TEACHING WHAT TO LOOK FOR: PRIVILEGED PERCEPTION QUESTIONS FOR AGENTIC ON-POLICY DISTILLATION IN VIDEO CAPTIONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Detailed video captioning requires comprehensive and faithful descriptions of objects, actions, and temporal events. Fixed-frame sampling, however, can miss brief motions, small objects, and other fine-grained evidence, motivating captioning agents that actively seek relevant moments and inspect local regions. Training such behavior is challenging because, unlike video question answering, generic captioning instructions do not specify what visual evidence should be acquired. We propose Perception-Question On-Policy Distillation (PQ-OPD), which uses training-only Perception Questions (PQs) to guide active video perception. PQs specify what should be inspected without revealing the corresponding answers, encouraging the teacher to explore the video more comprehensively and focus on relevant visual evidence. Along trajectories generated by the student, on-policy distillation transfers this guidance to the student's reasoning, tool calls, and captions; no PQs are required at inference. Across three detailed-captioning and two caption-to-QA benchmarks, PQ-OPD consistently improves the corresponding untrained agents across teacher–student scales and input-frame budgets. Under our evaluation protocol, PQ-OPD achieves the strongest primary-metric results among the compared open-source models on DREAM-1K, CaReBench, ShortVidBench, MotionBench, and VidCapBench-AE. Our ablations, together with analysis of temporal coverage and answer leakage, further indicates that the student internalizes the training-time guidance as a more effective video exploration policy.

1 INTRODUCTION

Detailed video captioning requires comprehensive and faithful descriptions of objects, actions, and events over time (Wang et al., 2024a; Chai et al., 2025). Modern benchmarks make these requirements explicit (Wang et al., 2024a; Xu et al., 2026a; Chai et al., 2025; Yao et al., 2026; Zhou et al., 2018; Krishna et al., 2017): DREAM-1K (Wang et al., 2024a) evaluates the precision and recall of described events, while CaReBench (Xu et al., 2026a) separately evaluates dynamic actions and static objects. Recent progress has come from large-scale captioning data and temporal alignment (Yuan et al., 2025), efficient architectures (Chai et al., 2025), QA-based caption rewards (Xing et al., 2026a; Yang et al., 2026a), and structured spatio-temporal supervision (Xu et al., 2026b). These approaches, however, still perceive through fixed sets of sampled frames. These approaches still rely on frames selected before generation, which can miss brief motions, small objects, and precise event boundaries.

Active visual perception offers a natural alternative. Recent methods allow video models to seek relevant temporal intervals, revisit moments, and inspect local regions (Ding et al., 2026; Wang et al., 2026; Li et al., 2026b). Tool-using video agents have shown the effectiveness of such interaction for video question answering and reasoning (Wang et al., 2024c; Fan et al., 2025; Li et al., 2026a; Zhang et al., 2026; Ma et al., 2026; Xing et al., 2026b; Zhao et al., 2026b; Shi et al., 2026). Unlike video question answering, where the question itself points to relevant visual evidence, captioning is typically guided only by a generic instruction such as "describe the video in detail." Such an instruction specifies the desired output but provides little guidance on what visual evidence should be attended to. Effective visual evidence exploration is therefore crucial for agentic captioning. This raises a natural question: *can a captioning agent learn an effective policy for deciding what visual evidence to acquire under such weakly specified instructions?*

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.

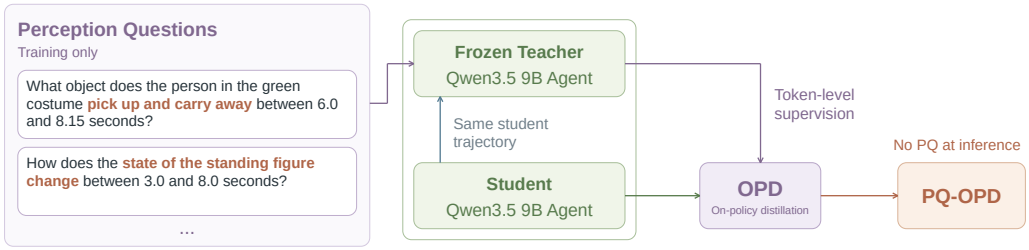


Figure 1: **Training with privileged Perception Questions.** The student generates its own tool-interaction trajectory, while a frozen teacher receives additional Perception Questions and provides token-level supervision. On-policy distillation transfers this guidance to the student, which receives no PQs at inference. The diagram illustrates the 9B teacher–student training.

One way to reduce this ambiguity is to make the observation targets explicit. We find that providing *Perception Questions* (PQs) encourages more active visual evidence exploration. For example, a PQ may ask "Which object is the person using between 2 and 4 seconds?" specifying what and when to inspect without revealing the answer. However, constructing PQs introduces an additional annotation or generation step for each video, making them undesirable as an inference-time dependency. We therefore aim to train the captioning agent to learn an effective visual exploration policy from PQ-guided supervision, while requiring only the generic captioning instruction at inference.

To this end, we propose **Perception-Question On-Policy Distillation (PQ-OPD)**, illustrated in Figure 1, based on on-policy distillation (OPD). The student generates its own trajectory from the generic captioning instruction, video duration, and tool specifications, acquiring evidence through its `seek`/`zoom` calls. A frozen teacher evaluates the same student-generated prefixes with PQs added to its context, enabling dense token-level supervision without a separate teacher rollout (Agarwal et al., 2024). A top-k approximation to Jensen–Shannon divergence (JSD) then supervises the student's reasoning, tool calls, and final caption. Since observation decisions are themselves generated tokens, the same objective trains both evidence acquisition and caption generation. Related self-distillation methods likewise show that students can benefit from training-only teacher context (Zhao et al., 2026a; Hübotter et al., 2026; Yuan et al., 2026).

We train PQ-OPD on 5,979 videos with 43,477 PQs and conduct PQ-free evaluation on three detailed video captioning benchmarks and two caption-to-QA benchmarks: CaReBench, DREAM-1K, VidCapBench-AE, ShortVidBench, and MotionBench. PQ-OPD consistently improves over the untrained Qwen3.5-9B agent across teacher–student scales and input-frame budgets. Under our evaluation settings, PQ-OPD achieves the strongest results among the compared open-source models on the primary metrics of all five benchmarks. We further conduct ablations to analyze PQ-OPD. The results show that training-time guidance about what to inspect can be learned by the captioning model and benefiting both tool-enabled agents and fixed-frame captioners.

Our contributions are summarized as follows:

- We introduce **Perception-Question On-Policy Distillation (PQ-OPD)**, an on-policy distillation framework that uses training-only Perception Questions to teach a video captioning agent what visual evidence to acquire through its own tool interactions.
- We construct 43,477 Perception Questions over 5,979 videos, covering objects, actions, and temporal developments, providing answer-free guidance for PQ-OPD training.
- We demonstrate consistent improvements across three detailed captioning and two caption-to-QA benchmarks, multiple teacher–student scales, and input-frame budgets, and show that PQ-OPD benefits both caption generation and interactive evidence acquisition.

2 Related Work

Detailed video captioning. Recent work on detailed video captioning has improved training data, temporal modeling, and fine-grained evaluation (Wang et al., 2024a; Yuan et al., 2025; Chai et al.,

2025; Xu et al., 2026a; Meng et al., 2026). Beyond supervised training, CapRL and CapRL++ (Xing et al., 2026a; Yang et al., 2026a) optimize descriptions through their utility for visual question answering, while PercepCap (Xu et al., 2026b) explicitly models spatio-temporal perception before caption generation. These approaches operate on visual inputs sampled before generation. PQ-OPD instead equips the captioner with `seek`/`zoom` tools and learns how to acquire visual observations during generation.

On-policy distillation. On-policy distillation supervises student-generated trajectories with dense token-level teacher distributions (Agarwal et al., 2024). Recent work (Zhao et al., 2026a; Hübotter et al., 2026; Li et al., 2026c; Zhou et al., 2026; Yang et al., 2026b) extends this paradigm through self-distillation, alternative objectives, and long-horizon training strategies. In multimodal settings, privileged teacher information has further been used to transfer fine-grained visual perception or structured feedback (Yuan et al., 2026; Tian et al., 2026; Yu et al., 2026; Liu et al., 2026; Lin et al., 2026; Wu et al., 2026). Our method applies on-policy distillation to video captioning, where a PQ-conditioned teacher provides token-level supervision along student-controlled video interactions.

Video agents. Recent video agents (Wang et al., 2024c; Fan et al., 2025; Ding et al., 2026; Li et al., 2026a; Wang et al., 2026; Li et al., 2026b; Zhang et al., 2026; Ma et al., 2026; Xing et al., 2026b; Zhao et al., 2026b; Shi et al., 2026) actively retrieve, focus, or refine visual evidence through iterative observation and tool use. These methods let the model adaptively select temporal intervals, revisit relevant moments, or inspect local regions instead of relying on a fixed visual input. Many further learn when and what to observe from interaction trajectories, supervised fine-tuning, or reinforcement learning. PQ-OPD extends this direction to open-ended video description through dense supervision from a PQ-conditioned teacher along the student's own interaction trajectories.

3 METHOD

3.1 PRELIMINARIES

In this section, we start with a brief review of relevant preliminaries. Let $\mathcal{D}$ denote the input distribution, and let p_θ and q_ϕ denote the student and teacher distributions, respectively. For an input x and an output sequence $y = (y_1, \ldots, y_T)$, the causal history at token t is $h_t = (x, y_{<t})$.

Off-Policy Distillation. Knowledge distillation transfers knowledge from a teacher model to a student by training the student to reproduce the teacher's outputs (Hinton et al., 2015). For language models, a common approach first collects responses generated by the teacher and then trains the student on these responses through supervised fine-tuning (SFT). At each position, the student is conditioned on the teacher-generated prefix $h_t = (x, y_{<t})$ and learns to predict the teacher's next token y_t. The objective is

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{x\sim\mathcal{D},\, y\sim q_\phi(\cdot|x)} \left[\frac{1}{T} \sum\nolimits_{t=1}^{T} \log p_\theta(y_t \mid h_t) \right]. \tag{1}$$

With access to the teacher's token probabilities, the student can match the teacher's next-token distribution at each position of these fixed responses. Writing $\mu(\cdot \mid x)$ for the fixed sequence distribution, this token-level distillation objective is

$$\mathcal{L}_{\mathrm{off}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\mu(\cdot|x)} \left[\frac{1}{T} \sum\nolimits_{t=1}^{T} D(q_\phi(\cdot \mid h_t), p_\theta(\cdot \mid h_t)) \right], \tag{2}$$

where D measures divergence between the teacher and student next-token distributions. Since the training sequences and their prefixes are drawn from a fixed data source independent of the current student, the objective is off-policy (Agarwal et al., 2024).

On-Policy Distillation. On-policy distillation lets the student generate its own responses and uses the teacher to provide supervision along these responses (Agarwal et al., 2024; Gu et al., 2024). For each student-generated prefix h_t, the teacher produces a next-token distribution, and the student learns to match this distribution. This provides a dense training signal at each position of the student's

own sequence. The objective is

$$\mathcal{L}_{\text{on}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim p_{\bar{\theta}}(\cdot|x)}\left[\frac{1}{T}\sum\nolimits_{t=1}^{T} D(q_\phi(\cdot \mid h_t), p_\theta(\cdot \mid h_t))\right], \tag{3}$$

where $\bar{\theta}$ denotes the current student parameters used to collect the sequences; the sampled histories are held fixed during each update. At each training step, the current student generates a new batch of responses for teacher supervision. During autoregressive generation, each predicted token becomes part of the context for subsequent predictions. Training on student-generated responses therefore allows the teacher to guide the continuation of sequences that the student itself produces (Agarwal et al., 2024). We use symmetric JSD as the divergence in our method, as detailed below.

3.2 Perception-Question On-Policy Distillation

Agentic video captioning. Let V be a video and x the input prompt including task instructions and tool specifications; the task is to produce a detailed caption of V, which the model emits as y. At interaction round r it may issue tool calls: $\texttt{seek}(t_{\text{start}}, t_{\text{end}})$ returns up to eight frames sampled uniformly from a time interval, and $\texttt{zoom}(t, \mathbf{b})$ returns a single spatial crop at time t, with $\mathbf{b}$ a normalized xyxy box in $[0, 1000]$. Inspection ends when the model emits y or when it reaches the round limit, and the returned images come from an environment $\mathcal{E}$, so we write a trajectory as

$$\tau = (x,\, z_1, a_1, o_1, \ldots, z_R, a_R, o_R,\, z_f, y), \tag{4}$$

where R is the number of tool-interaction rounds, z_r is the reasoning at round r, a_r denotes tool calls issued at round r, and $o_r = \mathcal{E}(V, a_r)$ the returned observations. The final reasoning z_f precedes the caption y.

A single autoregressive model generates z_r, a_r, and y from one distribution. We call the part of its behavior that decides what to inspect next and when to stop the *exploration policy*.

Perception-question guidance. For each training video V, we construct a set of Perception Questions $\mathcal{Q}(V) = \{q_1, \ldots, q_K\}$ from its visual content. These questions direct attention to visual aspects relevant to detailed captioning, including main objects, actions and interactions, and temporal developments such as action sequences, state changes, and repetition. Each question specifies what should be inspected and the temporal interval in which it can be resolved, while leaving the corresponding visual evidence unspecified, as illustrated in Figure 1. For example, a question may ask "Which object is the person using between 2 and 4 seconds?" without revealing the identity of the object. Taken together, the questions guide the model toward relevant visual evidence without specifying what content the video contains.

Notably, we do not provide PQ answers during training, as our purpose is to improve visual exploration rather than supply answers through text. We expose only what to inspect and when, while withholding the corresponding answers. Revealing the answers can create a textual shortcut, leading to less active visual exploration and fewer tool calls. This design is supported by prior work (Tian et al., 2026; Liu et al., 2025) and our answer-leakage study in Section 4.4, where providing PQ answers reduces the student's final caption quality.

The teacher and student are initialized from their respective pretrained Qwen3.5 checkpoints, and the teacher remains frozen throughout training. In the same-scale 9B $\rightarrow$ 9B setting, they share the same pretrained initialization; in cross-scale settings, they use checkpoints of different model sizes. For each student-generated trajectory τ, the student is conditioned on its causal history h_t, while the teacher receives the same history augmented with $\mathcal{Q}(V)$ (Section B.4 gives the exact template). At each assistant token t, they produce the next-token distributions $p_\theta(\cdot \mid h_t)$ and $q_{\phi_0}(\cdot \mid h_t, \mathcal{Q}(V))$, respectively. The teacher thus provides token-level supervision along the trajectory generated by the student, as illustrated in Figure 1.

On-policy training objective. At each training step, the student generates a batch of trajectories $\mathcal{B} = \{(V_i, x_i, \tau_i)\}_{i=1}^{B}$, with $\tau_i \sim P_{\bar{\theta}}(\cdot \mid V_i, x_i)$. Here $P_{\bar{\theta}}$ is the trajectory distribution induced by the student with sampling parameters $\bar{\theta}$ interacting with the video environment. The teacher provides token-level supervision along these trajectories. We minimize the token-averaged batch loss

$$\widehat{\mathcal{L}}(\theta; \mathcal{B}) = \frac{\sum_{i=1}^{B}\sum_{t\in\mathcal{A}(\tau_i)} D_{\text{JS}}^{(k)}(q_{\phi_0}(\cdot \mid h_{i,t}, \mathcal{Q}(V_i)), p_\theta(\cdot \mid h_{i,t}))}{\sum_{i=1}^{B}|\mathcal{A}(\tau_i)|}, \tag{5}$$

where $h_{i,t}$ is the causal history before assistant token t in trajectory τ_i, including the prompt, preceding assistant tokens, and tool observations returned so far. The set $\mathcal{A}(\tau_i)$ contains all assistant-token positions across reasoning, tool calls, and the final caption. $D_{\mathrm{JS}}^{(k)}$ denotes symmetric JSD computed on the teacher's top-k tokens and a tail bucket containing the remaining probability mass, with $k = 100$. Gradients are computed only through the student distribution, with sampled trajectories, observations, and teacher distributions held fixed during each update.

4 EXPERIMENTS

4.1 SETUP

Training data. We construct 43,477 Perception Questions for 5,979 training videos from Tarsier2-Recap-585K (Yuan et al., 2025) using Gemini 3.1 Pro (Google, 2026), averaging 7.27 questions per video. The prompt template is provided in Section B.1. Generated answers and evidence descriptions are retained only for quality control and analysis and are excluded from the training.

Model training. Our primary setting uses Qwen3.5 9B for both student and teacher, with the teacher frozen at initialization. We train on all 5,979 videos and 43,477 PQs for 372 optimizer steps with a peak learning rate of 2×10^{-6}, while freezing the vision encoder and aligner. We additionally train models under 27B $\rightarrow$ 9B and 9B $\rightarrow$ 4B teacher–student settings, initializing each model from the Qwen3.5 checkpoint of its corresponding size and keeping the teacher frozen. We approximate full-vocabulary JSD using the teacher's top-k support with $k = 100$ and aggregate the remaining probability mass into a single tail bucket. Full training hyperparameters are provided in Section A.

Benchmarks and inference setting. We evaluate caption quality on CaReBench (Xu et al., 2026a), DREAM-1K (Wang et al., 2024a), and VidCapBench-AE (Chen et al., 2025). CaReBench measures Object and Action content, DREAM-1K evaluates event coverage with AutoDQ, and VidCapBench-AE assesses detailed caption quality across multiple dimensions. As the original judges used by CaReBench and DREAM-1K are no longer available through public API services, we use GPT-4.1 (OpenAI, 2025) as the judge for both benchmarks; VidCapBench-AE uses GPT-4o (OpenAI, 2024). We further evaluate caption-to-QA performance on MotionBench and ShortVidBench by replacing the video with the generated caption. Gemini 3.1 Pro (Google, 2026) first answers each question from the caption, and a separate independent call evaluates the accuracy of the resulting answer. At inference time, the agent receives no PQs and may use up to 16 rounds of `seek`/`zoom` calls within a 128K context and a 128 total input frames limit. `seek` returns up to eight frames and `zoom` returns one crop, with each image limited to 460,800 pixels.

4.2 MAIN RESULTS

Table 1 and Table 2 compare PQ-OPD with direct video models and the untrained Qwen3.5 9B Agent across fine-grained captioning and caption-to-QA benchmarks. Qwen3.5 9B uses 32-frame direct video input, while the agent variants acquire visual observations through tools. VideoCap-R1 (Meng et al., 2026) has no publicly released weights, so we reproduced it following the training recipe in its paper and evaluated the reproduced model under the same protocol.

Fine-grained caption quality. Across DREAM-1K, CaReBench, and VidCapBench-AE, PQ-OPD consistently improves caption quality. The 27B $\rightarrow$ 9B setting achieves the best open-source Event F1 on DREAM-1K (33.4) and the best Action and Object F1 on CaReBench (40.6 and 41.7). On VidCapBench-AE, it also obtains the highest open-source overall accuracy (20.8) and precision (59.6), the highest video-aesthetics accuracy (18.0), and tied-highest video-content accuracy (21.5) among the compared open-source models. The 9B $\rightarrow$ 9B setting performs particularly well on dynamic content, reaching the best open-source motion accuracy (13.8) and physical-law accuracy (37.2).

Caption-to-QA evaluation. We further evaluate the generated captions by using them in place of the original video input on ShortVidBench (Ge et al., 2025) and MotionBench (Hong et al., 2025). The 27B $\rightarrow$ 9B model achieves 78.5 accuracy on ShortVidBench and 59.6 on MotionBench, obtaining the best open-source results on both benchmarks. It also surpasses the listed proprietary models on MotionBench and performs comparably on ShortVidBench.

Table 1: **Evaluation results on DREAM-1K (Wang et al., 2024a), CaReBench (Xu et al., 2026a), ShortVidBench (Ge et al., 2025), and MotionBench (Hong et al., 2025).** For PQ-OPD, the model size denotes the student; the teacher size is given in parentheses. Scores are reported as percentages. We highlight the **best** and second-best available open-source results.

Model	DREAM-1K	CaReBench		ShortVidBench	MotionBench
	Event F1/R/P	Action F1/R/P	Object F1/R/P	Acc.	Acc.
Proprietary models (fixed 32 frames)					
Gemini 3 Flash (Google, 2025)	37.6/48.5/30.7	42.4/39.6/45.7	40.3/37.8/43.2	78.6	57.6
Gemini 3.1 Pro (Google, 2026)	38.2/45.2/33.1	42.0/37.4/47.7	39.3/32.6/49.6	78.4	55.9
Open-source models (fixed 32 frames)					
Qwen2.5-VL 7B (Bai et al., 2025b)	27.2/27.9/26.6	30.9/23.6/45.0	34.1/27.6/44.6	70.0	50.7
Kimi-VL A3B (Du et al., 2025)	27.5/32.6/23.8	35.1/29.0/44.4	37.7/32.0/45.8	74.2	52.1
Qwen3-VL 8B (Bai et al., 2025a)	29.1/36.3/24.3	35.9/31.2/42.2	38.6/35.4/42.5	72.8	53.0
Qwen3.5 9B (Qwen Team, 2026)	30.3/40.5/24.1	36.9/34.8/39.3	38.3/42.1/35.1	74.8	56.6
VideoCap-R1 (Meng et al., 2026)	28.8/32.0/26.1	33.5/26.6/45.2	33.8/26.1/**48.0**	71.1	53.0
OwlCap-7B (Zhong et al., 2026)	27.2/28.2/26.2	33.7/27.3/44.0	34.5/26.9/47.9	70.3	53.1
Agent models (with seek/zoom tools)					
Qwen3.5 9B Agent	31.3/38.9/26.1	40.0/**37.4**/43.0	39.6/**47.1**/34.1	75.0	57.0
PQ-OPD 4B (9B Teacher)	32.6/39.4/**27.8**	38.9/34.8/44.3	41.4/41.5/41.4	77.0	57.9
PQ-OPD 9B (9B Teacher)	33.3/41.5/**27.8**	40.1/36.6/44.3	41.0/41.6/40.5	76.1	59.0
PQ-OPD 9B (27B Teacher)	**33.4/42.3**/27.6	**40.6**/36.3/**46.0**	**41.7**/44.4/39.3	**78.5**	**59.6**

Table 2: **Evaluation results on VidCapBench-AE (Chen et al., 2025).** Each group reports Accuracy/Precision/Coverage/Conciseness, as percentages. For PQ-OPD, the model size denotes the student and parentheses specify the teacher. The **best** and second-best available open-source results are highlighted for each metric.

Model	VidCapBench-AE				
	Overall	Video Aesthetics	Video Content	Video Motion	Physical Laws
	Acc./Pre./Cov./Con.	Acc./Pre./Cov./Con.	Acc./Pre./Cov./Con.	Acc./Pre./Cov./Con.	Acc./Pre./Cov./Con.
Proprietary models (fixed 32 frames)					
GPT-4o (OpenAI, 2024)	16.8/57.4/86.0/5.9	14.1/47.6/83.4/4.9	17.5/61.7/87.2/6.1	10.2/41.3/84.0/3.6	27.9/62.1/85.4/9.7
Gemini 3 Flash (Google, 2025)	19.3/58.1/90.5/7.3	17.1/47.3/87.5/6.4	19.6/62.8/91.7/7.4	12.9/49.5/89.8/4.9	33.9/58.4/91.7/12.8
Open-source models (fixed 32 frames)					
Qwen2-VL 7B (Wang et al., 2024b)	11.1/47.1/77.0/6.4	12.4/44.3/78.7/7.2	9.9/48.3/75.9/5.7	4.0/22.7/78.2/2.3	26.1/**59.4**/81.2/15.1
Kimi-VL A3B (Du et al., 2025)	15.4/54.5/85.9/2.8	13.5/46.9/81.1/2.4	15.7/58.3/87.7/2.8	6.2/36.5/85.3/1.1	28.1/52.0/89.4/5.1
MiMo-VL (Yue et al., 2025)	15.8/52.3/86.8/7.5	14.5/44.9/85.0/6.9	15.8/55.8/87.8/7.5	8.4/36.1/84.9/4.0	28.4/56.8/85.4/13.6
Intern2VL 8B (Chen et al., 2024)	10.2/43.0/84.9/2.5	9.1/36.3/84.4/2.2	10.0/46.1/85.2/2.4	4.4/18.0/81.3/1.1	23.6/52.8/85.7/5.8
Qwen3-VL 8B (Bai et al., 2025a)	17.8/56.8/88.5/4.4	14.5/47.3/84.8/3.6	18.6/61.1/90.3/4.6	9.3/39.2/84.0/2.3	32.7/57.3/87.7/8.1
Qwen3.5 9B (Qwen Team, 2026)	17.2/52.7/88.5/5.8	15.3/44.5/85.5/5.2	17.4/56.7/89.6/5.9	9.3/37.6/91.1/3.2	31.2/52.2/89.4/10.5
VideoCap-R1 (Meng et al., 2026)	14.2/50.1/84.5/**8.5**	12.3/42.9/82.7/**7.4**	14.1/53.3/85.0/**8.5**	7.6/35.9/85.3/**4.6**	31.9/55.0/87.7/**19.3**
OwlCap-7B (Zhong et al., 2026)	15.9/52.8/84.9/6.1	14.0/44.1/83.2/5.4	15.9/56.8/85.5/6.1	9.8/36.6/84.9/3.8	31.2/54.7/88.2/12.0
Agent models (with seek/zoom tools)					
Qwen3.5 9B Agent	19.9/58.2/91.4/4.4	15.3/46.7/86.1/3.4	21.3/63.5/93.7/4.7	8.0/36.8/89.3/1.8	33.9/57.5/91.0/7.5
PQ-OPD 4B (9B Teacher)	19.6/57.8/90.7/4.2	16.7/47.9/85.5/3.6	20.1/62.4/92.8/4.3	11.1/40.0/93.3/2.4	36.7/58.1/**93.5**/7.9
PQ-OPD 9B (9B Teacher)	20.6/58.7/91.2/4.6	16.6/48.5/**87.0**/3.7	**21.5**/63.5/92.9/4.8	**13.8**/40.4/**94.7**/3.1	**37.2**/59.3/92.0/8.3
PQ-OPD 9B (27B Teacher)	**20.8/59.6/91.8**/4.8	**18.0/49.6**/86.6/4.2	**21.5/64.2/94.0**/5.0	12.9/**48.5**/89.8/3.0	33.4/57.2/92.2/7.8

Distillation across model scales. The improvements are consistent across teacher–student scales. With a 9B teacher and student, PQ-OPD improves DREAM-1K Event F1 from 31.3 to 33.3, CaReBench Object F1 from 39.6 to 41.0, ShortVidBench accuracy from 75.0 to 76.1, and MotionBench accuracy from 57.0 to 59.0. The 4B student also outperforms the untrained 9B agent on DREAM-1K, ShortVidBench, and MotionBench, showing that the behavior can be transferred to a smaller student.

Table 3: Ablation study of PQ guidance and training strategies on the 9B backbone. CaReBench F1 is the arithmetic mean of Action F1 and Object F1; DREAM-1K F1 is event F1. "Train PQ" indicates that PQs are used during training, while "Infer. PQ" indicates that video-specific PQs are provided at inference. Rows #5–#8 receive no PQs at inference.

#	Model Variant	Train PQ	Infer. PQ	CaReBench F1	DREAM-1K F1
Fixed observations (no seek/zoom tools)					
#1	Baseline (Qwen3.5 9B)	–	–	37.6	30.3
#2	+ PQ at inference	–	✓	37.4	33.4
#5	+ PQ-OPD	✓	–	38.0	31.9
Interactive observations (seek/zoom tools)					
#3	Base agent (Qwen3.5 9B)	–	–	39.8	31.3
#4	+ PQ at inference	–	✓	40.6	33.4
#6	+ PQ trace SFT	✓	–	39.7	32.6
#7	+ Shuffled-PQ-OPD	✓	–	40.1	31.5
#8	+ PQ-OPD (Ours)	✓	–	**40.6**	**33.3**

4.3 Ablation Study

Effect of PQ guidance. For the inference-only variants (#2 and #4), PQs are generated from each CaReBench or DREAM-1K video using the same construction procedure as the training PQs. Providing these PQs at inference substantially improves DREAM-1K in both direct captioning and agentic settings. From #1 to #2, DREAM-1K F1 increases from 30.3 to 33.4, while CaReBench remains comparable. With `seek`/`zoom` tools, #3 to #4 improves DREAM-1K F1 from 31.3 to 33.4 and CaReBench F1 from 39.8 to 40.6. These results show that PQs provide useful video-specific guidance, particularly for event-level description.

Effect of tool-aware on-policy training. PQ-OPD benefits captioning even when visual observations are fixed. Without tools, training improves CaReBench F1 from 37.6 to 38.0 and DREAM-1K F1 from 30.3 to 31.9 (#1 $\rightarrow$ #5). In this setting, PQ-conditioned supervision can improve how the student interprets and describes the given frames, but cannot alter which visual evidence is observed.

With `seek`/`zoom` tools, the same supervision also covers the student's observation decisions along its interaction trajectory. PQ-OPD improves the untrained agent from 39.8 to 40.6 on CaReBench and from 31.3 to 33.3 on DREAM-1K (#3 $\rightarrow$ #8). This larger improvement is consistent with PQ guidance being particularly useful in the agentic setting, where training can shape both evidence acquisition and caption generation. Randomly redistributing individual PQs across training videos (Shuffled-PQ, #7) slightly improves the untrained agent, reaching 40.1 CaReBench F1 and 31.5 DREAM-1K F1. We hypothesize that PQs shift the teacher's output distribution to favor more exploration, consistent with the context-induced teacher behavior observed in OP^{2}SD (Ichihara et al., 2026).

We further compare on-policy distillation with SFT on PQ-conditioned teacher traces. For PQ trace SFT (#6), we sample four teacher rollouts per training video and fine-tune the student on the resulting interaction trajectories without PQ input. Under the same tool interface, PQ-OPD (#8) improves CaReBench F1 from 39.7 to 40.6 and DREAM-1K F1 from 32.6 to 33.3. Unlike SFT on fixed teacher trajectories, on-policy distillation provides supervision along the observation histories generated by the current student, making it better suited to learning interactive perception policies.

PQ guidance across model scales. Table 4 shows consistent gains from PQ-conditioned OPD across teacher–student scales. With a 27B teacher and 9B student, PQ-OPD improves CaReBench F1 from 40.1 to 41.2 and DREAM-1K F1 from 31.2 to 33.4. With a 9B teacher and 4B student, the corresponding scores increase from 39.4 to 40.2 and from 30.8 to 32.6. The gains at both scales show that PQ-conditioned supervision is not specific to a single teacher–student setting.

4.4 Analysis

Performance across input-frame budgets. A captioning agent must decide both where to inspect the video and how to use a limited visual input. We study this behavior by varying the input-frame budget from 2 to 64 frames for the 9B student trained with a 9B teacher. The budget specifies the maximum number of frames that may be returned across all tool calls. We compare uniformly sampled direct

Table 4: PQ guidance across teacher–student configurations. Both methods use seek/zoom tools; only PQ-OPD conditions the teacher on PQs, and all students are evaluated without PQs. Arrows denote teacher → student sizes. Scores are percentages; CaReBench F1 averages Action and Object F1, and DREAM-1K F1 is event F1. Bold denotes the better score within each teacher–student pair.

Teacher → Student	Training	CaReBench F1	DREAM-1K F1
27B → 9B	Vanilla OPD	40.1	31.2
27B → 9B	PQ-OPD	**41.2**	**33.4**
9B → 4B	Vanilla OPD	39.4	30.8
9B → 4B	PQ-OPD	**40.2**	**32.6**

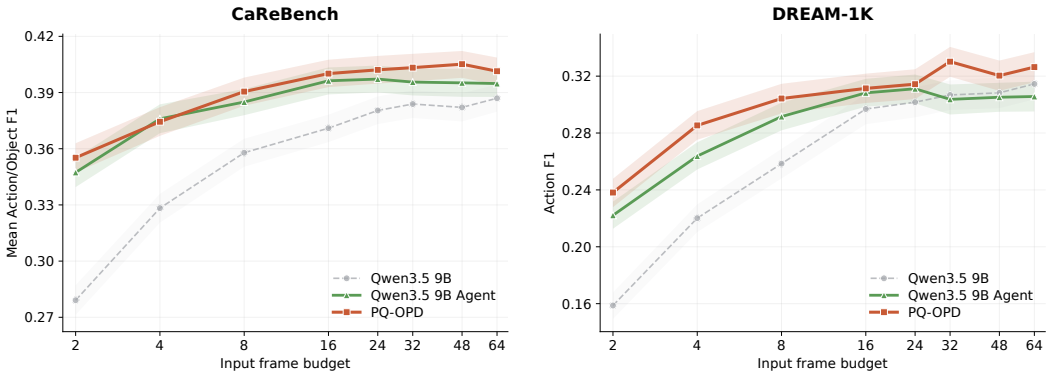


Figure 2: **Performance across input-frame budgets.** F1 under varying frame budgets, with a fixed evaluation cohort along each curve. CaReBench reports the arithmetic mean of Action F1 and Object F1; DREAM-1K reports event F1. Shaded regions show pointwise 95% bootstrap intervals.

video input, the untrained Qwen3.5 9B Agent, and PQ-OPD, with the two agents sharing the same tools, prompts, and frame limits.

PQ-OPD improves DREAM-1K F1 and CaReBench F1 across different frame budgets (Figure 2). The advantage is also clear under tight frame constraints, indicating that the learned agent uses its available visual input more effectively.

The comparison with uniformly sampled direct input further highlights the benefit of interactive observation. At small frame budgets, agentic models achieve higher scores on both benchmarks despite operating under the same input-frame limit. Across the full budget range, PQ-OPD consistently improves over the untrained agent, showing that PQ-OPD improves captioning across different levels of visual input rather than relying on a particular frame budget. Performance does not increase monotonically with the budget, indicating that a larger visual budget alone does not guarantee better performance and that the quality of the selected evidence also matters. Qualitative examples of caption details and observation patterns are provided in Section C.

Temporal exploration and caption length. We compare the base 9B agent and 9B → 9B PQ-OPD on 1,000 paired videos per benchmark. Figure 3 shows that the trained agent reaches PQ-relevant temporal intervals earlier, with Action/Object interval hit rates improving by 10.4–13.7 percentage points across the two benchmarks. Its `seek` windows are also better aligned with the PQ time intervals, increasing the mean best temporal intersection over union (tIoU) by 0.052–0.070. The final caption lengths remain similar, increasing by only 9.0% on CaReBench and 5.1% on DREAM-1K. These results suggest that the performance gains are empirically associated with a change in exploration policy, rather than simply with producing substantially different or longer final captions.

Effect of answer leakage. We vary the fraction of PQ answers provided to the 9B teacher during distillation and evaluate the resulting 9B student without questions or answers at inference. Figure 4 shows that increasing answer leakage is accompanied by lower captioning scores, fewer tool calls, and more zero-observation answers. From 0% to 100% leakage, CaReBench F1 falls from 40.6 to 35.8

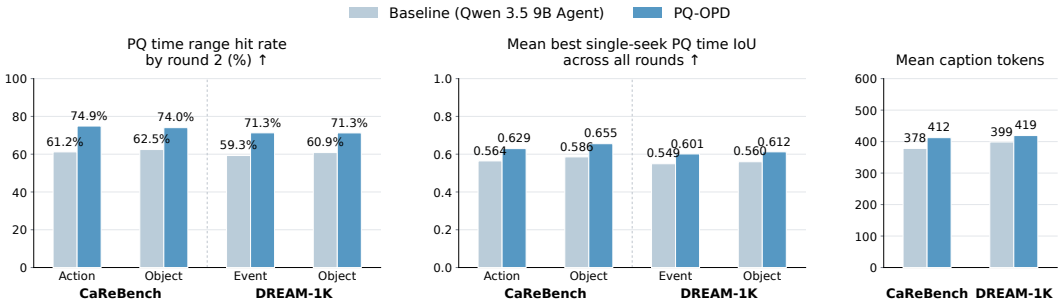


Figure 3: **Temporal exploration and caption length.** Left: PQ interval hit rate by round two (single-`seek` tIoU ≥ 0.5). Middle: mean best single-`seek` tIoU across all rounds. Right: final-caption tokens. Temporal scores are averaged within each video and then across videos.

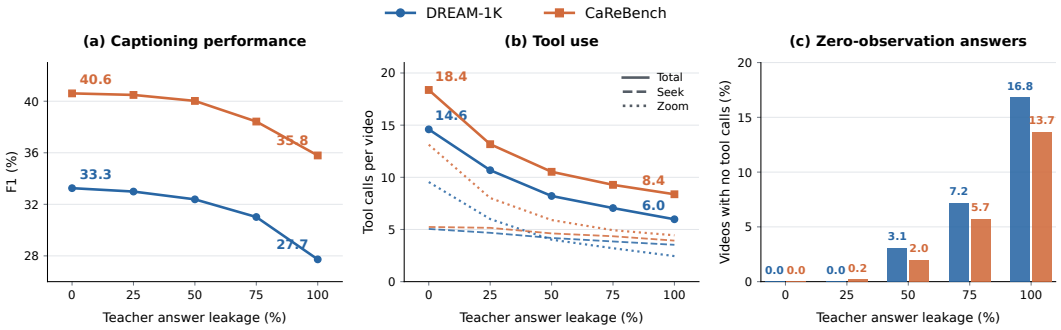


Figure 4: **Effect of answer leakage on PQ-OPD.** The rate of PQ answers leaked to the teacher during training varies from 0% to 100%; 0% is standard PQ-OPD. Both teacher and student are 9B models. (a) Captioning performance. (b) Mean tool calls per video. (c) Zero-observation answers: the student issues no tool calls and receives no visual input before producing a caption.

and DREAM-1K event F1 from 33.3 to 27.7. Average tool calls per video decrease from 18.4 to 8.4 on CaReBench and from 14.6 to 6.0 on DREAM-1K, with a particularly pronounced reduction in zoom calls. Meanwhile, the fraction of videos answered without any tool call rises from zero to 13.7% and 16.8%, respectively. Since evaluation starts without visual input, these answers are generated without observing the video. The growing zero-observation fraction at higher leakage suggests a failure to acquire visual evidence. Similar performance degradation from answer-conditioned supervision is also observed in ViCuR (Tian et al., 2026), which links answer-side privilege to shortcut learning. These results support withholding PQ answers, which guides the teacher toward relevant observations without supplying their outcomes.

5 Conclusion

We introduced PQ-OPD, an on-policy distillation framework that uses training-only Perception Questions to guide a video captioning agent's acquisition and description of visual evidence. A PQ-conditioned teacher supervises student-generated interaction trajectories, while the student receives no PQs at inference. Across five benchmarks, PQ-OPD improves caption quality and downstream accuracy across model scales and frame budgets. Ablations show benefits for both fixed-frame captioners and tool-enabled agents, with on-policy training outperforming fixed teacher traces SFT. Temporal-coverage analysis further shows earlier, better-aligned exploration, while answer leakage weakens visual evidence acquisition. These results suggest that answer-free observation targets can be internalized as a self-driven exploration policy.

Reproducibility Statement

Training data construction, evaluation protocols, and evaluation metrics are described in Section 4. Complete prompt templates are collected in Section B, and all training hyperparameters are listed in Section A. We promise to publicly release the PQ resource, generation code, and model weights.

AI Use Statement

We used large language and vision-language models (Gemini) to generate Perception Questions for training data construction (Section 4.1). We also used language models as automated evaluators in the evaluation process described in the paper, and to polish writing. We take responsibility for all claims, numbers, and content in this paper.

References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *ICLR*. OpenReview.net, 2024.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL Technical Report. *CoRR*, abs/2511.21631, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Ming-Hsuan Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL Technical Report. *CoRR*, abs/2502.13923, 2025b.

Wenhao Chai, Enxin Song, Yilun Du, Chenlin Meng, Vashisht Madhavan, Omer Bar-Tal, Jenq-Neng Hwang, Saining Xie, and Christopher D. Manning. AuroraCap: Efficient, performant video detailed captioning and a new benchmark. In *ICLR*, 2025.

Xinlong Chen, Yuanxing Zhang, Chongling Rao, Yushuo Guan, Jiaheng Liu, Fuzheng Zhang, Chengru Song, Qiang Liu, Di Zhang, and Tieniu Tan. VidCapBench: A comprehensive benchmark of video captioning for controllable text-to-video generation. In *ACL (Findings)*, pp. 8543–8563, 2025.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, Ji Ma, Jiaqi Wang, Xiaoyi Dong, Hang Yan, Hewei Guo, Conghui He, Botian Shi, Zhenjiang Jin, Chao Xu, Bin Wang, Xingjian Wei, Wei Li, Wenjian Zhang, Bo Zhang, Pinlong Cai, Licheng Wen, Xiangchao Yan, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. How far are we to GPT-4V? closing the gap to commercial multimodal models with open-source suites. *Sci. China Inf. Sci.*, 67(12): 220101, 2024. doi: 10.1007/s11432-024-4231-5.

Yang Ding, Xin Lai, Yizhen Zhang, Wei Li, Ruihang Chu, and Yujiu Yang. Videozoomer: Reinforcement-learned temporal focusing for long video reasoning. In *ICLR*, 2026.

Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, Congcong Wang, Dehao Zhang, Dikang Du, Dongliang Wang, Enming Yuan, Enzhe Lu, Fang Li, Flood Sung, Guangda Wei, Guokun Lai, Han Zhu, Hao Ding, Hao Hu, Hao Yang, Hao Zhang, Haoning Wu, Haotian Yao, Haoyu Lu, Heng Wang, Hongcheng Gao, Huabin Zheng, Jiaming Li, Jianlin Su, Jianzhou Wang, Jiaqi Deng, Jiezhong Qiu, Jin Xie, Jinhong

Wang, Jingyuan Liu, Junjie Yan, Kun Ouyang, Liang Chen, Lin Sui, Longhui Yu, Mengfan Dong, Mengnan Dong, Nuo Xu, Pengyu Cheng, Qizheng Gu, Runjie Zhou, Shaowei Liu, Sihan Cao, Tao Yu, Tianhui Song, Tongtong Bai, Wei Song, Weiran He, Weixiao Huang, Weixin Xu, Xiaokun Yuan, Xingcheng Yao, Xingzhe Wu, Xinhao Li, Xinxing Zu, Xinyu Zhou, Xinyuan Wang, Y. Charles, Yan Zhong, Yang Li, Yangyang Hu, Yanru Chen, Yejie Wang, Yibo Liu, Yibo Miao, Yidao Qin, Yimin Chen, Yiping Bao, Yiqin Wang, Yongsheng Kang, Yuanxin Liu, Yuhao Dong, Yulun Du, Yuxin Wu, Yuzhi Wang, Yuzi Yan, Zaida Zhou, Zhaowei Li, Zhejun Jiang, Zheng Zhang, Zhilin Yang, Zhiqi Huang, Zihao Huang, Zijia Zhao, Ziwei Chen, and Zongyu Lin. Kimi-vl technical report. *CoRR*, abs/2504.07491, 2025.

Sunqi Fan, Jiashuo Cui, Meng-Hao Guo, and Shuojin Yang. Tool-augmented spatiotemporal reasoning for streamlining video question answering task. In *NeurIPS*, 2025.

Yuying Ge, Yixiao Ge, Chen Li, Teng Wang, Junfu Pu, Yizhuo Li, Lu Qiu, Jin Ma, Lisheng Duan, Xinyu Zuo, Jinwen Luo, Weibo Gu, Zexuan Li, Xiaojing Zhang, Yangyu Tao, Han Hu, Di Wang, and Ying Shan. ARC-Hunyuan-Video-7B: structured video comprehension of real-world shorts. *CoRR*, abs/2507.20939, 2025. URL https://arxiv.org/abs/2507.20939.

Google. Gemini 3 Flash: frontier intelligence built for speed. Google Blog, 2025. URL https://blog.google/products/gemini/gemini-3-flash/.

Google. Gemini 3.1 pro: A smarter model for your most complex tasks, February 2026. URL https://blog.google/innovation-and-ai/models-and-research/gemini-models/gemini-3-1-pro/. Accessed: 2026-05-07.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *ICLR*. OpenReview.net, 2024.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

Wenyi Hong, Yean Cheng, Zhuoyi Yang, Weihan Wang, Lefan Wang, Xiaotao Gu, Shiyu Huang, Yuxiao Dong, and Jie Tang. MotionBench: Benchmarking and improving fine-grained video motion understanding for vision language models. In *CVPR*, pp. 8450–8460, 2025.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. In *ICML*, 2026.

Yuki Ichihara, Naoto Iwase, Mohammad Atif Quamar, and Junpei Komiyama. Privileged solutions or context-induced teacher behavior? dissecting on-policy self-distillation. *CoRR*, abs/2608.09228, 2026.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, pp. 706–715. IEEE Computer Society, 2017.

Chenglin Li, Qianglong Chen, Feng Han, Yikun Wang, Xingxi Yin, Yan Gong, Ruilin Li, Yin Zhang, and Jiaqi Wang. Videothinker: Building agentic videollms with llm-guided tool reasoning. In *CVPR Findings*, pp. 8226–8236, 2026a.

Keliang Li, Yansong Li, Hongze Shen, Mengdi Liu, Hong Chang, and Shiguang Shan. Lenswalk: Agentic video understanding by planning how you see in videos. In *CVPR*, pp. 19518–19528, 2026b.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *CoRR*, abs/2604.13016, 2026c.

Hao Lin, Kunyang Lv, Xu Jiang, Jingqi Tian, Zhongjing Du, Jiayu Ding, Qiaoman Zhang, and Hongbo Jin. VISD: Enhancing video reasoning via structured self-distillation. *CoRR*, abs/2605.06094, 2026.

Chunxu Liu, Jiyuan Yang, Ruopeng Gao, Yuhan Zhu, Feng Zhu, Rui Zhao, and Limin Wang. Reasoning guided embeddings: Leveraging MLLM reasoning for improved multimodal retrieval. *CoRR*, abs/2511.16150, 2025.

Ruiqi Liu, Xiaolei Lv, Gengsheng Li, Ximo Zhu, Zhiheng Wang, Zhengbo Zhang, Junkai Chen, Zhiheng Li, Bo Li, Jun Gao, and Shu Wu. Visual-advantage on-policy distillation for vision-language models. *CoRR*, abs/2605.21924, 2026.

Martin Q. Ma, Yuxiao Qu, Aditya Agrawal, Willis Guo, Paul Pu Liang, Ruslan Salakhutdinov, and Louis-Philippe Morency. Act2See: Emergent active visual perception for video reasoning. In *CVPR*, pp. 5455–5464, 2026.

Desen Meng, Rui Huang, Jiaqi Li, Zhilin Dai, Xinhao Li, Yifan Xu, Jun Zhang, Zhenpeng Huang, Meng Zhang, Yi Liu, and Limin Wang. VideoCap-R1: Enhancing multimodal large language models for detailed video captioning via structured thinking. *Comput. Vis. Image Underst.*, 272: 104905, 2026. doi: 10.1016/j.cviu.2026.104905.

OpenAI. GPT-4o System Card. *CoRR*, abs/2410.21276, 2024.

OpenAI. Introducing GPT-4.1 in the API. `https://openai.com/index/gpt-4-1/`, April 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Yudi Shi, Shangzhe Di, Qirui Chen, Qinian Wang, Jiayin Cai, Xiaolong Jiang, Yao Hu, and Weidi Xie. Weaver: End-to-end agentic system training for video interleaved reasoning. In *CVPR Findings*, pp. 9248–9258, 2026.

Kanghui Tian, Siyuan Liu, Ziang Yan, Sheng Xia, Shuai Dong, and Yi Wang. ViCuR: Visual cues as recoverable privilege for multimodal on-policy distillation. *CoRR*, abs/2606.05718, 2026.

Jiawei Wang, Liping Yuan, Yuchen Zhang, and Haomiao Sun. Tarsier: Recipes for training and evaluating large video description models. *CoRR*, abs/2407.00634, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *CoRR*, abs/2409.12191, 2024b.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. In *ECCV*, pp. 58–76, 2024c.

Ziyang Wang, Honglu Zhou, Shijie Wang, Junnan Li, Caiming Xiong, Silvio Savarese, Mohit Bansal, Michael S. Ryoo, and Juan Carlos Niebles. Active video perception: Iterative evidence seeking for agentic long video understanding. In *CVPR Findings*, pp. 9088–9099, 2026.

Keming Wu, Baoyi Wang, Kaichen Zhang, Xiang An, Zuhao Yang, Sudong Wang, Haowei Zhu, Tingxuan Huang, Hongcheng Gao, and Bin Wang. StreamOPD: A post-training recipe with spatio-temporal cue gating for streaming video understanding. *CoRR*, abs/2608.16320, 2026.

Long Xing, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Jianze Liang, Qidong Huang, Jiaqi Wang, Feng Wu, and Dahua Lin. CapRL: Stimulating dense image caption capabilities via reinforcement learning. In *ICLR*, 2026a.

Zhenghao Xing, Ruiyang Xu, Yuxuan Wang, Jinzheng He, Ziyang Ma, Qize Yang, Yunfei Chu, Jin Xu, Junyang Lin, Chi-Wing Fu, and Pheng-Ann Heng. Native active perception as reasoning for omni-modal understanding. In *ICML*, 2026b.

Yifan Xu, Xinhao Li, Yichun Yang, Desen Meng, Rui Huang, and Limin Wang. CaReBench: A fine-grained benchmark for video captioning and retrieval. In *ICLR*, 2026a.

Yifan Xu, Zihao Wang, Zhixiao Wang, Jiaming Zhang, Yichun Yang, Desen Meng, Yuanxing Zhang, Pengfei Wan, and Limin Wang. PercepCap: Video captioner with structured spatio-temporal perception. *CoRR*, abs/2607.20389, 2026b.

Penghui Yang, Long Xing, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Yibin Wang, Yujie Zhou, Jiazi Bu, Jianze Liang, Qidong Huang, Jiaqi Wang, Feng Wu, and Dahua Lin. Caprl++: Unified reinforcement learning with verifiable rewards for dense image and video captioning. *CoRR*, abs/2606.09393, 2026a.

Wenkai Yang, Weijie Liu, Ruobing Xie, Kai Yang, Saiyong Yang, and Yankai Lin. Learning beyond teacher: Generalized on-policy distillation with reward extrapolation. *CoRR*, abs/2602.12125, 2026b.

Linli Yao, Yuancheng Wei, Yaojie Zhang, Lei Li, Xinlong Chen, Feifan Song, Ziyue Wang, Kun Ouyang, Yuanxin Liu, Lingpeng Kong, Qi Liu, Pengfei Wan, Kun Gai, Yuanxing Zhang, and Xu Sun. Timechat-captioner: Scripting multi-scene videos with time-aware and structural audio-visual captions. In *ICML*, 2026.

Xinlei Yu, Gen Li, Qingyi Si, Guibin Zhang, Yuqi Xu, Congcong Wang, Shuai Dong, Kaiwen Tuo, Xiangyu Zeng, Kaituo Feng, Qunzhong Wang, Yang Shi, Xiaobin Hu, Xiangyu Yue, Jiaqi Wang, and Shuicheng Yan. DOPD: Dual on-policy distillation. *CoRR*, abs/2606.30626, 2026.

Liping Yuan, Jiawei Wang, Haomiao Sun, Yuchen Zhang, and Yuan Lin. Tarsier2: Advancing large vision-language models from detailed video description to comprehensive video understanding. *CoRR*, abs/2501.07888, 2025.

Qianhao Yuan, Jie Lou, Xing Yu, Hongyu Lin, Le Sun, Xianpei Han, and Yaojie Lu. Vision-OPD: Learning to see fine details for multimodal LLMs via on-policy self-distillation. *CoRR*, abs/2605.18740, 2026.

Zihao Yue, Zhenru Lin, Yifan Song, Weikun Wang, Shuhuai Ren, Shuhao Gu, Shicheng Li, Peidian Li, Liang Zhao, Lei Li, Kainan Bao, Hao Tian, Hailin Zhang, Xiao-Gang Wang, Dawei Zhu, Cici, Chenhong He, Bowen Ye, Bowen Shen, Zihan Zhang, Zihan Jiang, Zhixian Zheng, Zhichao Song, Zhenbo Luo, Yue Yu, Yudong Wang, Yuanyuan Tian, Yu Tu, Yihan Yan, Yi Huang, Xu Wang, Xinzhe Xu, Xingchen Song, Xing Zhang, Xing Yong, Xin Zhang, Xiangwei Deng, Wenyu Yang, Wenhan Ma, Weiwei Lv, Weiji Zhuang, Wei Liu, Sirui Deng, Shuo Liu, Shimao Chen, Shihua Yu, Shaohui Liu, Shande Wang, Rui Ma, Qiantong Wang, Peng Wang, Nuo Chen, Menghang Zhu, Kangyang Zhou, Kang Zhou, Kai Fang, Jun Shi, Jinhao Dong, Jiebao Xiao, Jiaming Xu, Huaqiu Liu, Hongshen Xu, Heng Qu, Haochen Zhao, Hanglong Lv, Guoan Wang, Duo Zhang, Dong Zhang, Di Zhang, Chong Ma, Chang Liu, Can Cai, and Bingquan Xia. Mimo-vl technical report. *CoRR*, abs/2506.03569, 2025.

Haoji Zhang, Xin Gu, Jiawen Li, Chixiang Ma, Sule Bai, Chubin Zhang, Bowen Zhang, Zhichao Zhou, Dongliang He, and Yansong Tang. Thinking with videos: Multimodal tool-augmented reinforcement learning for long video reasoning. In *CVPR*, pp. 32903–32914, 2026.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *CoRR*, abs/2601.18734, 2026a.

Yiming Zhao, Yu Zeng, Wenxuan Huang, Zhen Fang, Qing Miao, Qisheng Su, Jiawei Zhao, Jiayin Cai, Lin Chen, Zehui Chen, Yukun Qi, Yao Hu, Xiaolong Jiang, and Feng Zhao. VideoSeeker: Incentivizing instance-level video understanding via native agentic tool invocation. *CoRR*, abs/2605.16079, 2026b.

Chunlin Zhong, Qiuxia Hou, Zhangjun Zhou, Yanhao Zhang, Shuang Hao, Haonan Lu, He Tang, and Xiang Bai. Owlcap: Harmonizing motion-detail for video captioning via HMD-270K and caption set equivalence reward. In *AAAI*, pp. 13503–13511. AAAI Press, 2026.

Luowei Zhou, Chenliang Xu, and Jason J. Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, pp. 7590–7598. AAAI Press, 2018.

Yuhang Zhou, Kai Zheng, Haoling Li, Dengyun Peng, Can Xu, and Jingjing Chen. TurnOPD: Making on-policy distillation turn-aware for efficient long-horizon agent training. *CoRR*, abs/2607.05804, 2026.

A TRAINING DETAILS

Table 5 lists the training configuration for the Qwen3.5-9B student.

Table 5: Training hyperparameters and configuration.

Parameter	Value
Data and rollout	
Training videos	5,979 (Section 4.1)
Perception Questions	43,477
Maximum tool-image pixels	460,800
Unique prompts per refresh	64
Rollouts per prompt	4 (256 trajectories per refresh)
Optimization	
Optimizer	AdamW
Adam coefficients	$\beta_1 = 0.9$, $\beta_2 = 0.95$
Optimizer steps	372
Peak learning rate	2×10^{-6}
Learning-rate schedule	Cosine
Warmup ratio	0.05
Weight decay	0.1
Per-device batch size	1
Gradient accumulation steps	8
JSD β	0.5
Temperature	1.0
Memory and trainable components	
Memory optimization	DeepSpeed ZeRO-3
Vision encoder and aligner	Frozen

B PROMPT TEMPLATES

This section collects the four prompt templates. Colored placeholders are replaced by the video duration (in seconds, to three decimal places) or its numbered Perception Questions. Typography and line wrapping are adapted for readability; the prompt wording is preserved.

B.1 PERCEPTION QUESTION GENERATION

The prompt template covers object, action, and temporal questions. The generator receives the video and its duration; generated answers and evidence are retained for quality control. Only questions reach the teacher.

PQ generation **Instructions and coverage**

Watch the entire video and create a compact set of timestamped questions that will help another video model inspect the video and produce a complete, accurate description.

The video duration is `{duration:.3f}` seconds.

Generate approximately 6 to 10 questions in total. Choose the number according to the visual and temporal complexity of the video. Do not add redundant questions merely to reach a target count.

Use three equally important question types:

1. `object`

Ask about concrete visible objects that are important for understanding the video, especially objects that are used, manipulated, exchanged, carried, or otherwise involved in the main activity.

2. `action`

Ask about meaningful visible actions, movements, or interactions occurring within the stated interval.

3. `temporal`

Ask about temporal order, action sequences, activity progression or completion, meaningful state changes, or repeated behavior across the stated interval.

Include all three question types, but do not follow a fixed numerical ratio. Allocate questions according to what is actually important in the video. Object, action, and temporal questions should be treated as equally valid ways to guide inspection rather than as separate generation stages.

For temporal questions, prioritize:

- complete action chains;
- chronological progression or completion of an activity;
- meaningful changes in the state, position, or arrangement of a subject or object;
- repeated or cyclic behavior.

At least one temporal question should cover the main progression of the video and span at least 60% of its duration.

Avoid creating a static inventory of every visible object. Do not focus on clothing details, colors, background decoration, text, watermarks, lighting, camera movement, or other incidental content unless it is important to the main activity.

Every question must:

- contain exactly one explicit time range written as `between X and Y seconds` or `from X to Y seconds`;
- be answerable from visible evidence within that interval;
- leave the requested object identity, action, event sequence, or resulting state unknown;
- use only the timestamp and, when necessary, a neutral visible anchor such as `the main subject`, `the seated person`, or `the main visible object`;
- contribute information useful for producing a detailed video description.

Collectively, the questions should:

- cover the important content near the beginning, middle, and end of the video;
- cover central subjects, objects, actions, interactions, and temporal progression;
- avoid duplication and unnecessary detail.

PQ generation — **Constraints and output format (continued)**

Leakage restrictions:

- Do not reveal the answer in the question.
- Do not state an observed event as the premise for another question.
- Do not use premises such as `after the person picks up the cup`, `before the vehicle stops`, `following the action`, or `once the subject enters`.
- Do not reveal concrete actions, object identities, or final states that the question is intended to ask about.
- Do not use proper names or non-visual identities.
- Do not ask why something happens or infer intent, causality, emotion, or hidden state.
- Do not generate yes/no questions.
- Do not duplicate or paraphrase another generated question.

Preferred forms:

- `Which object is the person using between 2.0 and 4.0 seconds?`
- `What important object is involved from 4.0 to 6.0 seconds?`
- `What happens between 2.0 and 4.0 seconds?`
- `What does the main subject do from 4.0 to 6.0 seconds?`
- `What sequence of actions does the main subject perform from 0.0 to 12.0 seconds?`
- `What happens first, next, and finally between 2.0 and 10.0 seconds?`
- `How does the main subject's activity progress from 1.0 to 8.0 seconds?`
- `How does the state of the main visible object change between 3.0 and 9.0 seconds?`
- `What action pattern repeats, and how does it develop, from 0.0 to 15.0 seconds?`

For quality control, provide a concise answer and a short visual evidence statement for every question. Answers, evidence, type metadata, and temporal subtype metadata will not be shown to the evaluated model.

For a temporal question, assign one of these subtypes: `action_chain`, `activity_progression`, `state_change`, or `repetition`. Object and action questions do not need a temporal subtype.

Return JSON only:

```
{
  "questions": [
    {
      "question": "...?",
      "answer": "...",
      "evidence": "...",
      "start_time": 0.0,
      "end_time": 2.0,
      "type": "object|action|temporal",
      "temporal_subtype": null
    },
    {
      "question": "...?",
      "answer": "...",
      "evidence": "...",
      "start_time": 0.0,
      "end_time": 10.0,
      "type": "temporal",
      "temporal_subtype":
     "action_chain|activity_progression|state_change|repetition"
    }
  ]
}
```

B.2 AGENT SYSTEM PROMPT

The student uses this system prompt for tool-interaction rollouts. The teacher scores the same history under this setting with a PQ-conditioned user query.

Agent — **System message**

You are a video-analysis agent. The source video is {duration:.3f} seconds long and is available through the provided tools.

You must use the tools to explore the video before answering:

seek(t_start, t_end) takes times in seconds and returns up to 8 representative frames from that interval.
zoom(timestamp, bbox) returns a crop from one frame; bbox is [x1, y1, x2, y2] in 0-1000 coordinates.

You may call multiple tools in the same turn. In the final answer, do not mention the tools, hidden-input setup, or these instructions.

B.3 STUDENT USER PROMPT

The student receives the generic instruction during training and inference.

Student — **User message**

Describe the video in detail.

B.4 PQ-CONDITIONED TEACHER USER PROMPT

The frozen teacher receives numbered PQs while sharing the student's actions and observations. Answers and reference captions are excluded.

Teacher — **User message · Training only**

Describe the video in detail. Include all visually supported information needed to answer the questions below. Do not mention or repeat the questions, and do not answer them one by one.

Questions:
1. {q_1}
2. {q_2}
...
K. {q_K}

Output only the video description.

C QUALITATIVE EXAMPLES

We compare selected caption excerpts and observation patterns from the non-agentic Qwen3.5 9B model, the untrained agent, and PQ-OPD on two DREAM-1K videos. Both agents operate without PQs at inference.

Costume interaction. Figure 5 compares captions and observation patterns on a DREAM-1K video. Both agentic models receive 35 visual observations: the base agent uses 32 initial seek frames and 3 subsequent single-frame zoom crops, whereas PQ-OPD uses 24 seek frames and 11 zoom crops. The base agent captures a background child omitted by the fixed-frame caption. PQ-OPD describes the detached costume head more specifically than the base agent's "toy or prop" and further captures the fallen character getting onto its knees and beginning to stand. Despite identical returned-image counts, the two agents capture different details.

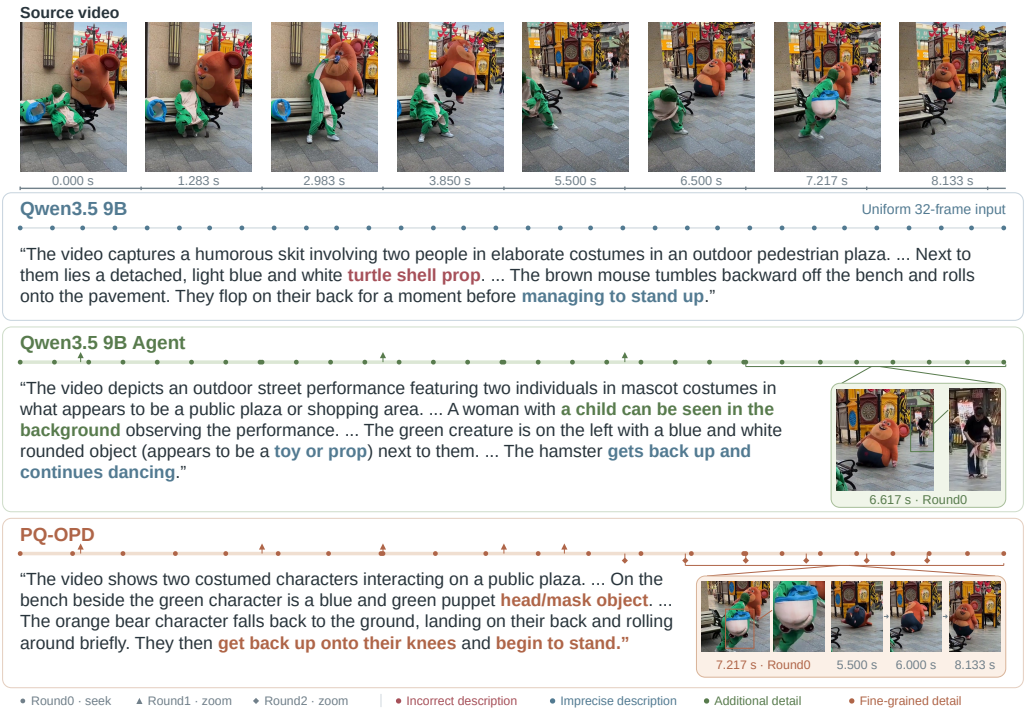


Figure 5: **Qualitative comparison on DREAM-1K.** Qwen3.5-9B receives 32 uniformly sampled frames, while Qwen3.5-9B Agent and PQ-OPD each obtain 35 visual observations through tool use. Circles denote the initial `seek` observations, triangles denote first-round single-frame inspections, and diamonds denote second-round inspections. The enlarged panels show cropped regions for easier visual comparison. The examples highlight differences in object identification, background details, and the fallen character's recovery motion.

Table-tennis match. Figure 6 compares identity recognition and action details in a table-tennis match. The base agent correctly reads "M.YOSHIMURA" on the player's jersey, whereas the non-agentic caption misspells the name. PQ-OPD preserves the correct identity and further describes the player pumping his right fist before the video cuts to the team bench.

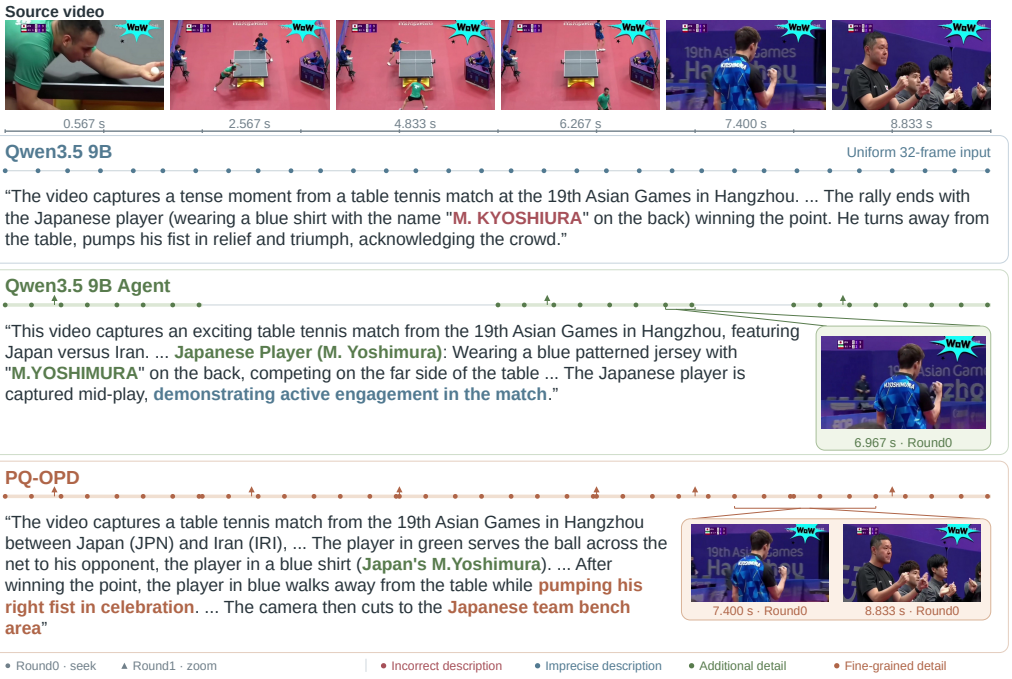


Figure 6: **Player identity and action details on DREAM-1K.** Qwen3.5-9B receives 32 uniformly sampled frames, while the two agentic models acquire observations through `seek` and `zoom`. Circles denote the initial `seek` observations, and triangles denote first-round `zoom` inspections. The comparison highlights jersey-name recognition and the description of the player's fist-pump celebration and the subsequent cut to the team bench. Enlarged panels show the relevant visual evidence.

D LIMITATIONS

PQ-OPD relies on an external video model to generate Perception Questions offline, introducing additional data-construction cost and potential generator bias. Our evaluation also relies substantially on model-based judges, including replacement judges for benchmarks whose original evaluators are no longer publicly accessible; absolute scores may therefore vary with the evaluator and prompt. At inference, the trained agent requires no PQs, but iterative `seek`/`zoom` calls are more expensive than a single fixed-frame forward pass. Finally, our experiments focus on a specific `seek`/`zoom` interface and Qwen3.5 backbones. Whether the learned observation policy transfers to other tool interfaces, and other model families remains an open question.