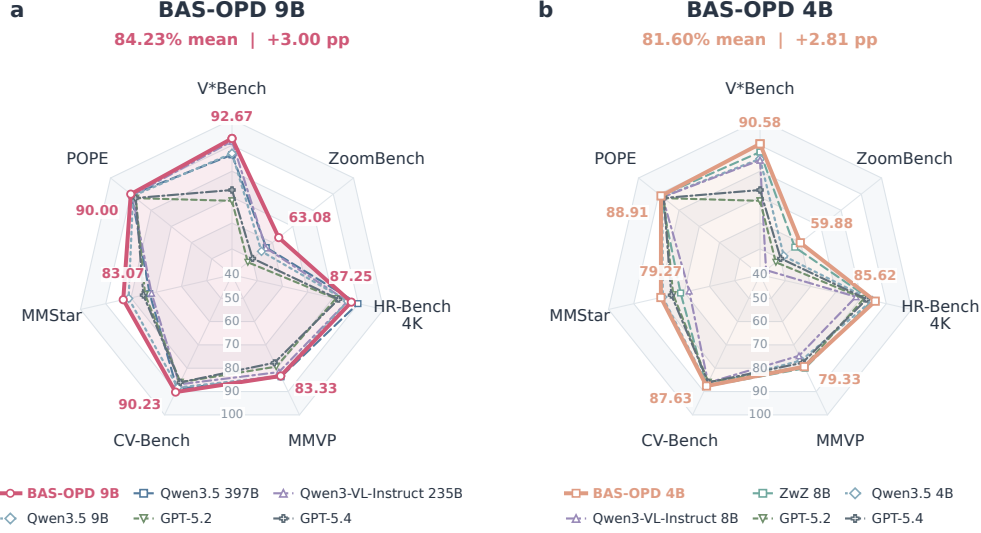


BAS-OPD: BUDGET-AWARE SELECTIVE ON-POLICY SELF-DISTILLATION FOR FINE-GRAINED MULTIMODAL PERCEPTION

Zihan Chen^{1,2}, Hengguang Zhou³, Yuan Kang^{1,2}, Yiming Zhang^{1,2},
Wenhui Fang^{1,2}, Zenghui Ding¹, Yining Sun^{1,2}, Cho-Jui Hsieh³

¹HFIPS, Chinese Academy of Sciences ²University of Science and Technology of China

³University of California, Los Angeles



Accuracy (%) across seven visual understanding benchmarks: V*Bench, ZoomBench, HR-Bench 4K, MMVP, CV-Bench, MMStar and POPE.

ABSTRACT

Multimodal large language models (MLLMs) often struggle with fine-grained visual perception when processing complete images, as critical evidence may only appear in local regions. On-policy self-distillation (OPD) enables transferring privileged visual knowledge from informative views to full-image policies, but querying the teacher for every rollout introduces substantial supervision costs. In this work, we propose BAS-OPD, a budget-aware selective OPD framework that allocates teacher supervision under limited query budgets. Instead of querying all rollouts, BAS-OPD selects informative samples while maintaining full-batch student generation. We explore random, uncertainty-based, and learned utility-based selection strategies, where the learned selector estimates query value from detached rollout statistics and online utility signals derived from student-teacher agreement and teacher confidence without additional student forward passes. BAS-OPD only changes training-time supervision allocation and preserves single-pass full-image inference. Experiments on fine-grained multimodal perception benchmarks demonstrate that BAS-OPD achieves strong performance while substantially reducing teacher supervision costs, highlighting the effectiveness of selective OPD under constrained budgets.

1 INTRODUCTION

Multimodal large language models (MLLMs) have become general-purpose interfaces for combining visual perception with language reasoning. Their performance, however, remains brittle when

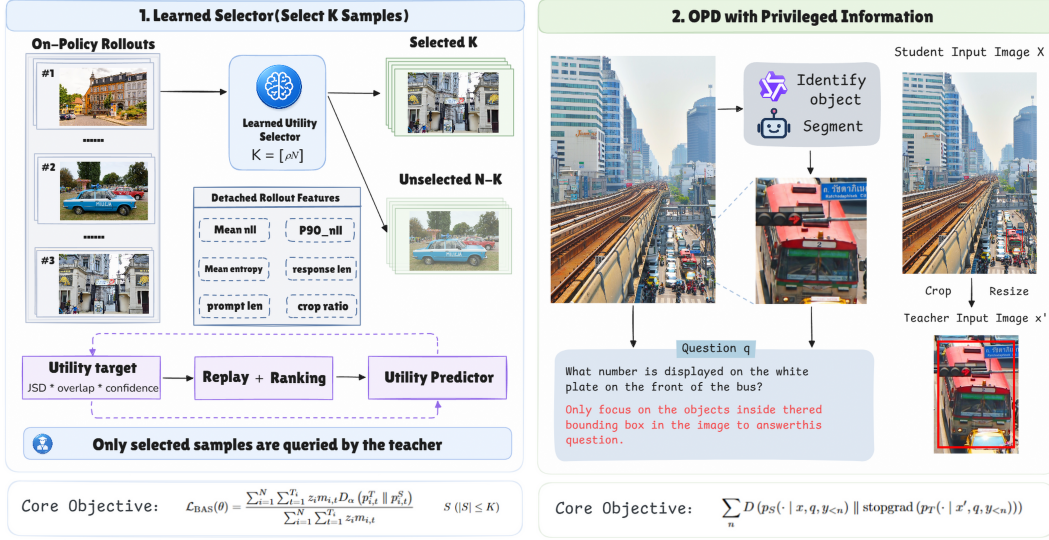


Figure 1: Overview of BAS-OPD. The full-image policy generates all rollouts; a budget-aware selector sends only K samples to the crop teacher for distillation and utility learning. Selection is removed at inference.

an answer depends on a small object, a local attribute, or a spatial relation embedded in a high-resolution scene. V*Bench and HR-Bench expose this limitation by requiring models to locate and interpret fine visual evidence in large images (Wu & Xie, 2024; Wang et al., 2025). The underlying difficulty is not only local recognition. Resizing an image or encoding the scene at nearly uniform spatial fidelity can weaken the few visual signals that determine the answer. Reliable fine-grained perception must therefore preserve local evidence while directing limited computation toward the regions that matter.

Existing methods improve access to fine visual evidence along two main directions. Model-side approaches retain more pixels through high-resolution encoders or dynamic tiling; for example, InternVL 1.5 processes an image as up to 40 tiles to support 4K inputs (Chen et al., 2024b). Inference-time approaches instead decide where to look. V* performs language-guided visual search, while DC² recursively partitions an image, describes its patches, and retrieves relevant regional information (Wu & Xie, 2024; Wang et al., 2025). These methods reduce information loss, but they trade against larger visual-token budgets, repeated patch processing, or additional inference calls. More importantly, they recover detail as an inference-time operation rather than teaching a single-pass full-image policy to internalize the advantage of focused regional evidence.

On-policy distillation transfers teacher behavior along trajectories sampled from the student. In autoregressive language models, it reduces the mismatch between teacher-forced training and student-generated inference while supporting sequence-level knowledge transfer (Agarwal et al., 2024; Ko et al., 2024). A recent multimodal formulation applies this principle to transfer privileged regional evidence into a single-pass full-image policy (Yuan et al., 2026). Existing methods mainly focus on learning objectives or supervision quality and typically query the teacher for every eligible rollout. This ties supervision cost to the full rollout batch, even when feedback is redundant or weakly informative. The unresolved challenge is to allocate teacher queries selectively under a strict budget while preserving the benefits of student-generated training.

We introduce BAS-OPD, which selects on-policy rollouts before crop-teacher scoring under a strict sample-level budget. The full-image policy generates responses for the entire batch, but only selected samples enter teacher scoring and distillation. We support three selection strategies: random, student-entropy, and learned utility selection. The learned selector uses detached rollout statistics, requiring no additional student forward pass. Online utility targets combine student-teacher divergence, top- k overlap, and teacher confidence to prioritize large, confident corrections from privileged regional evidence. Selection is used only during training, preserving single-pass full-image inference.

We evaluate BAS-OPD with Qwen3.5-4B and Qwen3.5-9B on seven benchmarks, measuring downstream accuracy together with teacher calls and scored tokens. With the 4B backbone, learned selection under a 25% teacher-call budget achieves an average accuracy of 81.60%, 1.84 points above full querying, while using only 15.78% of its teacher-scored tokens. Scaling BAS-OPD to 9B yields an average accuracy of 84.23%, 3.00 points above the strongest non-BAS open-source single-forward-pass model. Within this comparison group, BAS-OPD 9B ranks first on five benchmarks and second on the remaining two. A frozen-selector audit further shows that learned selection retrieves higher-utility rollouts than student-entropy selection. Our contributions are:

- **Budget-Aware OPD.** We formulate regional-to-global on-policy self-distillation with a strict, directly measurable teacher-query budget.
- **Online Utility Selection.** We unify random, student-entropy, and learned utility policies while ensuring that unselected rollouts never enter the crop-teacher forward pass.
- **Accuracy–Cost and Selection Diagnostics.** We evaluate accuracy and teacher cost across policies and budgets, with utility, ranking, and gradient analyses characterizing how learned selection concentrates supervision.

2 METHOD

2.1 BUDGET-AWARE SELECTIVE OPD

Figure 1 summarizes the training and inference paths. Algorithm 1 details the learned-selection training loop. BAS-OPD augments regional-to-global OPD with an explicit budget on crop-teacher supervision (Yuan et al., 2026). Consider a training batch $\mathcal{B} = \{(x_i, \tilde{x}_i, q_i)\}_{i=1}^N$, where x_i is the full image, $\tilde{x}_i$ is its privileged evidence-centered crop, and q_i is the language prompt. The full-image policy first samples an on-policy response $y_i = (y_{i,1}, \dots, y_{i,T_i})$ for every example in the batch. For response position t , we denote the student distribution conditioned on the full image by $p_{i,t}^S$ and the crop-teacher distribution over the same response prefix by $p_{i,t}^T$:

$$p_{i,t}^S = p_\theta(\cdot \mid x_i, q_i, y_{i,<t}), \quad (1)$$

$$p_{i,t}^T = p_{\bar{\theta}}(\cdot \mid \tilde{x}_i, q_i, y_{i,<t}). \quad (2)$$

Here, θ denotes the trainable full-image policy and $\bar{\theta}$ denotes the crop-conditioned teacher maintained by the underlying OPD procedure. Standard OPD minimizes the expected token-level divergence along responses sampled from the student:

$$\mathcal{L}_{\text{OPD}}(\theta) = \mathbb{E}_{\substack{(x_i, \tilde{x}_i, q_i) \sim \mathcal{D} \\ y_i \sim p_\theta(\cdot \mid x_i, q_i)}} \left[\frac{1}{T_i} \sum_{t=1}^{T_i} D_\alpha(p_{i,t}^T \parallel p_{i,t}^S) \right], \quad (3)$$

where $\mathcal{D}$ is the training distribution and D_α is the distillation divergence; our experiments use Jensen–Shannon divergence with $\alpha = 0.5$. The teacher scores the same response prefixes generated by the student. BAS-OPD does not change how responses are generated or how the teacher is regularized; it changes which rollouts are allowed to enter the crop-teacher forward pass.

Let $\mathcal{C} \subseteq \{1, \dots, N\}$ be the samples for which privileged crops are available. Given a query ratio $\rho \in (0, 1]$, we impose the sample budget

$$K = \lfloor \rho |\mathcal{C}| \rfloor, \quad \mathcal{S} \subseteq \mathcal{C}, \quad |\mathcal{S}| \leq K, \quad (4)$$

where $\mathcal{S}$ is chosen before any crop-teacher computation. The full-image policy therefore produces rollouts for the complete batch, while the crop teacher scores only the selected subset. This ordering is important: the budget reduces the actual teacher input batch rather than masking losses after an already completed teacher forward pass. For $i \notin \mathcal{S}$, no crop-teacher distribution or OPD target is constructed.

Selection aims to maximize the total utility of the queried subset under this budget. Because utility is observed only after teacher scoring, we predict it from statistics already produced by student rollouts (Algorithm 1, lines 3–9).

Algorithm 1 BAS-OPD with learned utility selection**Input:** Policy θ ; query ratio ρ ; exploration ratio ϵ ; replay threshold M ; EMA rate τ .**Output:** Trained full-image policy θ for inference without crops or a selector.

```

1: Initialize  $\bar{\theta} \leftarrow \theta$ , predictor  $f_\psi$ , feature normalizer, and FIFO buffer  $\mathcal{R} \leftarrow \emptyset$ .
2: for each training step  $s$  with rollout batch  $\mathcal{B}$  do
3:   Generate  $y_i \sim p_\theta(\cdot \mid x_i, q_i)$  for all  $i \in \mathcal{B}$ ; identify crop-eligible candidates  $\mathcal{C}$ .
4:   Form detached features  $\phi_i$  (Eq. 5) and update normalization on  $\mathcal{C}$ ; set  $K \leftarrow \lfloor \rho|\mathcal{C}| \rfloor$ .
5:   if  $f_\psi$  has not yet been updated then
6:      $\mathcal{S} \leftarrow \text{Uniform}(\mathcal{C}, K)$ , sampling without replacement.
7:   else
8:     Predict  $\hat{u}_i \leftarrow f_\psi(\text{Normalize}(\phi_i))$  for  $i \in \mathcal{C}$ ; set  $K_e \leftarrow \lceil \epsilon K \rceil$ .
9:     Form  $\mathcal{S}$  from the top  $K - K_e$  candidates by  $\hat{u}_i$  plus  $K_e$  candidates sampled uniformly without replacement from the remainder.
10:  end if
11:  Run the standard OPD student forward for  $p^S$ ; obtain detached crop-teacher  $p^T$  on the same tokens only for  $i \in \mathcal{S}$ .
12:  Compute detached utility labels  $u_i$  for  $i \in \mathcal{S}$  (Eq. 9).
13:  Update  $\theta$  using  $\mathcal{L}_{\text{BAS}}$  (Eq. 11); after a valid update, set  $\bar{\theta} \leftarrow (1 - \tau)\bar{\theta} + \tau\theta$ .
14:  Add valid detached  $(\phi_i, u_i, s)$  to bounded replay  $\mathcal{R}$ ; if new pairs were added and  $|\mathcal{R}| \geq M$ , update only  $\psi$  from replay using  $\mathcal{L}_{\text{sel}}$  (Eq. 10).
15: end for

```

2.2 LEARNED UTILITY SELECTION

For each rollout, Algorithm 1 (line 4) constructs a detached feature vector

$$\phi_i = [\bar{\ell}_i, Q_{0.9}(\ell_i), \bar{H}_i, T_i, L_i, a_i]^\top, \quad (5)$$

which summarizes rollout likelihood and entropy, response and prompt lengths, and crop area. These quantities require no additional MLLM forward pass. A lightweight predictor estimates query utility from normalized features; feature definitions, running normalization, and network architecture are given in Appendix A.2:

$$\hat{u}_i = f_\psi(\text{Normalize}(\phi_i)). \quad (6)$$

Algorithm 1 (lines 5–9) selects K candidates uniformly before the first predictor update. Thereafter, it reserves $K_e = \lceil \epsilon K \rceil$ queries for random exploration and assigns the remaining $K - K_e$ queries to the highest-scoring candidates. Exploration samples from the remaining candidates to avoid a self-confirming selection loop while preserving the strict budget.

The supervision for f_ψ is obtained online from selected rollouts (Algorithm 1, line 12). Let $P_{i,t}^S$ and $P_{i,t}^T$ denote student and teacher probabilities evaluated on the student’s top- k token indices and renormalized over this shared support. We measure their disagreement with the Jensen–Shannon divergence

$$d_{i,t} = \text{JSD}(P_{i,t}^S, P_{i,t}^T). \quad (7)$$

Divergence alone can be uninformative when the teacher is diffuse or favors different tokens. We define $o_{i,t}$ as the fraction of student top- k indices also in the teacher’s own top- k set, and compute the normalized teacher confidence

$$c_{i,t} = 1 - \frac{H(P_{i,t}^T)}{\log k}, \quad (8)$$

and define the detached per-sample target as

$$u_i = \frac{1}{\sum_t m_{i,t}} \sum_t m_{i,t} d_{i,t} o_{i,t} c_{i,t}, \quad (9)$$

where $m_{i,t}$ masks padding and invalid response positions. This target is high when the crop teacher makes a large, support-consistent, and confident correction to the full-image policy. Only queried samples yield (ϕ_i, u_i) pairs; unselected samples are never assigned synthetic utility labels.

Observed feature–utility pairs train the predictor through replay using weighted Huber regression and pairwise ranking:

$$\mathcal{L}_{\text{sel}} = \mathcal{L}_{\text{reg}} + \lambda_{\text{rank}} \mathcal{L}_{\text{rank}}, \quad (10)$$

Regression calibrates utility values, while ranking learns their ordering. Algorithm 1 (line 14) updates only the selector from detached features and targets; the replay procedure and loss definitions are given in Appendix A.3.

2.3 SELECTIVE DISTILLATION AND INFERENCE

Algorithm 1 (lines 11–13) scores only $\mathcal{S}$ with the crop teacher and compares its detached distributions with the standard OPD student distributions on the same response tokens. Using D_α from Eq. 3, we aggregate selected valid tokens with a masked mean. With $z_i = \mathbf{1}[i \in \mathcal{S}]$, the budgeted objective is

$$\mathcal{L}_{\text{BAS}}(\theta) = \frac{\sum_{i=1}^N \sum_{t=1}^{T_i} z_i m_{i,t} D_\alpha(p_{i,t}^T \| p_{i,t}^S)}{\max\left(1, \sum_{i=1}^N \sum_{t=1}^{T_i} z_i m_{i,t}\right)}. \quad (11)$$

At full budget, $\rho = 1$ and $\mathcal{S} = \mathcal{C}$, this objective recovers full-query OPD with the same token-mean reduction. Under a reduced budget, unselected rows are absent from the crop-teacher forward pass and are masked out of the distillation loss. Empty-selection and update handling are detailed in Appendix A.3. The MLLM parameters θ and selector parameters ψ are optimized separately by $\mathcal{L}_{\text{BAS}}$ and $\mathcal{L}_{\text{sel}}$, respectively. Baselines and supervision-cost definitions are given in Appendices A.1 and C. All selection components are training-only. At inference, BAS-OPD retains the original full-image policy and produces an answer in a single forward pass without privileged crops, a selector, or an external visual tool.

3 EXPERIMENTS

3.1 EXPERIMENTAL SETUP

MLLM and selector training. We evaluate BAS-OPD with Qwen3.5-4B and Qwen3.5-9B; policy ablations and cost measurements use the 4B model. Training uses 3,120 preprocessed full-image–crop pairs from the released training set (Yuan et al., 2026), JSD with $\alpha = 0.5$, top-100 distillation, a crop-teacher EMA rate of 0.05, and maximum prompt/response lengths of 8,192/1,024 tokens. The MLLM parameters θ use AdamW (lr 2×10^{-6} ; weight decay 0.01); the selector parameters ψ use separate Adam (lr 1×10^{-3}) in Learned-10/25/50. Each run uses 28 prompts per step and eight rollouts per prompt, trains for one epoch on four NVIDIA A100-SXM4 80 GB GPUs, and uses training seed 42. Appendix A.2 and Table 6 give the remaining selector settings and run lengths.

Evaluation benchmarks. We evaluate on seven benchmarks spanning high-resolution perception, general visual understanding, and object hallucination. V*Bench (Wu & Xie, 2024), ZoomBench (Wei et al., 2026), and HR-Bench 4K (Wang et al., 2025) assess fine-grained perception; MMVP (Tong et al., 2024b), CV-Bench (Tong et al., 2024a), and MMStar (Chen et al., 2024a) measure broader visual capabilities. POPE (Li et al., 2023) evaluates object hallucination across adversarial, popular, and random splits.

Evaluation protocol. Evaluation uses only the original full image. Privileged crops and all selection modules are disabled. We use deterministic decoding with temperature 0 and seed 42. Benchmark-specific parsers score multiple-choice and exact-answer outputs. GPT-5.5 at temperature 0 judges the remaining free-form responses. We report accuracy or the official aggregate score for each benchmark.

3.2 EXPERIMENTAL RESULTS

Comparison with state-of-the-art MLLMs. Table 1 compares BAS-OPD with agentic, closed-source, and open-source MLLMs. BAS-OPD 9B averaged 84.23%, exceeding Qwen3.5-9B, the strongest non-BAS open-source single-forward-pass model by average score, by 3.00 points. Among open-source single-forward-pass models, it ranked first on five benchmarks and second on two. It also outperformed Qwen3-VL-Instruct 235B on all seven benchmarks. BAS-OPD 4B averaged 81.60%, exceeding Qwen3.5-9B and Qwen3.5-397B by 0.37 and 0.46 points, respectively. Against closed-source models, BAS-OPD 9B led on four benchmarks, while Gemini remained stronger on the other three.

Table 1: Comparison with state-of-the-art MLLMs across seven benchmarks. We report accuracy (%) for each benchmark. Among the reported open-source single-forward-pass results, the best value is shown in bold and the second-best is underlined.

Model	Param. Size	V*Bench	ZoomBench	HR-Bench 4K	MMVP	CV-Bench	MMStar	POPE
<i>“Thinking-with-Images” Agentic Models</i>								
DeepEyes	7B	79.58	46.04	74.25	72.00	76.81	61.33	88.42
Thyme	7B	81.68	46.39	76.75	70.33	75.81	60.53	60.53
DeepEyesV2	7B	80.63	46.27	75.00	72.33	80.48	61.07	89.08
<i>Closed-Source Models (Single Forward Pass)</i>								
GPT-5.2	–	68.59	47.81	81.12	79.33	85.99	74.40	87.51
GPT-5.4	–	72.77	50.18	82.25	77.67	86.08	75.13	87.58
Gemini-3.1-Pro	–	88.48	60.83	89.62	91.00	89.98	83.67	89.72
Gemini-3.5-Flash	–	90.05	60.95	89.50	94.33	88.83	84.20	89.88
<i>Open-Source Models (Single Forward Pass)</i>								
Qwen3.5	4B	84.82	51.72	84.38	76.67	87.13	78.53	88.28
Qwen3.5	9B	86.91	54.56	85.75	<u>83.33</u>	88.29	<u>80.93</u>	88.88
Qwen3-VL-Instruct	8B	84.29	42.96	78.00	74.67	86.22	68.07	88.12
MiMo-VL-RL	7B	81.68	43.20	73.25	75.67	80.45	69.87	88.36
ZwZ	8B	87.43	57.16	84.50	80.00	87.77	71.47	88.97
MiniCPM-V-4.5	9B	69.63	42.84	68.75	78.00	79.88	67.53	87.51
Qwen3-VL-Instruct	235B	<u>91.62</u>	56.69	85.75	81.67	86.90	71.87	89.26
Qwen3.5	397B	86.39	56.92	89.88	83.67	<u>88.92</u>	72.60	<u>89.63</u>
BAS-OPD (4B)	4B	90.58	<u>59.88</u>	85.62	79.33	87.63	79.27	88.91
BAS-OPD (9B)	9B	92.67	63.08	<u>87.25</u>	<u>83.33</u>	90.23	83.07	90.00

Comparison with alternative training objectives. Table 2 compares BAS-OPD with vanilla inference, SFT, GRPO, DAPO, and OPSD on matched Qwen3.5 backbones. At 4B and 9B, BAS-OPD led on six benchmarks, tied the best MMVP score, and improved the OPSD average by 3.03 and 3.24 points, respectively, under a 25% teacher-query budget. Its largest gains over the strongest alternatives were on V*Bench and ZoomBench.

Table 2: Accuracy (%) of vanilla Qwen3.5 (Qwen Team, 2026), SFT (Ouyang et al., 2022), GRPO (Shao et al., 2024), DAPO (Yu et al., 2025), OPSD (Zhao et al., 2026), and BAS-OPD (learned selection, 25% teacher-query budget). Best results per backbone are bold.

Method	Fine-Grained Visual Tasks			Holdout Tasks			
	V*Bench	ZoomBench	HR-Bench 4K	MMVP	CV-Bench	MMStar	POPE
<i>Qwen3.5-4B</i>							
Vanilla	84.82	51.72	84.38	76.67	87.13	78.53	88.28
SFT on Self-Teacher	78.53	54.44	80.00	78.00	85.78	68.27	87.44
GRPO	83.25	55.62	82.50	79.33	87.16	70.67	86.39
DAPO	85.86	55.38	84.38	79.33	87.07	72.53	86.59
OPSD	83.77	54.32	82.00	79.00	87.19	74.87	88.88
BAS-OPD	90.58	59.88	85.62	79.33	87.63	79.27	88.91
<i>Qwen3.5-9B</i>							
Vanilla	86.91	54.56	85.75	83.33	88.29	80.93	88.88
SFT on Self-Teacher	81.68	58.82	83.13	81.33	88.05	73.53	87.50
GRPO	85.86	57.40	87.00	80.67	87.70	73.33	87.72
DAPO	87.43	56.09	86.00	79.67	87.39	75.80	87.51
OPSD	90.58	57.75	84.50	80.33	87.37	78.93	87.50
BAS-OPD	92.67	63.08	87.25	83.33	90.23	83.07	90.00

3.3 ABLATION AND EFFICIENCY ANALYSIS

Accuracy–cost trade-off. Table 3 reports seven-benchmark accuracy and teacher cost. Figure 5 in the appendix visualizes the comparison for full querying, Entropy-25, and Learned-25. Full querying averaged 79.76%, versus 79.25%, 80.24%, and 81.60% for Random-25, Entropy-25, and Learned-25. Learned-25 improved accuracy by 2.35, 1.36, and 1.84 points over random selection, entropy selection, and full querying, respectively. The three budgeted policies used identical query ratios, isolating how teacher calls were allocated. Learned-25 exceeded full querying on every benchmark, with the largest gains on HR-Bench 4K (3.37 points) and V*Bench (3.15 points).

Table 3: Selection-policy accuracy and teacher cost. Query ratio is selected/candidate; token ratio is relative to full querying. Bold marks the best 25%-budget result.

Policy	Teacher Cost		Fine-Grained Visual Tasks			Holdout Tasks				
	Query Ratio	Token Ratio	V*Bench	Zoom Bench	HR-Bench 4K	MMVP	CV-Bench	MMStar	POPE	Avg.
Full querying	100.00	100.00	87.43	58.46	82.25	78.33	85.56	77.73	88.58	79.76
Random-25	25.00	26.28	86.43	56.51	82.88	78.33	85.77	77.00	87.83	79.25
Entropy-25	25.00	27.65	88.48	58.58	83.62	77.33	86.41	77.87	89.40	80.24
Learned-25	25.00	15.78	90.58	59.88	85.62	79.33	87.63	79.27	88.91	81.60

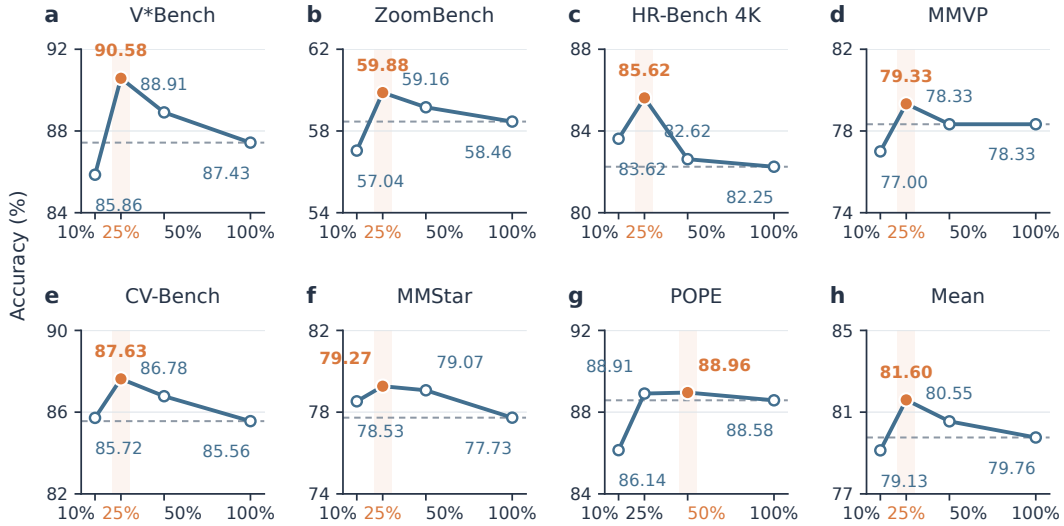


Figure 2: Qwen3.5-4B accuracy across teacher-query budgets on seven benchmarks; panel (h) is their mean. Orange marks each panel’s maximum; dashed lines mark full querying.

Learned-25 exceeded full querying and Random-25 on all seven benchmarks and outperformed Entropy-25 on six, with gains of 1.22–2.10 points; Entropy-25 led on POPE by 0.49 points. Unlike entropy, the learned target also scores confident, support-consistent teacher corrections. At a 25% query budget, Learned-25 achieved the highest mean accuracy while using 15.78% of full-query scored tokens (Table 3). Appendix C reports detailed token and wall-clock costs.

Effect of query budget. Figure 2 compares four budgets on seven benchmarks. The 25% budget leads six sweeps, while 50% leads on POPE by 0.05 points. At 10%, 25%, 50%, and 100%, the seven-benchmark means are 79.13%, 81.60%, 80.55%, and 79.76%, confirming non-monotonic returns.

Utility concentration and ranking behavior. Figure 3 audits 2,240 rollouts across ten batches. At 25%, learned ranking captures 58.14% of utility, versus 72.70% for the oracle and 25% for random ranking (Figure 3a). Utility decreases across rank bands; the top-quartile mean is 1.75 times the next quartile (5.52×10^{-3} versus 3.15×10^{-3} ; Holm-adjusted $p = 0.006$; Figure 3b). Using training-

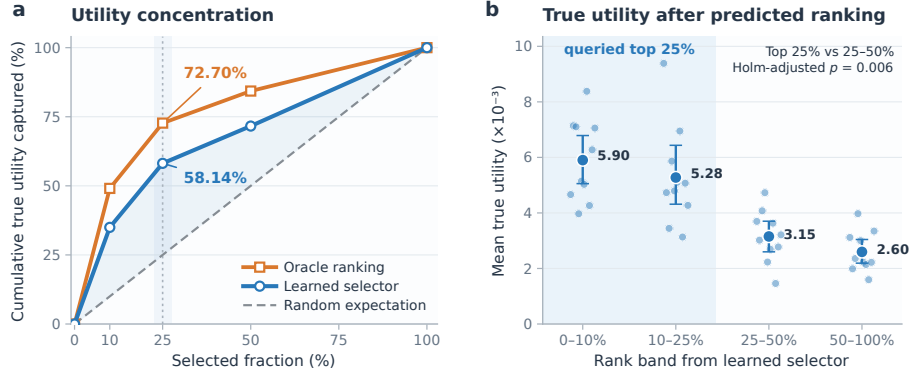


Figure 3: Utility diagnostics over 2,240 candidates from ten batches. (a) Cumulative utility captured by oracle, learned, and random rankings. (b) Mean utility by predicted rank band with 95% bootstrap confidence intervals; p is Holm-adjusted over three one-sided paired sign-flip tests.

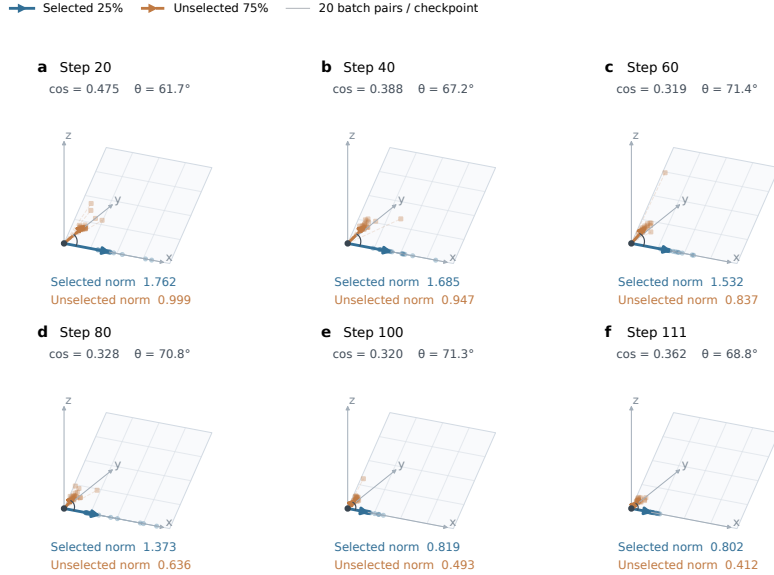


Figure 4: Gradient geometry for Learned-25 at six checkpoints. Arrows show selected and unselected mean gradients; faint vectors show 20 batch pairs per checkpoint. Coordinates are aligned for display.

distribution batches and a frozen final student instead of the unserialized EMA teacher, this audit diagnoses policy behavior rather than out-of-distribution generalization.

Gradient geometry. Figure 4 shows that, across six checkpoints, the selected 25% produced 1.66–2.16 $\times$ larger mean OPD gradient norms than the unselected 75%. Cosine similarities of 0.319–0.475 (angles of 61.7°–71.4°) indicate partial alignment. These single-trajectory results use 20 batch pairs per checkpoint and do not establish that unselected rollouts are harmful or that 25% is optimal.

Utility-target and selector-design ablations. Table 4 isolates top- k overlap, teacher confidence, ranking supervision, and random exploration. The full selector achieved the highest seven-benchmark average of 81.60%. Removing individual components lowered average accuracy by 2.79–3.66 points, with teacher confidence causing the largest drop, consistent with its role in down-weighting diffuse teacher distributions. On MMStar, the ablations caused larger drops of 7.80–9.87 points, showing that the gains extended to holdout evaluation, although their magnitude varied by benchmark.

Table 4: Component ablations of the learned utility selector across seven benchmarks. Avg. is the unweighted mean across the seven benchmarks.

Selector variant	Fine-Grained Visual Tasks			Holdout Tasks				Avg.
	V*Bench	Zoom Bench	HR-Bench 4K	MMVP	CV-Bench	MMStar	POPE	
Without top- k overlap (dc)	86.39	57.04	82.25	76.33	86.09	70.80	88.97	78.27
Without teacher confidence (do)	88.48	55.50	81.00	77.33	84.99	69.40	88.87	77.94
Without ranking loss	89.53	56.69	79.38	76.33	85.46	70.53	89.11	78.15
Without random exploration	85.86	58.70	83.38	77.00	86.23	71.47	89.03	78.81
Full learned selector (doc)	90.58	59.88	85.62	79.33	87.63	79.27	88.91	81.60

4 RELATED WORK

4.1 ON-POLICY DISTILLATION

Conventional distillation supervises a student on a fixed data distribution, whereas on-policy distillation evaluates the teacher along student-generated trajectories to reduce the train–inference mismatch. GKD trains on student-generated sequences, while MiniLLM and DistiLLM combine student rollouts with alternative divergence objectives or adaptive on- and off-policy mixtures (Agarwal et al., 2024; Gu et al., 2024; Ko et al., 2024). Recent self-distillation methods condition the same model on privileged reasoning traces or demonstrations, extending on-policy learning to mathematical reasoning and continual learning (Zhao et al., 2026; Shenfeld et al., 2026). A complementary analysis identifies teacher–student compatibility, novel teacher capability, and dense token-level supervision cost as central factors in successful on-policy distillation (Li et al., 2026). Privileged regional supervision has also been explored for multimodal perception (Yuan et al., 2026). BAS-OPD instead addresses query allocation: under a fixed budget, it selects which rollouts should receive supervision before the teacher forward pass.

4.2 FINE-GRAINED VISUAL UNDERSTANDING FOR MLLMS

Fine-grained MLLM perception is limited when small but decisive evidence is lost during image encoding. High-resolution architectures retain more detail through dynamic tiling, variable visual-token counts, or adaptive image slicing, as in InternVL 1.5, Qwen2-VL, and LLaVA-UHD (Chen et al., 2024b; Wang et al., 2024; Guo et al., 2024). Inference-time methods such as V* and DC² instead search, partition, or retrieve relevant regions for each query (Wu & Xie, 2024; Wang et al., 2025). These approaches improve access to local evidence, but increase visual processing or introduce additional inference steps. BAS-OPD is complementary: it changes neither the visual architecture nor the inference procedure, but allocates privileged regional supervision selectively during training and retains single-pass full-image inference.

5 CONCLUSION AND LIMITATIONS

We introduced BAS-OPD, which allocates crop-teacher supervision under a fixed query budget using online utility estimates. Learned-25 raised the Qwen3.5-4B seven-benchmark average from 79.76% to 81.60% while querying 25% of candidates and using 15.78% of full-query scored tokens; BAS-OPD 9B reached 84.23%. BAS-OPD thus improves the training accuracy–cost trade-off while preserving single-pass full-image inference.

Limitations.

Selection analyses use Qwen3.5-4B and one seed, and the budget sweep covers seven benchmarks; broader seeds, backbones, and matched 9B ablations remain untested. Wall-clock results depend on implementation and hardware utilization. Because the EMA teacher was not serialized, the frozen-selector audit measures training-distribution subset retrieval rather than out-of-distribution generalization.

REFERENCES

- Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024.
- Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models? In *Advances in Neural Information Processing Systems*, volume 37, 2024a.
- Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Jia-peng Luo, Zheng Ma, et al. How far are we to GPT-4V? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024b.
- Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*, 2024.
- Zonghao Guo, Ruyi Xu, Yuan Yao, Junbo Cui, Zhanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, and Gao Huang. LLaVA-UHD: An LMM perceiving any aspect ratio and high-resolution images. In *European Conference on Computer Vision*, 2024.
- Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. DistiLLM: Towards streamlined distillation for large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 24872–24895, 2024.
- Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan-gao Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026.
- Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 292–305, 2023.
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kel-ton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.
- Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL <https://qwen.ai/blog?id=qwen3.5>.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- Idan Shenfeld, Mehul Damani, Jonas Hübötter, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.
- Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Ziteng Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal LLMs. In *Advances in Neural Information Processing Systems*, volume 37, 2024a.
- Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9568–9578, 2024b.
- 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. *arXiv preprint arXiv:2409.12191*, 2024.

- Wenbin Wang, Liang Ding, Minyan Zeng, Xiabin Zhou, Li Shen, Yong Luo, Wei Yu, and Dacheng Tao. Divide, conquer and combine: A training-free framework for high-resolution image perception in multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7907–7915, 2025.
- Lai Wei, Liangbo He, Jun Lan, Lingzhong Dong, Yutong Cai, Siyuan Li, Huijia Zhu, Weiqiang Wang, Linghe Kong, Yue Wang, Zhuosheng Zhang, and Weiran Huang. Zooming without zooming: Region-to-image distillation for fine-grained multimodal perception. *arXiv preprint arXiv:2602.11858*, 2026.
- Penghao Wu and Saining Xie. V^* : Guided visual search as a core mechanism in multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13084–13094, 2024.
- Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Weinan Dai, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 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. *arXiv preprint arXiv:2605.18740*, 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. *arXiv preprint arXiv:2601.18734*, 2026.

A IMPLEMENTATION DETAILS

A.1 RANDOM AND ENTROPY SELECTION BASELINES

Random and student-entropy selection use the same query-budget interface as learned selection, but neither trains a utility predictor.

Random-25 samples $K = \lfloor 0.25|C| \rfloor$ eligible rollouts uniformly without replacement at each training step. Sampling uses a CPU random permutation with a seed derived from the selector seed and training step. Only these selected rollouts are sent to the crop teacher. The policy has no trainable selector and requires no additional student forward pass.

Entropy-25 is a fixed student-uncertainty heuristic. When token entropy is already available from the rollout path, each candidate is scored by its mean valid-token entropy. Otherwise, the selector uses mean sampled-token negative log-likelihood (NLL) from the stored rollout log-probabilities. The top- K rows are queried. This policy adds no trainable parameters and requires no additional student forward pass.

A.2 PREDICTOR CONFIGURATION AND FEATURE HANDLING

Algorithm 1 (line 4) forms the detached feature vector in Eq. 5. The sampled-token negative log-likelihood is $\ell_{i,t} = -\log p_{\text{old}}(y_{i,t} \mid x_i, q_i, y_{i,<t})$; $\bar{\ell}_i$ and $Q_{0.9}(\ell_i)$ are its response-level mean and 90th percentile. The remaining features are mean token entropy $\bar{H}_i$ when available, response length T_i , prompt length L_i , and crop-to-full-image area ratio a_i .

The predictor f_ψ consists of layer normalization, one hidden linear layer with a SiLU activation, and a scalar output. The operator $\text{Normalize}(\cdot)$ uses running feature statistics, updated on crop-eligible candidates during feature extraction and held fixed during replay updates.

Table 5 gives the selector settings shared by Learned-10, Learned-25, and Learned-50. The predictor runs on the CPU and ranks candidates by predicted utility, without dividing by response-token cost. Running feature normalization uses exponential mean and variance with numerical constant 10^{-6} . Unavailable entropy, prompt-length, or crop-area values are represented by zero. If separate teacher top- k indices are unavailable, the implementation sets the overlap factor to one.

Table 5: Learned-selector configuration shared across query budgets.

Hyperparameter	Value
Feature dimension / hidden dimension	6 / 32
Selector optimizer / learning rate	Adam / 1×10^{-3}
Running-normalizer momentum	0.99
Replay capacity / start threshold	512 / 96 labels
Replay minibatch size	256
Selector updates per training step	2
Replay age half-life	10 steps
Pairwise ranking weight	0.2
Minimum utility gap for a pair	1×10^{-6}
Exploration ratio	0.10
Selector seed	0
Score mode	predicted utility

A.3 SELECTOR TRAINING AND UPDATE HANDLING

Algorithm 1 (line 14) stores observed feature–utility pairs in a bounded first-in, first-out replay buffer together with their collection steps. Replay updates require newly observed valid pairs and the minimum buffer size specified in Table 5. Each update samples a minibatch uniformly without replacement, capped by the current buffer size. At training step s , an item collected at s_i receives the recency weight $w_i = 2^{-(s-s_i)/h}$, where h is the age half-life.

For each replay update, we normalize stored features using the current statistics without changing those statistics, and recompute $\hat{u}_i$ with the current predictor while retaining gradients to ψ . The weighted Huber term in Eq. 10 is

$$\mathcal{L}_{\text{reg}} = \frac{\sum_i w_i \text{Huber}(\hat{u}_i - u_i)}{\sum_i w_i}. \quad (12)$$

The ranking term $\mathcal{L}_{\text{rank}}$ trains the ordering of labeled samples. Each pair (i, j) with $|u_i - u_j| > 10^{-6}$ contributes $\text{softplus}[-\text{sgn}(u_i - u_j)(\hat{u}_i - \hat{u}_j)]$, weighted by $\sqrt{w_i w_j}$. Features and targets are detached, and the selector has a separate optimizer, so $\mathcal{L}_{\text{sel}}$ does not update the MLLM.

The denominator of Eq. 11 is clamped to at least one. When $\mathcal{S} = \emptyset$, crop-teacher scoring is skipped and the distillation term is zero; the remaining updates follow the standard OPD optimizer and teacher-update rules. The teacher EMA follows a valid student update (Algorithm 1, line 13). An empty selected set provides no new utility labels and therefore does not trigger a replay update.

B TRAINING AND EVALUATION DETAILS

B.1 TRAINING CONFIGURATION

Table 6 records the batch configuration and step counts for the four retained policy runs. The maximum model length is 9,216 tokens; the optimizer and remaining training settings are specified in the main experimental setup.

Table 6: Training configurations for the four selection-policy runs.

Configuration	Full querying	Random-25	Entropy-25	Learned-25
Prompt batch	28	28	28	28
Rollouts per prompt	8	8	8	8
Candidates per step	224	224	224	224
Selected per step	224	56	56	56
Completed steps	111	111	111	111
Internal query ratio	100%	25%	25%	25%
Training seed	42	42	42	42

B.2 EVALUATION PROTOCOL

Generation uses temperature 0 and seed 42. Benchmark-native option or exact-answer parsing is used when available; remaining free-form answers are judged with GPT-5.5 at temperature 0. CV-Bench is reported as the macro-average of its 2D and 3D accuracy categories, and POPE is reported by accuracy in the policy comparison. The POPE evaluation contains 9,000 samples.

C TEACHER-SUPERVISION COST

Training cost is recorded directly as the number of selected teacher calls, $|\mathcal{S}|$, and the number of response tokens scored by the teacher, $\sum_{i \in \mathcal{S}} T_i$.

Figure 5 summarizes the average accuracy and teacher cost. Tables 7 and 8 provide the recorded supervision counts and observed training times underlying the main accuracy–cost comparison.

Teacher-scored tokens. Random-25 and Entropy-25 used 26.28% and 27.65% of full-query teacher-scored tokens, whereas Learned-25 used 15.78% (Table 3). These reductions of 10.50 and 11.87 percentage points reflect shorter selected responses, since all three policies queried one quarter of their candidates. Learned-25 therefore combined the highest aggregate accuracy with the fewest scored tokens.

Observed training time. Wall-clock savings were smaller: Learned-25 reduced accumulated teacher-forward time by 16.02% and total step time by 9.84% relative to full querying. Student generation, communication, data loading, and rollout scheduling remain substantial costs. We therefore

report teacher calls, scored tokens, teacher-forward time, and total step time together, with measurement details and comparability limits in Table 8.

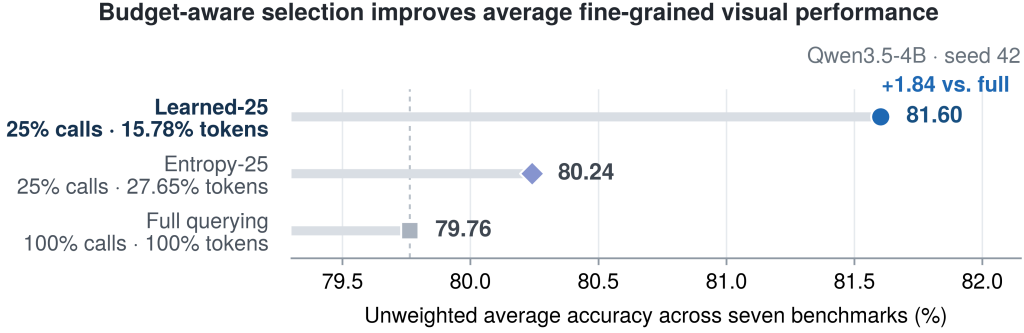


Figure 5: Accuracy–cost comparison on Qwen3.5-4B (seed 42). Points show the unweighted mean accuracy across seven benchmarks. Row labels report the teacher-query budget and the ratio of scored tokens to full querying.

Table 7: Measured teacher-supervision cost. Token ratios use Full querying as the denominator. For Learned-25, the 25% budget is computed directly from the recorded aggregate counts as 6,216/24,864.

Policy	Candidates	Teacher calls	Query ratio	Scored tokens	Tokens vs. Full	Max overrun
Full querying	24,864	24,864	100.00%	2,167,850	100.00%	0
Random-25	24,864	6,216	25.00%	569,674	26.28%	0
Entropy-25	24,864	6,216	25.00%	599,321	27.65%	0
Learned-25	24,864	6,216	25.00%	342,089	15.78%	0

Table 8: Observed training time on the original machines. These measurements are secondary because code path, device utilization, and rollout scheduling are not fully controlled across runs.

Policy	Teacher forward sum	Mean teacher/step	Mean step	Total step time
Full querying	5,174.38 s	46.62 s	499.88 s	55,487.15 s
Random-25	4,705.55 s	42.39 s	468.95 s	52,053.55 s
Entropy-25	4,690.39 s	42.26 s	469.30 s	52,092.85 s
Learned-25	4,345.38 s	39.15 s	450.70 s	50,027.87 s