reasoning suffers from hallucinations or generic descriptions detached from specific visual cues, leading to unfaithful reasoning. In contrast, as shown in Figure 1 (2), other approaches like Video-VER [23] which we catego-

rize as *Process-based Supervised Reasoning* typically rely on an external LVLM to generate ground truth thoughts (GT Think) and explicitly score the reasoning process via a reward to think process. While this achieves faithful reasoning, excessive reliance on such explicit supervision leads to high computational cost and rigid reasoning output patterns that lack autonomy, reducing the model to a template-matcher that merely parrots predefined structures (e.g., "The person is doing [Action]") to satisfy the external scorer.

To address these challenges, we propose **TAR** (**T**emporal **A**nchor-Constrained **R**easoning), as illustrated in Figure 1 (3), which serves as an efficient semi-supervised reasoning paradigm. The core innovation of TAR lies in embedding explicit `<T-anchor>...</T-anchor>` tags within the reasoning trace to serve as auditable checkpoints. Instead of relying on heavy external models to evaluate the intermediate steps, we design a progressive refinement strategy driven by a new reward mechanism. Specifically, alongside the standard IoU Reward for the final `<answer>`, we introduce a novel Temporal Anchor Reward applied directly to the intermediate `<T-anchor>` tags. During generation, the model first proposes a coarse interval at an initial anchor. It is then compelled to re-think and ground its subsequent reasoning in visual evidence, producing a more precise interval at the next anchor. By rewarding the model strictly for these incremental accuracy gains, our mechanism ensures that the reasoning process actively aligns with the visual content. Consequently, TAR achieves faithful and highly autonomous reasoning with minimal computational overhead.

Another significant challenge is that base models often struggle to strictly follow formatting instructions for T-anchors. Experiments show that even with explicit prompting, up to 76% of samples fail to generate valid T-anchors (e.g., unclosed tags), rendering anchor-based constraints ineffective and hindering training efficiency. Instead of distilling from larger models, we propose a bootstrapping strategy: performing lightweight RL directly on the 7B model to harvest $30k$ well-formatted reasoning chains as seed data for high-quality training.

Our main contributions are as follows:

- We propose the **TAR** framework, which embeds verifiable T-anchors into the reasoning chain to encourage the model to progressively refine its temporal predictions and continuously ground its intermediate thoughts in visual evidence. This effectively bridges the gap between the reasoning process and localization results, enhancing both faithfulness and accuracy.
- We introduce a resource-efficient bootstrapping data generation paradigm, eliminating the dependency on ultra-large models for generating high-quality reasoning chains.
- Experimental results demonstrate the superiority of TAR, establishing a new performance benchmark on the Charades-STA dataset with 61.1 mIoU and 50.2 R1@0.7.

2 Related Works

Temporal Video Grounding. Temporal Video Grounding (TVG) [7,15] aims to precisely locate video segments corresponding to text queries. Early methods mainly rely on Vision-Language Pre-training (VLP) models [3, 16, 22, 40], while subsequent approaches leverage large language models (LLMs) with fine-tuned visual encoders or open-source large vision-language models [9, 12, 28, 30, 39], though both lack interpretability and temporal awareness. Recently, reasoning and reinforcement learning (RL) have been introduced into LLMs [14, 35], with Time-R1 [32] achieving state-of-the-art grounding performance through a reasoning-driven LVLM and VideoChat-R1 [21] extending to broader spatio-temporal tasks, yet these RL-based methods still lack explicit constraints on the reasoning process, which our method addresses.

Reasoning in Large Vision-Language Models. Building on the success of reasoning models like OpenAI o1 and DeepSeek-R1 [10] in symbolic domains, recent efforts have extended Chain-of-Thought (CoT) capabilities to the visual domain [42, 43]. In the context of video understanding, existing reasoning-based methods generally follow two paradigms: (1) *Outcome-based Unsupervised Reasoning*: Recent methods such as TVG-R1 [4], Video-R1 [5], and Time-R1 [32] rely solely on final task-specific metrics (e.g., IoU) as rewards. While computationally efficient, this paradigm lacks intermediate supervision, often resulting in a disconnect between the reasoning process and the final answer. Consequently, the thinkprocess may suffer from hallucinations, yielding "the right answer for the wrong reason." (2) *Process-based Supervised Reasoning*: Some approaches [23, 31] incorporate additional reward models or external LLMs to evaluate the reasoning process itself. However, these model-based rewards are often computationally expensive and can lead to rigid, template-like reasoning patterns. To strike a balance, our TAR framework introduces a novel paradigm: *Anchor-based Constrained Reasoning*. Unlike unsupervised methods, we enforce intermediate verification using rule-based temporal anchors; unlike fully supervised methods, we maintain semantic autonomy by only constraining the anchor format rather than the entire linguistic content. This effectively combines the efficiency of rule-based rewards with the faithfulness of process-level supervision.

3 Methodology

Given an untrimmed video V and a query Q, Temporal Video Grounding (TVG) aims to predict the target segment $T = (t_s, t_e)$. We utilize a Large Vision-Language Model (LVLM) parameterized by θ to address this task. Instead of directly regressing the boundaries, we explicitly model the reasoning process by prompting the LVLM to generate a reasoning chain R, which includes a sequence of intermediate anchors $\mathcal{A} = \{A_1, ..., A_K\}$, followed by the final prediction A_{final}. The joint generation process is expressed as:

$$P_\theta(R, A_{final}|V, Q) = \prod_t P_\theta(y_t|y_{<t}, V, Q) \tag{1}$$

Here, the reasoning trace R is delimited by `<think>` tags, and each temporal anchor $A_i = (t_s^i, t_e^i)$ is enclosed in `<T-anchor>` tags.

First, Sec. 3.1 provides the intuition behind the effectiveness of the anchor mechanism. Next, Sec. 3.2 introduces the reward function for TAR. Finally, Sec. 3.3 details the bootstrapping data collection method used to generate the reasoning chains.

3.1 Intuition: Why Temporal Anchors Work.

Instead of relying on rigid token-level supervision or unconstrained generation, TAR leverages the inherent autoregressive nature of Large Vision-Language Models (LVLMs) to balance reasoning faithfulness and semantic autonomy. Purely unsupervised Chain-of-Thought (CoT) often suffers from "textual inertia" where the model pays more attention to its newly generated text than the visual input, leading to ungrounded hallucinations. The anchor mechanism acts as a structural interruption to break this inertia. Because our reward strictly penalizes any drop in localization accuracy, the model cannot simply guess the next timestamp based on text alone. It is forced to shift its attention back to the raw video tokens to find concrete evidence. Thus, each T-anchor serves as a "visual grounding checkpoint," ensuring the reasoning is deeply faithful to the actual video. At the same time, TAR preserves the model's semantic autonomy. Unlike supervised methods that reduce the model to a template-matcher forced to output fixed patterns (e.g., "The person is doing [Action]"), TAR only constrains the format of the numerical anchors. The natural language reasoning between these checkpoints remains completely free-form. This allows the model to autonomously explore and describe the video content without being restricted by rigid pseudo-labels.

In essence, this decoupled design of strict temporal anchor verification and free-form text generation empowers the model to achieve highly accurate localization while maintaining both robust reasoning faithfulness and genuine semantic autonomy.

3.2 The TAR Framework

Guided by the analysis in Sec. 3.1, we introduce the **TAR** framework. We employ Reinforcement Learning (specifically GRPO [10]) to optimize the model, using the proposed T-anchors as auditable checkpoints. We design a composite reward function $R(o)$ consisting of three components:

(1) Format Reward (r_{Format}). We prompt the model to output in the following structure: `<think>...<T-anchor>`
`...</T-anchor>...<T-anchor>...</T-anchor>... </think> <answer>...</answer>`.

In our `<think>` block, the model first provides an initial analysis of the video, then outputs a preliminary answer in the first `<T-anchor>` block. It then continues to analyze this initial answer in combination with its earlier reasoning and the video content, producing a refined or corrected answer in the second `<T-anchor>` block, and so on until it wants to stop. In the `<answer>` block, the

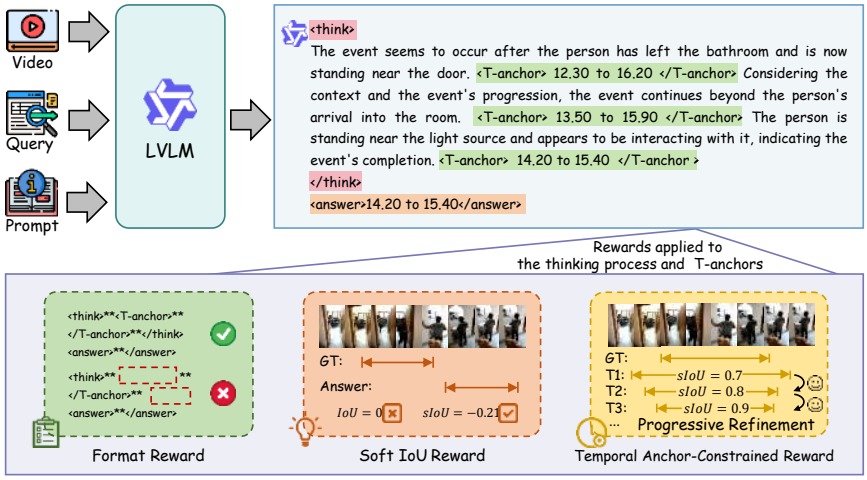

Fig. 2: Overview of our proposed Temporal Anchor-constrained Reasoning for Temporal Video Grounding (TAR) method. We apply a format reward, a soft IoU reward, and a temporal anchor-constrained reward to the think, answer and T-anchor tags generated by the LVLM.

model provides its final answer, which we set as the same in the last T-anchor tag in the think block. We use regular expressions to verify whether the model's output conforms to this required format. If the format is correct, we give a score of α to the model, otherwise 0:

$$r_{\text{Format}}(o) = \begin{cases} 0, \text{if } o \text{ does not match format} \\ \alpha, \text{if } o \text{ matches format} \end{cases} \tag{2}$$

where we use $\alpha = 3$, see Supplementary Material for ablation study.
(2) Soft IoU Reward (r_{sIoU}). In previous work, they usually adopt the following IoU reward:

$$r_{\text{IoU}}(o) = \frac{\max\left(0, \min(t_e, \hat{t}_e) - \max(t_s, \hat{t}_s)\right)}{\max(t_e, \hat{t}_e) - \min(t_s, \hat{t}_s)}. \tag{3}$$

In this paper, we propose a slightly different soft IoU to compute the reward for the model's output:

$$r_{\text{sIoU}}(o) = \frac{\min(t_e, \hat{t}_e) - \max(t_s, \hat{t}_s)}{\max(t_e, \hat{t}_e) - \min(t_s, \hat{t}_s)}. \tag{4}$$

The only difference is that the numerator has no lower bound of zero. Soft IoU can measure the distance between the predicted and ground-truth segments even when there is no overlap which the previous IoU cannot achieve, enabling stable

training. For example, in the early stages of training, if the model predicts a segment from 15.4s to 18.0s while the ground truth is from 3.4s to 12.2s, soft IoU yields a score of -0.21, whereas standard IoU gives 0. As a result, the standard IoU is not sensitive to the degree of misalignment in the predictions, leading to slower convergence.

(3) Temporal Anchor-Constrained Reward (r_{TAR}). Besides the standard format and soft IoU reward functions, we further design a temporal anchor-constrained reward to apply explicit constraints to the generated anchors. First, we identify format-correct `<T-anchor>` tags within the reasoning content and extract the time durations in-between them. Let o be the model output, s be the target number of `<T-anchor>` tags, and $\hat{s}$ be the actual number generated by the model. The r_{TAR} consists of three sub-terms:

Weighted Accuracy: We assign higher weights to later anchors to encourage convergence and enforce them to be more accurate towards the final prediction. We compute the weighted sum of the soft IoUs between the predicted time ranges and the ground truth:

$$\text{TAR}_{\text{sIoU}}(o) = \sum_{i=1}^{\hat{s}} i \cdot \text{sIoU}_i(o), \tag{5}$$

where $\text{sIoU}_i(o)$ computes the soft IoU score for the i-th anchor of output o, and i acts as the increasing weight.

Progressive Refinement: We explicitly reward the model only when it refines its prediction, requiring that later temporal anchors be more accurate than earlier ones. The progressive refinement step for the i-th anchor is formalized as:

$$\delta_i(o) = \begin{cases} 1 & \text{if sIoU}_i(o) > \text{sIoU}_{i-1}(o), \\ -1 & \text{otherwise.} \end{cases} \tag{6}$$

Based on this, the total progressive refinement reward is defined as:

$$\text{TAR}_{\text{refine}}(o) = \sum_{i=2}^{\hat{s}} \delta_i(o). \tag{7}$$

Length Penalty: If the size of $\hat{s}$ is unbounded, the model might engage in reward hacking by generating infinite anchors to accumulate TAR_{sIoU}, which increases the difficulty of progressive refinement and can ultimately degrade the model's training performance. To prevent $\hat{s}$ from deviating from the target count s, we impose a quadratic penalty:

$$\text{TAR}_{\text{num}}(o) = (\hat{s} - s)^2. \tag{8}$$

Finally, our complete temporal anchor-constrained reward is formulated as:

$$r_{\text{TAR}}(o) = \text{TAR}_{\text{sIoU}}(o) + \beta \cdot \text{TAR}_{\text{refine}}(o) - \gamma \cdot \text{TAR}_{\text{num}}(o), \tag{9}$$

where $\beta = 1$ and $\gamma = 5$ are hyperparameters balancing the refinement incentive and length constraint.

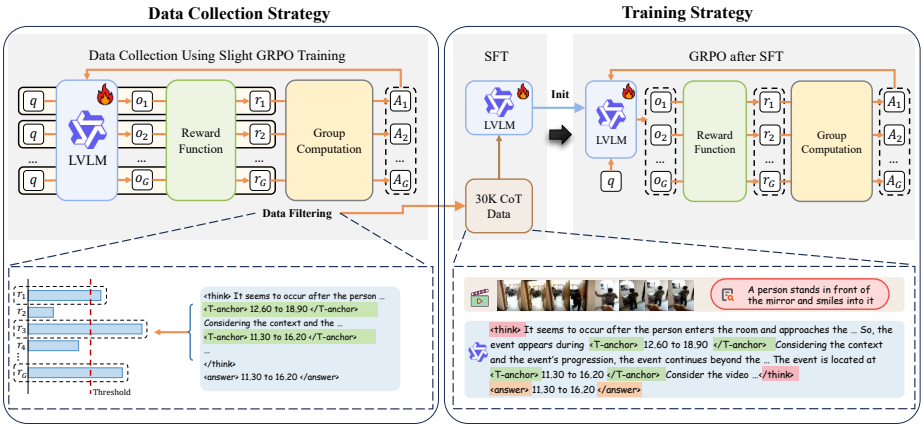

Fig. 3: Data collection strategy (left) and training strategy (right). We first employ a lightweight GRPO training to generate reasoning data, which is then filtered to obtain high-quality samples. Using this data, we perform Supervised Fine-Tuning (SFT) on the base Qwen2.5-VL model for a cold start, followed by final GRPO training on the SFT model.

The overall reward of the whole reinforcement learning is formulated as:

$$R(o) = r_{\text{Format}}(o) + r_{\text{sIoU}}(o) + r_{\text{TAR}}(o). \tag{10}$$

3.3 Bootstrapping Strategy for Data Efficiency

Training the proposed model directly is non-trivial. Despite extensive experimentation with various prompts to encourage the model to output `<T-anchor>` tags, it consistently struggles to generate them reliably. This is partly because the baseline model we use (i.e., Qwen2.5-VL-3B/7B) has limited capability in fully comprehending complex temporal reasoning prompts. When no `<T-anchor>` tags are produced, the temporal anchor-constrained reward cannot be computed, which prevents effective training.

To circumvent this, we introduce a Self-Bootstrapping Data Collection strategy via Slight GRPO Training (Fig. 3). Unlike conventional knowledge distillation methods that incur prohibitive financial and computational costs by relying on massive, proprietary closed-source models (e.g., GPT-4o or Gemini 1.5 Pro) for data generation, our approach is highly cost-effective. It leverages the baseline model's own exploratory capabilities to synthesize reasoning trajectories. Given a video V, prompt p, and query q sourced the training splits of dataset, the GRPO algorithm generates a diverse group of candidates o and computes their respective rewards r to optimize the policy.

To ensure the quality of the bootstrapped data, we apply a rigorous, deterministic filtering pipeline. From an initial pool of 186K raw exploratory outputs

generated from the training set, we distilled 30K high-quality reasoning samples using precise criteria (as illustrated in Fig. 3). Specifically, each retained trajectory must achieve a high total reward, which requires the reasoning trajectory to contain a minimum of `<T-anchor>` tags numbers. Crucially, each sequential anchor must demonstrate progressively increasing localization accuracy against the ground truth, with the specific accuracy thresholds (e.g., sIoU). This automated pipeline effectively bootstraps a high-quality supervision signal from extensive exploratory trajectories, completely eliminating the need for manual intervention or costly external large-model distillation.

4 Experiments

4.1 Experimental Setup

Benchmarks. We evaluate our model on four video grounding datasets: Charades-STA [7] (6,672 indoor activity videos, 30.6s avg, 12,408 train/3,720 test clip-query pairs), QVHighlights [16] (4,855 train/1,020 val video-text pairs that contain only a single segment), ActivityNet-Caption [2] (37,421 train/17,505 val/17,031 test samples), and TVGBench [32] (800 samples). The last one aggregates part test sets from ActivityNet-Caption, Charades-STA, HiREST [38], EgoNLQ [8] and TaCoS [29] as a comprehensive benchmark, evaluating 11 query types with balanced video-query distribution for fair assessment.

Implementation details. We use Qwen2.5-VL-3B/7B as our base models. We first generate and filter the 30K CoT data, and then train an SFT model using this data. Subsequently, we employ GRPO to further train the model on the corresponding datasets. All experiments are conducted using 8 $\times$ NVIDIA A100 GPUs. Further details regarding the training process, including SFT and GRPO configurations along with other experimental settings, can be found in the Supplementary Material.

4.2 Evaluation

We compare our TAR-TVG model with state-of-the-art TVG methods, including feature-based vision-language pretraining (VLP) models, large vision-language models fine-tuned through supervised fine-tuning (SFT), and recent approaches that leverage reinforcement learning (RL) to generate chain-of-thought reasoning.

Comparison on Charades-STA. As shown in Table 1, our TAR (7B) achieves the highest mIoU of 61.1 and R1@0.3 of 83.6 on the Charades-STA benchmark. This performance is achieved using only the Charades-STA training set, surpassing methods (denoted with *) that rely on additional training data, including but not limited to YT-Temporal [37], DiDeMo [1], QuerYD [27], InternVid [33], and HowTo100M [25]. For instance, in terms of R1@0.3, our model outperforms Time-R1 (82.8) trained with an additional 2.5k external samples, as well as all other types of TVG models. Our smaller 3B version of TAR surpasses

Table 1: Performance of temporal video grounding on Charades-STA, QVHighlights. Methods marked with * are trained with additional TVG datasets (See the Supplementary Material for their implementation and the dataset they use), while FT with ✓indicates that the model is fine-tuned on the corresponding Charades-STA or QVHighlights dataset. We compare our method against existing 3B, 7B open-source LVLMs, as well as state-of-the-art VLP models.

Type	Method	Size	FT	Charades-STA				QVHighlights			
				mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7
VLP	2D-TAN [41]	-	✓	-	57.3	45.8	27.9	-	-	-	-
	M-DETR [16]	-	✓	45.5	65.8	52.1	30.6	-	-	53.9	34.8
	UniVTG [22]	-	✓	50.1	70.8	58.1	35.6	-	-	59.7	-
	EaTR [13]	-	✓	-	-	68.4	44.9	-	-	61.4	45.8
	CG-DETR [26]	-	✓	50.1	70.4	58.4	36.3	-	-	67.4	52.1
	FlashVTG [3]	-	✓	-	-	70.32	49.87	-	-	<u>73.1</u>	<u>57.3</u>
SFT	ChatVTG [28]	7B		-	52.7	33.0	15.9	-	-	-	-
	TimeChat* [30]	7B	✓	-	32.2	13.4	36.2	-	-	-	-
	VTimeLLM [12]	7B		-	51.0	27.5	11.4	-	-	-	-
	TimeSuite [39]	7B		-	69.9	48.7	24.0	-	-	-	-
	TRACE* [11]	7B	✓	-	-	40.3	19.4	-	-	-	-
	TimeSuite* [39]	7B	✓	-	79.4	67.1	43.0	-	-	-	-
RL	VideoChat-R1.5* [36]	3B	✓	50.8	74.9	58.6	30.9	-	-	-	-
	Time-R1* [32]	3B	✓	<u>55.3</u>	<u>79.5</u>	<u>65.1</u>	<u>37.5</u>	-	-	-	-
	TAR (ours)	3B	✓	**57.0**	**81.0**	**67.2**	**40.6**	-	-	-	-
	Temporal-RLT* [17]	7B	✓	57.0	-	-	-	-	-	-	-
	VideoChat-R1.5* [36]	7B	✓	<u>60.6</u>	<u>82.8</u>	<u>71.6</u>	48.3	-	-	-	-
	Time-R1* [32]	7B	✓	58.8	<u>82.8</u>	**72.2**	<u>50.1</u>	<u>59.1</u>	<u>74.1</u>	66.2	52.7
	TAR (ours)	7B	✓	**61.1**	**83.6**	71.4	**50.2**	**65.9**	**85.6**	**76.1**	**58.5**

most VLP-based models and reaches performance comparable to some existing 7B models.

Comparison on QVHighlights. On the QVHighlights dataset, our method significantly outperforms most VLP-based models and reinforcement learning-based approaches (Table. 1). Specifically, we observe improvements of +6.8 mIoU, +9.9 R1@0.5, and +5.8 R1@0.7 compared to Time-R1, demonstrating strong generalization of our method between different TVG scenarios.

Zero-Shot Comparison on ActivityNet-Captions and TVGBench. In the zero-shot setting, we further evaluate our method on two challenging benchmarks: ActivityNet-Captions and TVGBench. Table 2 presents the results on ActivityNet-Captions, the experimental results show that our method achieves the highest R1@0.3 score (61.5) and the highest mIoU (41.1) under the zero-shot setting. Similarly, our approach achieves the highest mIoU of 30.6 on TVGBench.

Generalization to VQA. To evaluate the generalization capability of TAR-TVG on Video Question Answering (VQA) tasks, we allow the model to maintain its "Progressive Refinement" reasoning pattern: generating localization results as intermediate anchors within the <think> tag before producing the final categorical answer.

Table 2: Zero-shot comparison on the ActivityNet-Captions and TVGBench benchmarks. All models have 7B parameters.

Method	ActivityNet-Captions				TVGBench			
	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7
VideoChat [18]	7.2	8.8	3.7	1.5	-	-	-	-
Video-ChatGPT [24]	18.9	26.4	13.6	6.1	-	-	-	-
ChatVTG [28]	27.2	40.7	22.5	9.4	-	-	-	-
TimeChat [30]	-	36.2	20.2	9.5	-	22.4	11.9	5.3
HawkEye [34]	-	49.1	29.3	10.7	-	-	-	-
VTime-LLM [12]	-	44.0	27.8	14.3	-	-	-	-
Videochat-Flash [20]	-	-	-	-	-	32.8	19.8	10.4
TRACE [11]	-	-	-	-	-	37.0	25.5	14.6
TimeSuite [39]	-	-	-	-	-	31.1	18.0	8.9
VideoChat-R1.5 [36]	35.5	52.4	32.3	16.8	-	-	-	-
Temporal-RLT [17]	<u>39.0</u>	-	-	-	-	-	-	-
Time-R1 [32]	-	<u>58.6</u>	<u>39.0</u>	**21.4**	<u>29.2</u>	41.8	<u>29.4</u>	**16.4**
TAR(Ours)	**41.1**	**61.5**	**39.8**	<u>19.8</u>	**30.6**	**42.8**	**31.1**	<u>16.0</u>

Table 3: Comparison of Zero-shot VQA performance on MVbench [19] and VideoMME [6].

Model	**MVBench**	**VideoMME**
Qwen2.5-VL-7B	65.2	53.0
Time-R1	<u>65.6</u>	<u>54.2</u>
Ours	**66.1**	**54.9**

Experimental results demonstrate that, despite being trained on TVG data, TAR-TVG has successfully acquired a "Reasoning Pattern" of continuously refining anchors during the thinking process. This ability to self-correct through intermediate anchors can be directly transferred to VQA tasks to effectively assist logical judgment, as TVG is fundamentally a subtask of video understanding. Effective temporal grounding is crucial for VQA because it helps the model focus on the most relevant moments while reducing distractions from irrelevant or redundant content. Notably, our method outperforms baseline models such as Qwen2.5-VL-7B and Time-R1 in Zero-shot VQA performance.

4.3 Ablation Study and Discussion

We conduct a series of ablation studies on the 7B model.

Impact of Training Strategy. As shown in Table 4, incorporating the CoT data for supervised fine-tuning substantially enhances the model's temporal grounding ability during subsequent GRPO training. Training with reinforcement learning alone achieves only 45.9 R1@0.7, while applying chain-of-thought (CoT) supervision solely in the SFT stage yields just 21.0 R1@0.7, as

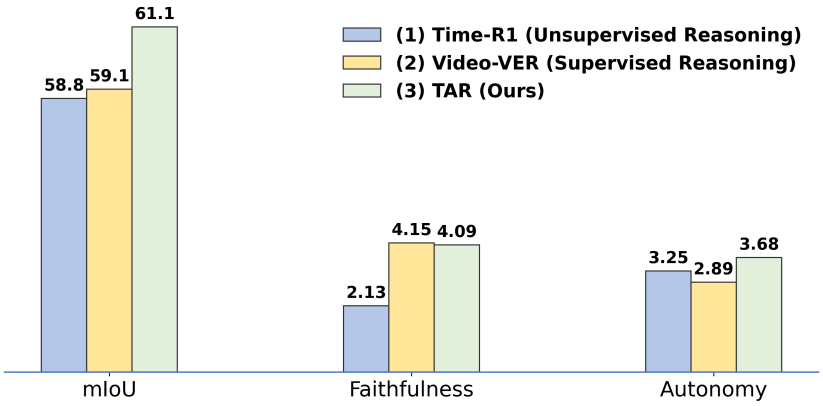

Fig. 4: Quantitative comparison of reasoning trajectories and grounding performance. Faithfulness and Autonomy are evaluated using Qwen2.5-VL-72B on a scale of 1 to 5. All methods are based on Qwen2.5-VL-7B. Our method (TAR-TVG) outperforms both Time-R1 and Video-VER, achieving the highest mIoU while maintaining robust reasoning faithfulness and enhanced autonomy.

SFT mainly focuses on token-level optimization and thus provides limited improvement in temporal understanding. However, when reinforcement learning is further applied to the SFT-trained model, both reasoning ability and video comprehension are strengthened, resulting in the best performance of 50.2 R1@0.7 and 61.1 mIoU.

Faithfulness and Autonomy of Reasoning Trajectories. As illustrated in Figure 4, we utilize Qwen2.5-VL-72B as an evaluator to score the reasoning trajectories of the three methods on a 5-point scale. Detailed evaluation protocols, including the specific prompts and scoring criteria, are provided in the supplementary material. Compared to Time-R1, which relies solely on outcome supervision, our method demonstrates superior faithfulness. Conversely, in contrast to Video-VER, which imposes strict supervision on the reasoning process, our approach exhibits enhanced autonomy. Ultimately, benefiting from our progressive refinement mechanism, our method achieves the highest mIoU. Furthermore, we provide concrete qualitative examples in the supplementary material to further illustrate the improved faithfulness and autonomy of our generated trajectories.

Ablation on TAR Rewards. Table 5 describes the ablation study on TAR rewards, involving TAR_{sIoU}, TAR_{num}, and $\text{TAR}_{\text{refine}}$. In all rows, TAR_{sIoU} is always employed because it is a soft IoU constraint applied to timestamp anchors. Omitting it may generate an overly wide time span and only slightly narrow it at each step. TAR_{num} is very useful, since omitting it the model may generate an excessive number of timestamp anchors, resulting in worse performance as shown in the first two rows. With TAR_{sIoU}, adding TAR_{num} achieves an R1@0.7

Table 4: Ablation on Training Strategy.

Type	mIoU	R1@0.3	R1@0.5	R1@0.7
+SFT	40.0	58.8	41.8	21.0
+GRPO	59.3	81.5	70.1	45.9
+SFT + GRPO	**61.1**	**83.6**	**71.4**	**50.2**

Table 5: Ablation on TAR_{sIoU}, TAR_{num} and $\text{TAR}_{\text{refine}}$ rewards.

TAR_{sIoU}	$\text{TAR}_{\text{refine}}$	TAR_{num}	mIoU	R1@0.5	R1@0.7
✓			39.1	50.2	31.2
✓	✓		41.1	51.3	32.4
✓		✓	58.5	70.2	47.9
✓	✓	✓	**61.1**	**71.4**	**50.2**

of 47.9. Combining all components reaches the highest 50.2, demonstrating their complementary effect.

Performance Gain from the Anchor Mechanism. To verify that the performance improvements stem directly from our proposed components, we compare our full TAR model (which incorporates the anchor reward detailed above) against two baselines: the standard Time-R1 (supervised solely by standard IoU) and a variant supervised by our proposed sIoU. As shown in Table 6, under identical training datasets and configurations, our complete method consistently outperforms both baselines across all metrics. This clearly demonstrates the distinct effectiveness of both the sIoU and the anchor mechanism.

Table 6: Ablation study on the performance gain from the anchor mechanism. All models are trained on the identical data scale.

Model Size	Method	R1@0.3	R1@0.5	R1@0.7	mIoU
3B	Time-R1	79.5	65.1	37.5	55.3
	Time-R1(+sIoU)	80.1	65.5	38.1	56.5
	TAR (Ours)	**81.0**	**67.2**	**40.6**	**57.0**
7B	Time-R1	82.4	70.6	47.5	59.8
	Time-R1(+sIoU)	82.8	70.6	48.1	60.0
	TAR (Ours)	**83.6**	**71.4**	**50.2**	**61.1**

Soft IoU vs Standard IoU. We compared soft IoU with the standard IoU in Figure 5. In the early stages of training, soft IoU can provide informative feedback even when the predicted and ground-truth segments do not overlap, offering meaningful rewards to guide optimization. As shown in Figure 5, sIoU consistently outperforms standard IoU throughout training and leads to better final performance.

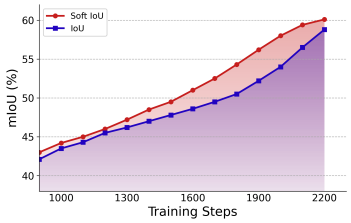

Fig. 5: Soft IoU vs Standard IoU.

About Thinking Length. We recorded the thinking lengths of models of different sizes. Figure 6 show that the 3B model generally produces longer thinking sequences compared to the 7B model, while 7B model has better grouding performance than 3B. For example, when only GRPO is applied, the 3B model generates on average 11.1 (186.1-175.0) more tokens than the 7B model. Second, we observe that after SFT, all models show improved performance. With SFT, the thinking length of the 3B model remains nearly unchanged. For the 7B model, the thinking length decreases from 175 tokens to 158.4. Both results indicate that stronger models require less thinking to achieve better performance.

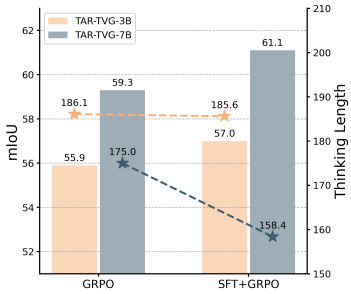

Fig. 6: The thinking length of different model size. The bar chart represents the mIoU, while the stars indicate the average number of tokens.

Failure Analysis. Out of 3,720 samples, we identify three major failure modes. Case 1 and Case 2 share the same quantitative behavior: the initial sIoU is 0 and remains 0 across subsequent anchors. However, they arise from different causes. In Case 1, the target action is visually similar to background activity, making it difficult to disambiguate the foreground segment. In Case 2, the target segment is extremely short, providing insufficient temporal evidence for the anchor to latch onto. Case 3 samples start near-optimal, e.g., with an sIoU of 0.87, and exhibit only minor boundary variation later, e.g., an sIoU of 0.83, with negligible impact on localization quality. Overall, 92.1% of samples are successfully refined, indicating that incorrect initial anchors are seldom reinforced into persistent errors; the remaining cases stem from visual ambiguity, extremely short targets, or near-optimal variation.

5 Conclusion

In this paper, we present **TAR**, a Temporal Anchor-constrained Reasoning framework that balances reasoning faithfulness with semantic autonomy in Video Temporal Grounding. By introducing *T-anchors* as auditable intermediate checkpoints, our method effectively breaks the textual inertia common in unsupervised reasoning while avoiding the rigidity of fully supervised approaches. Furthermore, we propose a resource-efficient bootstrapping paradigm that enables the model to autonomously harvest high-quality reasoning data, thereby eliminating the reliance on proprietary ultra-large models and significantly reducing computational overhead. TAR achieves state-of-the-art performance, notably reaching a record-high mIoU of 61.1 on the Charades-STA benchmark. This anchor-based constraint paradigm offers a promising direction for developing more interpretable and faithful reasoning model in the video domain.

Acknowledgments

This work was supported by the Guangdong Basic and Applied Basic Research Foundation (2025A1515011884, 2026A1515010184), the Open Research Fund from Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) (GML-KF-24-17), and the Research Task Assignment Project from Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) (GML-26420007).

References

1. Anne Hendricks, L., Wang, O., Shechtman, E., Sivic, J., Darrell, T., Russell, B.: Localizing moments in video with natural language. In: Proceedings of the IEEE international conference on computer vision. pp. 5803–5812 (2017)
2. Caba Heilbron, F., Escorcia, V., Ghanem, B., Carlos Niebles, J.: Activitynet: A large-scale video benchmark for human activity understanding. In: Proceedings of the ieee conference on computer vision and pattern recognition. pp. 961–970 (2015)
3. Cao, Z., Zhang, B., Du, H., Yu, X., Li, X., Wang, S.: Flashvtg: Feature layering and adaptive score handling network for video temporal grounding. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 9226–9236. IEEE (2025)
4. Chen, R., Fan, Z., Luo, T., Zou, H., Feng, Z., Xie, G., Zhang, H., Wang, Z., Liu, Z., Zhang, H.: Datasets and recipes for video temporal grounding via reinforcement learning. arXiv preprint arXiv:2507.18100 (2025)
5. Feng, K., Gong, K., Li, B., Guo, Z., Wang, Y., Peng, T., Wu, J., Zhang, X., Wang, B., Yue, X.: Video-r1: Reinforcing video reasoning in mllms. arXiv preprint arXiv:2503.21776 (2025)
6. Fu, C., Dai, Y., Luo, Y., Li, L., Ren, S., Zhang, R., Wang, Z., Zhou, C., Shen, Y., Zhang, M., et al.: Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 24108–24118 (2025)

7. Gao, J., Sun, C., Yang, Z., Nevatia, R.: Tall: Temporal activity localization via language query. In: Proceedings of the IEEE international conference on computer vision. pp. 5267–5275 (2017)
8. Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al.: Ego4d: Around the world in 3,000 hours of egocentric video. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18995–19012 (2022)
9. Guo, C., He, Y., Nie, Y., Ma, F., Xu, X., Long, C.: 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)
10. Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al.: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv preprint arXiv:2501.12948 (2025)
11. Guo, Y., Liu, J., Li, M., Liu, Q., Chen, X., Tang, X.: Trace: Temporal grounding video llm via causal event modeling. arXiv preprint arXiv:2410.05643 (2024)
12. Huang, B., Wang, X., Chen, H., Song, Z., Zhu, W.: Vtimellm: Empower llm to grasp video moments. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14271–14280 (2024)
13. Jang, J., Park, J., Kim, J., Kwon, H., Sohn, K.: Knowing where to focus: Event-aware transformer for video grounding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13846–13856 (2023)
14. Jin, B., Zeng, H., Yue, Z., Yoon, J., Arik, S., Wang, D., Zamani, H., Han, J.: Search-r1: Training llms to reason and leverage search engines with reinforcement learning. arXiv preprint arXiv:2503.09516 (2025)
15. Krishna, R., Hata, K., Ren, F., Fei-Fei, L., Carlos Niebles, J.: Dense-captioning events in videos. In: Proceedings of the IEEE international conference on computer vision. pp. 706–715 (2017)
16. Lei, J., Berg, T.L., Bansal, M.: Detecting moments and highlights in videos via natural language queries. Advances in Neural Information Processing Systems **34**, 11846–11858 (2021)
17. Li, H., Han, S., Liao, Y., Luo, J., Gao, J., Yan, S., Liu, S.: Reinforcement learning tuning for videollms: Reward design and data efficiency. arXiv preprint arXiv:2506.01908 (2025)
18. Li, K., He, Y., Wang, Y., Li, Y., Wang, W., Luo, P., Wang, Y., Wang, L., Qiao, Y.: Videochat: Chat-centric video understanding. arXiv preprint arXiv:2305.06355 (2023)
19. Li, K., Wang, Y., He, Y., Li, Y., Wang, Y., Liu, Y., Wang, Z., Xu, J., Chen, G., Luo, P., et al.: Mvbench: A comprehensive multi-modal video understanding benchmark. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22195–22206 (2024)
20. Li, X., Wang, Y., Yu, J., Zeng, X., Zhu, Y., Huang, H., Gao, J., Li, K., He, Y., Wang, C., et al.: Videochat-flash: Hierarchical compression for long-context video modeling. arXiv preprint arXiv:2501.00574 (2024)
21. Li, X., Yan, Z., Meng, D., Dong, L., Zeng, X., He, Y., Wang, Y., Qiao, Y., Wang, Y., Wang, L.: Videochat-r1: Enhancing spatio-temporal perception via reinforcement fine-tuning. arXiv preprint arXiv:2504.06958 (2025)
22. Lin, K.Q., Zhang, P., Chen, J., Pramanick, S., Gao, D., Wang, A.J., Yan, R., Shou, M.Z.: Univtg: Towards unified video-language temporal grounding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2794–2804 (2023)

23. Luo, M., Xue, Z., Dimakis, A., Grauman, K.: When thinking drifts: Evidential grounding for robust video reasoning. arXiv preprint arXiv:2510.06077 (2025)
24. Maaz, M., Rasheed, H., Khan, S., Khan, F.S.: Video-chatgpt: Towards detailed video understanding via large vision and language models. arXiv preprint arXiv:2306.05424 (2023)
25. Miech, A., Zhukov, D., Alayrac, J.B., Tapaswi, M., Laptev, I., Sivic, J.: Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2630–2640 (2019)
26. Moon, W., Hyun, S., Lee, S.B., Heo, J.P.: Correlation-guided query-dependency calibration in video representation learning for temporal grounding. CoRR (2023)
27. Oncescu, A.M., Henriques, J.F., Liu, Y., Zisserman, A., Albanie, S.: Queryd: A video dataset with high-quality text and audio narrations. In: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 2265–2269. IEEE (2021)
28. Qu, M., Chen, X., Liu, W., Li, A., Zhao, Y.: Chatvtg: Video temporal grounding via chat with video dialogue large language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1847–1856 (2024)
29. Regneri, M., Rohrbach, M., Wetzel, D., Thater, S., Schiele, B., Pinkal, M.: Grounding action descriptions in videos. Transactions of the Association for Computational Linguistics **1**, 25–36 (2013)
30. Ren, S., Yao, L., Li, S., Sun, X., Hou, L.: Timechat: A time-sensitive multimodal large language model for long video understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14313–14323 (2024)
31. Wang, X., Wang, P., Pei, J., Shen, W., Peng, Y., Hao, Y., Qiu, W., Jian, A., Xie, T., Song, X., et al.: Skywork-vl reward: An effective reward model for multimodal understanding and reasoning. arXiv preprint arXiv:2505.07263 (2025)
32. Wang, Y., Wang, Z., Xu, B., Du, Y., Lin, K., Xiao, Z., Yue, Z., Ju, J., Zhang, L., Yang, D., et al.: Time-r1: Post-training large vision language model for temporal video grounding. arXiv preprint arXiv:2503.13377 (2025)
33. Wang, Y., He, Y., Li, Y., Li, K., Yu, J., Ma, X., Li, X., Chen, G., Chen, X., Wang, Y., et al.: Internvid: A large-scale video-text dataset for multimodal understanding and generation. arXiv preprint arXiv:2307.06942 (2023)
34. Wang, Y., Meng, X., Liang, J., Wang, Y., Liu, Q., Zhao, D.: Hawkeye: Training video-text llms for grounding text in videos. arXiv preprint arXiv:2403.10228 (2024)
35. Wu, J., Deng, Z., Li, W., Liu, Y., You, B., Li, B., Ma, Z., Liu, Z.: Mmsearch-r1: Incentivizing lmms to search. arXiv preprint arXiv:2506.20670 (2025)
36. Yan, Z., Li, X., He, Y., Yue, Z., Zeng, X., Wang, Y., Qiao, Y., Wang, L., Wang, Y.: Videochat-r1. 5: Visual test-time scaling to reinforce multimodal reasoning by iterative perception. arXiv preprint arXiv:2509.21100 (2025)
37. Yang, A., Nagrani, A., Seo, P.H., Miech, A., Pont-Tuset, J., Laptev, I., Sivic, J., Schmid, C.: Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10714–10726 (2023)
38. Zala, A., Cho, J., Kottur, S., Chen, X., Oguz, B., Mehdad, Y., Bansal, M.: Hierarchical video-moment retrieval and step-captioning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23056–23065 (2023)

39. Zeng, X., Li, K., Wang, C., Li, X., Jiang, T., Yan, Z., Li, S., Shi, Y., Yue, Z., Wang, Y., Wang, Y., Qiao, Y., Wang, L.: Timesuite: Improving MLLMs for long video understanding via grounded tuning. In: The Thirteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=nAVejJURqZ
40. Zhang, S., Peng, H., Fu, J., Lu, Y., Luo, J.: Multi-scale 2d temporal adjacency networks for moment localization with natural language. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)
41. Zhang, S., Peng, H., Fu, J., Luo, J.: Learning 2d temporal adjacent networks for moment localization with natural language. In: Proceedings of the AAAI Conference on Artificial Intelligence (2020)
42. Zhang, X., Gao, Z., Zhang, B., Li, P., Zhang, X., Liu, Y., Yuan, T., Wu, Y., Jia, Y., Zhu, S.C., et al.: Chain-of-focus: Adaptive visual search and zooming for multimodal reasoning via rl. arXiv preprint arXiv:2505.15436 (2025)
43. Zhang, Y.F., Lu, X., Yin, S., Fu, C., Chen, W., Hu, X., Wen, B., Jiang, K., Liu, C., Zhang, T., et al.: Thyme: Think beyond images. arXiv preprint arXiv:2508.11630 (2025)