RESTORE TEXT WITHOUT BREAKING VISION: VISION-PRESERVING ON-POLICY DISTILLATION FOR VLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language models (VLMs) are built by augmenting a pre-trained large language model (LLM) with a vision encoder and fine-tuning on multimodal data, but this process often weakens the language capabilities inherited from the base LLM. A natural remedy is to distill the base LLM back into the VLM. On-policy distillation (OPD), which trains the student on its own sampled responses under teacher supervision, appears well suited to this setting because the VLM student and its base LLM teacher already have closely aligned output distributions. However, this intuition is incomplete in multimodal models. When applied with text-only rollouts, OPD improves language performance but substantially degrades vision performance, revealing a failure mode we term cross-modal distillation interference. We attribute this to a *shared-parameter asymmetry*. In text-only OPD, the student generates rollouts in pure-text mode, which implicitly regularizes its text-mode policy and keeps it close to the original student. In contrast, the vision-conditioned policy is not directly sampled during text-only OPD, but still shares the parameters updated by text-mode rollouts. It therefore absorbs these updates without direct regularization, leading to substantially greater drift than the text-mode policy. Motivated by this analysis, we propose **Vision-Preserving On-Policy Distillation (VP-OPD)**, a structure-aware and mechanism-driven approximation to constrained distillation that restricts text-only updates to parameters less critical for vision-conditioned behavior. VP-OPD estimates structured vision importance on a small multimodal calibration set, freezes high-importance components, and performs distillation only on the remaining subnetwork. Experiments show that, with less than 40% trainable parameters, VP-OPD recovers at least 95% of the language gains of full-parameter OPD while substantially preserving vision performance, achieving a significantly better text–vision tradeoff.

1 INTRODUCTION

Vision-language models (VLMs) are commonly built by equipping a pre-trained large language model (LLM) with a vision encoder and fine-tuning on multimodal data, achieving strong performance on visual question answering, chart reasoning, and document comprehension (Liu et al., 2023; 2024a; An et al., 2025b; Bai et al., 2025; Zhu et al., 2025a). However, despite their strong LLM backbones, VLMs often exhibit substantial degradation in language capabilities, particularly in mathematical reasoning, code generation, and instruction following (Huang et al., 2024b; Chen et al., 2025b; Bousselham et al., 2025) This degradation is typically attributed to modality interference or catastrophic forgetting, where multimodal fine-tuning repurposes shared parameters to encode visual information, thereby distorting pre-trained language representations (Zhai et al., 2023; Hwang et al., 2026; Irawan et al., 2026) Notably, this interference can also adversely affect vision tasks, many of which rely on strong underlying language reasoning abilities (Huang et al., 2025; Wei et al., 2025). Prior work has explored low-rank adaptation(Hu et al., 2022), additional alignment modules (Wang et al., 2025; Zhang et al., 2024), model merging(Li et al., 2025; Yu & Ananiadou, 2025; Zeng et al., 2025), and large-scale retraining with mixed data to mitigate this issue. However, these approaches either restrict model capacity, introduce additional inference overhead, or require expensive retraining. As a result, effectively recovering language capabilities in VLMs while preserving vision performance remains an open challenge.

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.

A natural solution is to distill the VLM from its base LLM, which remains independently available after multimodal training. In particular, on-policy distillation (OPD) (Agarwal et al., 2024; Lu & Thinking Machines Lab, 2025) is a promising candidate because it trains the student on its own sampled responses to match the teacher's token-level predictions, and has shown strong empirical performance in language-model compression and knowledge transfer (Bai et al., 2025; DeepSeek-AI, 2026; GLM-5-Team et al., 2026). We argue that OPD is well-suited to VLM recovery not merely due to distributional similarity, but because it induces a **self-reinforcing update dynamic**: the student is trained on trajectories it already prefers, while the update remains anchored to the base LLM policy (Zhu et al., 2025b; Chen et al., 2025a; Shenfeld et al., 2025; 2026; Zhao et al., 2026). However, we show that this intuition is incomplete in multimodal settings: when OPD is applied using **text-only rollouts**, language performance improves but vision performance degrades substantially. We refer to this phenomenon as *cross-modal distillation interference*, which reveals a key limitation of naively applying OPD to shared-parameter multimodal models.

We trace this failure to a shared-parameter asymmetry in VLMs. The same parameters induce two conditional policies: a text-mode policy on text-only inputs and a vision-conditioned policy on image-text inputs. Under text-only OPD, the text-mode policy is sampled to generate rollouts, which implicitly regularizes it and keeps it relatively close to the original VLM. In contrast, the vision-conditioned policy receives no direct constraint and instead passively absorbs parameter updates driven by the text-mode policy. Figure 1 shows the consequence: text-only OPD induces much larger drift in the vision-conditioned policy than in the text-mode policy, and this drift coincides with severe degradation on vision benchmarks. The core problem is therefore not merely that OPD changes the model, but that it regularizes the sampled conditional policy while leaving the unsampled one insufficiently constrained.

Motivated by this analysis, we propose **Vision-Preserving On-Policy Distillation (VP-OPD)**, a structure-aware OPD method for VLMs (overview in Fig. 2). The central idea is to approximate constrained OPD by restricting text-only distillation updates to parameters that are less important to the vision-conditioned policy. Concretely, VP-OPD defines structured units, including attention heads and MLP intermediate channels, as the granularity for update restriction, estimates their vision importance via first-order Taylor saliency on a small multimodal calibration set, and freezes the high-importance units while applying OPD to the remaining subnetwork. In this way, VP-OPD restores language capabilities while preserving vision-critical structures. Extensive experiments on LLaVA-OneVision-1.5 and LLaVA-OneVision-2.0 show that VP-OPD recovers most of the language gains of full-parameter OPD while substantially preserving vision performance, yielding a significantly better text–vision tradeoff.

Our main contributions are as follows:

- We identify *cross-modal distillation interference* as a previously underexplored failure mode of text-only on-policy distillation in VLMs, where language capabilities improve but vision performance degrades sharply.
- We explain this phenomenon through an asymmetric implicit-regularization mechanism across conditional policies: on-policy training constrains the rollout-generating text-mode policy but leaves the shared-parameter vision-conditioned policy insufficiently protected.
- We propose **Vision-Preserving On-Policy Distillation (VP-OPD)**, an importance-guided OPD recipe that restricts distillation updates to a sparse subnetwork and achieves a substantially better text–vision tradeoff on LLaVA-OneVision.

2 Understanding Vision Degradation in Text-Only OPD for VLMs

2.1 On-Policy Distillation

On-policy distillation (OPD) (Agarwal et al., 2024; Lu & Thinking Machines Lab, 2025) trains the student to match the teacher's token-level predictions on the student's own generated responses, rather than on a fixed corpus. Concretely, given a teacher LLM $q_\phi(y \mid x)$ and a student with parameters θ, the OPD loss is:

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \mathbb{E}_{x\sim\mathcal{D}}\, \mathbb{E}_{y\sim\pi_\theta(\cdot|x)} \left[\sum_{n=1}^{L} \mathrm{KL}\big(\pi_\theta(\cdot \mid x, y_{<n}) \,\big\|\, q_\phi(\cdot \mid x, y_{<n})\big) \right], \tag{1}$$

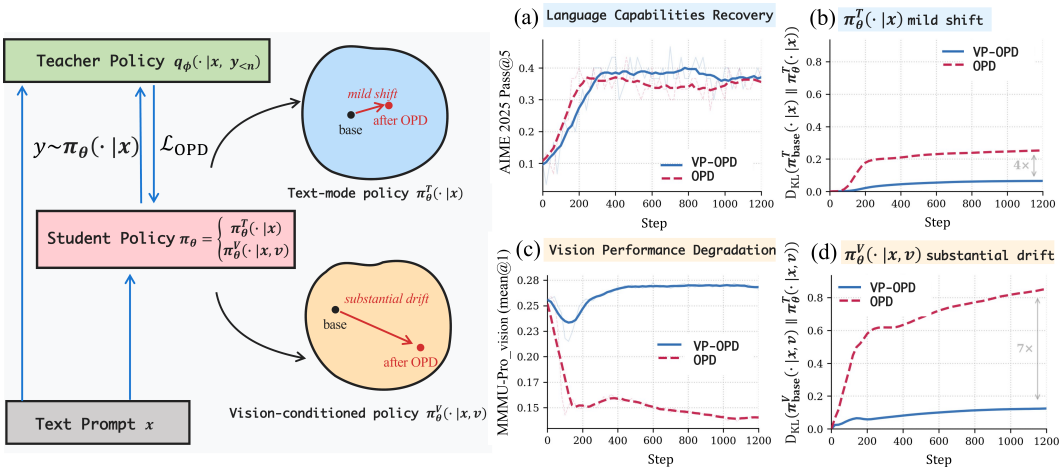


Figure 1: **Mechanism and empirical analysis of cross-modal interference in text-only OPD. Left:** The text-mode policy π_θ^T drifts only mildly from the base, while the vision-conditioned policy π_θ^V drifts substantially. **Right:** Training dynamics on LLaVA-OV-1.5-4B, OPD (dashed) vs. VP-OPD (solid). **(a)** AIME-25 pass@5: both improve. **(b)** text-mode KL: both drift mildly. **(c)** MMMU-Pro$_{\text{vision}}$: OPD collapses, VP-OPD stable. **(d)** vision-conditioned KL: OPD diverges, VP-OPD bounded.

where $\mathcal{D}$ is a prompt dataset and π_θ denotes the student's current policy. In our implementation, we optimize a top-K truncation of Eq. (1) for tractability. Let $c_n = (x, y_{<n})$ and let $\mathcal{S}_K(c_n) = \text{TopK}_{q_\phi}(c_n)$ denote the K highest-probability tokens under the teacher at prefix c_n. The resulting top-K truncated OPD objective is:

$$\mathcal{L}_{\text{OPD}}^{\text{top-}K}(\theta) = \mathbb{E}_{x\sim\mathcal{D}}\,\mathbb{E}_{y\sim\pi_\theta(\cdot|x)}\left[\sum_{n=1}^{L}\sum_{a\in\mathcal{S}_K(c_n)}\frac{\pi_\theta(a\mid c_n)}{\sum_u \pi_\theta(u\mid c_n)}\log\frac{\pi_\theta(a\mid c_n)\sum_u q_\phi(u\mid c_n)}{q_\phi(a\mid c_n)\sum_u \pi_\theta(u\mid c_n)}\right], \tag{2}$$

where the inner sums over u range over $\mathcal{S}_K(c_n)$. This top-K surrogate provides a denser training signal than single-token sampling while remaining cheaper than full-vocabulary KL. We use Eq. (2) as the OPD objective throughout this paper.

OPD is most effective when student and teacher have closely aligned output distributions (Li et al., 2026; Fu et al., 2026; Zhu et al., 2026), making the base LLM a natural teacher for its derived VLM; on-policy training also preserves capabilities outside the training distribution (Shenfeld et al., 2026; Chen et al., 2025a). Consistent with this, text-only OPD restores language capabilities and preserves out-of-distribution text-only performance, yet still substantially degrades vision tasks (Fig. 1).

2.2 Why Does Text-Only OPD Degrade Vision?

To analyze this degradation, we view VLM generation as a conditional policy parameterized by θ. On a text-only input x, the model defines the text-mode policy π_θ^T; on an image-text input (x, v), it defines the vision-conditioned policy π_θ^V:

$$\pi_\theta^T(y\mid x) = \prod_{n=1}^{L}\pi_\theta^T(y_n\mid x, y_{<n}), \qquad \pi_\theta^V(y\mid x, v) = \prod_{n=1}^{L}\pi_\theta^V(y_n\mid x, v, y_{<n}). \tag{3}$$

We use LLaVA-OneVision-1.5-4B-Instruct as the student and its base LLM, Qwen3-4B-Instruct-2507, as the teacher. During text-only OPD training, we periodically measure the KL drift of π_θ^T on a text-only probe set and of π_θ^V on a vision-conditioned probe set (both relative to the base student), together with task accuracy on AIME-25 (pass@5) and MMMU-Pro$_{\text{vision}}$ (mean@1).

Cross-modal asymmetry. As Fig. 1 shows, the two policies behave very differently during training. On the text-only probe set (Fig. 1(b)), OPD increases the KL of π_θ^T from the base model only moderately. On the vision-conditioned probe set (Fig. 1(d)), however, the KL of π_θ^V rises far more

sharply and reaches a much higher final value. This policy-level asymmetry is consistent with the observed task performance: AIME-25 pass@5 improves steadily (Fig. 1(a)), but MMMU-Pro$_{\text{vision}}$ mean@1 collapses (Fig. 1(c)). Text-only OPD thus succeeds at its intended objective of recovering language capabilities, but at the cost of a much larger drift in π_θ^V, which in turn destroys the performance in vision tasks.

This asymmetry stems from a key property of on-policy training: it implicitly regularizes *only* the policy that generates the rollouts, preventing it from drifting far from the base model (Shenfeld et al., 2025). In text-only OPD, that policy is π_θ^T, which is therefore kept close to the base model. π_θ^V shares every parameter with π_θ^T and is moved by the same gradient updates, but it is never sampled during training and receives no analogous constraint. Preserving vision-task performance during text-only OPD therefore requires introducing an *explicit* constraint on π_θ^V to compensate for the missing implicit one.

Implications for method design. Recent work shows that structural components in LLMs contribute unevenly to different capabilities (Ma et al., 2023; Gao et al., 2025; Shi et al., 2025), and that effective adaptation can be achieved by updating only a parameter subset (Mukherjee et al., 2025). These findings suggest that the drift of π_θ^V can be constrained by selectively restricting parameter updates during text-only OPD.

3 METHOD

VP-OPD is a structure-aware and mechanism-driven approximation to constrained distillation for shared-parameter vision-language models. Its goal is to recover language capabilities through text-only OPD while explicitly limiting drift in the unsampled vision-conditioned policy. VP-OPD consists of three steps: **(1) Vision Importance Estimation** (Section 3.1), which aggregates gradient-based saliency into structured units; **(2) Structured Sparse Subnetwork Construction** (Section 3.2), which selects low-importance units per layer and propagates the resulting mask; and **(3) Vision-Preserving On-Policy Distillation Training** (Section 3.3), which performs masked-gradient OPD on the selected subnetwork. Figure 2 summarizes the pipeline.

3.1 VISION IMPORTANCE ESTIMATION

To approximate which shared structures are most critical for the vision-conditioned policy, we estimate parameter importance on a small multimodal calibration set $\mathcal{D}_{\text{mm}}$ using first-order Taylor expansion (Molchanov et al., 2016; Ma et al., 2023). Since Taylor importance scores are data-dependent (Williams & Aletras, 2024), evaluating on $\mathcal{D}_{\text{mm}}$ makes the scores reflect each parameter's contribution to vision-conditioned predictions.

For a parameter w_i in an attention or MLP projection matrix, we define its vision importance as the loss change when w_i is set to zero. A second-order Taylor expansion gives

$$|\mathcal{L}_{\text{NTP}}(\mathcal{D}_{\text{mm}};\, \theta) - \mathcal{L}_{\text{NTP}}(\mathcal{D}_{\text{mm}};\, \theta_{w_i \leftarrow 0})| \;=\; \left|\, g_i\, w_i \;-\; \tfrac{1}{2}\, H_{ii}\, w_i^2 \;+\; \mathcal{O}(w_i^3) \,\right|, \tag{4}$$

with $g_i = \partial \mathcal{L}_{\text{NTP}} / \partial w_i$ and $H_{ii} = \partial^2 \mathcal{L}_{\text{NTP}} / \partial w_i^2$. The Hessian is intractable on LLMs ($\mathcal{O}(N^2)$) and the cubic remainder is negligible, so following (Molchanov et al., 2016) we retain only the first-order term, $|g_i\, w_i|$. Squaring gives a non-negative score that prevents cancellation during aggregation and amplifies high-importance parameters:

$$s_i = (g_i\, w_i)^2 . \tag{5}$$

All scores are obtained from a single forward-backward pass on $\mathcal{D}_{\text{mm}}$, making the computation negligible relative to training.

3.2 STRUCTURED SPARSE SUBNETWORK CONSTRUCTION

We compute the per-parameter vision importance in Eq. (5) only on the query projection in attention and the gate and up projections in the MLP, which respectively determine attention patterns and MLP intermediate activations, and use these scores as the basis for subnetwork construction. Within each transformer block, VP-OPD aggregates the parameter scores into structured units (KV-groups in

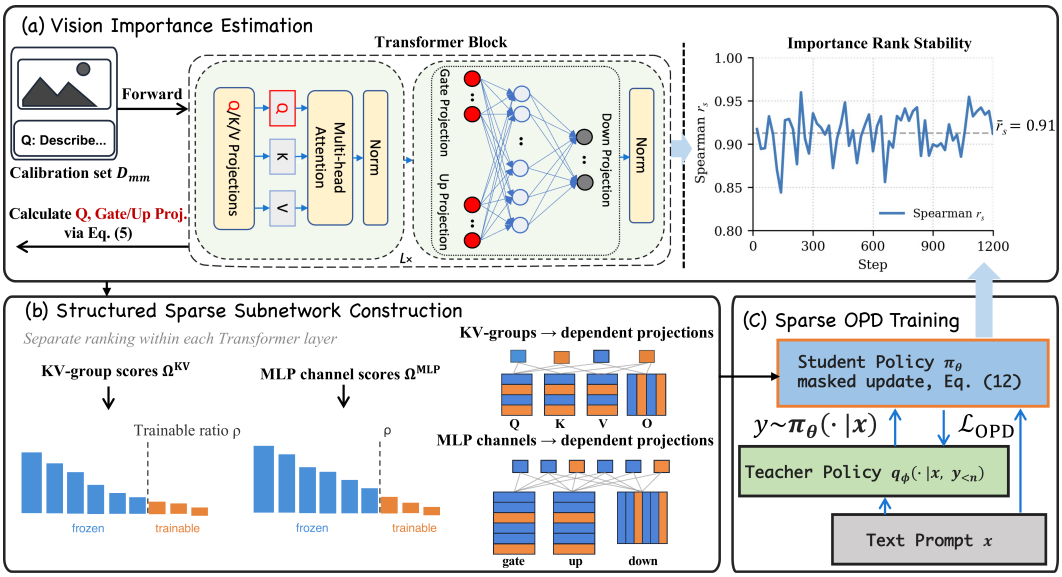


Figure 2: **Overview of VP-OPD. (a)** Vision importance is estimated *once before training* on a small multimodal calibration set by computing first-order Taylor importance on input-side projections (Q, gate, up); the top-right inset shows that recomputing the ranking during training yields a nearly invariant order (Spearman $\bar{r}_s \approx 0.91$), so one-shot scoring is sufficient. **(b)** Element-wise scores are aggregated into structured units (KV-groups in attention, intermediate channels in MLP) and ranked separately within each Transformer layer. Given a trainable ratio ρ, the lowest-importance units form the sparse subnetwork, and the mask is broadcast to all dependent projections (Q/K/V/O for attention, gate/up/down for MLP). **(c)** Text-only OPD is performed with gradient updates restricted to the selected subnetwork, keeping high-importance structures frozen.

attention and MLP intermediate channels) and selects the lowest-importance fraction of units at the target trainable ratio. The selected structural units are then mapped onto their dependent projections to form the final trainable subnetwork. This subnetwork contains complete structural units, since weights within each such unit are functionally coupled and should be frozen or trained jointly.

Structured Aggregation. Let $s^{(\cdot)}_{l,r,c} = \left(g^{(\cdot)}_{l,r,c} \cdot W^{(\cdot)}_{l,r,c}\right)^2$ denote the per-parameter vision importance (Eq. (5)) at entry (r, c) of projection matrix $W^{(\cdot)}_l$, where r and c index its output rows and input columns. For each attention head h at layer l, we sum $s^{(q)}_{l,r,c}$ over the rows of $W^{(q)}_l$ assigned to that head to obtain its per-unit vision importance:

$$\Omega^{\text{head}}_{l,h} = \sum_{r \in \mathcal{H}_{l,h}} \sum_{c} s^{(q)}_{l,r,c}, \tag{6}$$

where $\mathcal{H}_{l,h}$ is the set of $W^{(q)}_l$ rows belonging to head h. For each MLP intermediate channel m at layer l, we sum the per-parameter scores along its row of $W^{(\text{gate})}_l$ and $W^{(\text{up})}_l$:

$$\Omega^{\text{MLP}}_{l,m} = \sum_{c} s^{(\text{gate})}_{l,m,c} + \sum_{c} s^{(\text{up})}_{l,m,c}. \tag{7}$$

Under grouped-query attention (GQA) (Ainslie et al., 2023), multiple query heads share a single key/value projection, so any freeze decision must be made jointly over the heads of a KV-group. We therefore aggregate the per-head scores within each KV-group g before mask construction:

$$\Omega^{\text{KV}}_{l,g} = \sum_{h \in \mathcal{G}(g)} \Omega^{\text{head}}_{l,h}, \tag{8}$$

where $\mathcal{G}(g)$ is the set of head indices in group g. Multi-head attention is the special case where each KV-group contains a single head, in which case $\Omega^{\text{KV}}_{l,g} = \Omega^{\text{head}}_{l,h}$.

Per-Layer Mask Construction. Since modern LLMs (*e.g.*, the LLaMA and Qwen families) predominantly adopt GQA, we hereafter take the KV-group as the basic attention unit for mask construction. Within each layer, KV-groups and MLP intermediate channels are ranked separately, and the lowest-scoring fraction of each type is selected as trainable. Let N_l^{attn} and N_l^{MLP} denote the number of KV-groups and intermediate channels at layer l, and let $\rho \in (0, 1]$ be the trainable ratio. The per-unit binary masks are

$$M_{l,g}^{\text{attn}} \;=\; \mathbf{1}\big[\Omega_{l,g}^{\text{KV}} \le \tau_l^{\text{attn}}\big], \qquad M_{l,m}^{\text{MLP}} \;=\; \mathbf{1}\big[\Omega_{l,m}^{\text{MLP}} \le \tau_l^{\text{MLP}}\big], \tag{9}$$

where τ_l^{attn} and τ_l^{MLP} are the $\lfloor \rho\, N_l^{\text{attn}} \rfloor$-th and $\lfloor \rho\, N_l^{\text{MLP}} \rfloor$-th smallest values of $\{\Omega_{l,g}^{\text{KV}}\}_g$ and $\{\Omega_{l,m}^{\text{MLP}}\}_m$, respectively; $M{=}1$ marks a unit as trainable. We rank within each layer rather than globally across all layers to keep every layer's share of trainable capacity proportional.

Mask Propagation. Each unit's freeze decision is broadcast to all rows and columns of its dependent projections, yielding a parameter-level mask $\mathbf{M}_l^{(\cdot)}$ for each weight matrix. For an attention KV-group g at layer l:

$$\begin{aligned} \mathbf{M}_l^{(q)}[r,:] \;=\; \mathbf{M}_l^{(o)}[:,r] \;=\; M_{l,g}^{\text{attn}}, \quad \forall r \in \bigcup_{h \in \mathcal{G}(g)} \mathcal{H}_{l,h}, \\ \mathbf{M}_l^{(k)}[r',:] \;=\; \mathbf{M}_l^{(v)}[r',:] \;=\; M_{l,g}^{\text{attn}}, \quad \forall r' \in \mathcal{K}_{l,g}, \end{aligned} \tag{10}$$

where $\mathcal{K}_{l,g}$ indexes the $W_l^{(k)}, W_l^{(v)}$ rows shared by group g. For an MLP intermediate channel m at layer l:

$$\mathbf{M}_l^{(\text{gate})}[m,:] \;=\; \mathbf{M}_l^{(\text{up})}[m,:] \;=\; \mathbf{M}_l^{(\text{down})}[:,m] \;=\; M_{l,m}^{\text{MLP}}. \tag{11}$$

Parameters outside attention and MLP projections (*e.g.*, embeddings, layer norms, language model head) remain frozen throughout. Collecting $\{\mathbf{M}_l^{(\cdot)}\}$ over all weight matrices yields the full parameter-level binary mask $\mathbf{M}$.

3.3 Training with Vision-Preserving On-Policy Distillation

With the parameter-level mask $\mathbf{M}$ determined offline, training reduces to standard OPD with gradients masked at every step:

$$\theta \leftarrow \theta - \eta \left(\mathbf{M} \odot \nabla_\theta \mathcal{L}_{\text{OPD}}^{\text{top-}K}(\theta)\right), \tag{12}$$

where η is the learning rate and $\odot$ denotes element-wise multiplication. Parameters with $\mathbf{M}{=}0$ retain their pre-distillation values exactly, while the remaining parameters receive full-rank updates from the OPD objective. Because $\mathbf{M}$ is fixed and applied as a single element-wise product, VP-OPD incurs no per-step computation or memory overhead beyond standard OPD.

Algorithm 1 summarizes the full procedure: importance estimation (Steps 1–3) is a one-time offline computation requiring a single forward-backward pass over the calibration set, and the subsequent distillation (Steps 4–7) follows the standard OPD protocol with masked gradients.

Algorithm 1 Vision-Preserving On-Policy Distillation (VP-OPD)

Require: VLM with parameters θ; teacher LLM q_ϕ; multimodal calibration set $\mathcal{D}_{\text{mm}}$; text prompt set $\mathcal{D}$; trainable ratio ρ

1: **Importance Estimation:** Compute element-wise saliency $s_i = (w_i \cdot \partial \mathcal{L}_{\text{NTP}}/\partial w_i)^2$ via one forward-backward pass on $\mathcal{D}_{\text{mm}}$

2: **Structured Aggregation:** Aggregate $\{s_{l,r,c}^{(\cdot)}\}$ into KV-group scores $\{\Omega_{l,g}^{\text{KV}}\}$ and MLP channel scores $\{\Omega_{l,m}^{\text{MLP}}\}$ via Eqs. (6)–(8) and Eq. (7)

3: **Mask Construction:** For each layer l, select the bottom-$\lfloor \rho N_l^{\text{attn}} \rfloor$ KV-groups and bottom-$\lfloor \rho N_l^{\text{MLP}} \rfloor$ MLP channels by importance, then propagate the unit-level masks $\{M_{l,g}^{\text{attn}}\}, \{M_{l,m}^{\text{MLP}}\}$ to the dependent projections to form the parameter-level mask $\mathbf{M}$

4: **Sparse OPD:** For each training step:

5: Sample prompts $x \sim \mathcal{D}$; generate $y \sim \pi_\theta^T(\cdot \mid x)$

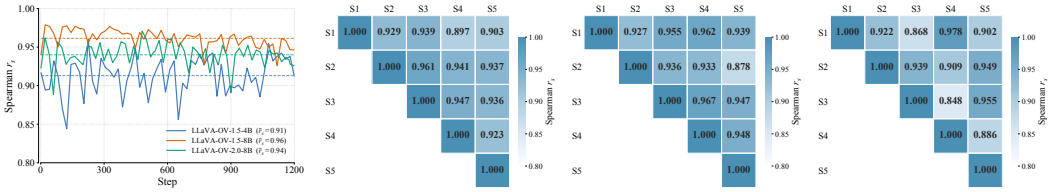
6: Compute $\mathcal{L}_{\text{OPD}}^{\text{top-}K}(\theta)$ via Eq. (2)

7: Update: $\theta \leftarrow \theta - \eta\,(\mathbf{M} \odot \nabla_\theta \mathcal{L}_{\text{OPD}}^{\text{top-}K})$

Ensure: Distilled VLM with restored language capabilities and preserved vision performance

Table 1: **Main results on LLaVA-OV-1.5-4B/8B and LLaVA-OV-2.0-8B.** We compare Base, SFT, LoRA, OPD, and VP-OPD across 7 text-centric and 10 vision-centric benchmarks. **Bold** marks the best result per model. Δ reports the block-average change relative to Base (↑ gain, ↓ drop).

		LLaVA-OV-1.5-4B					LLaVA-OV-1.5-8B					LLaVA-OV-2.0-8B				
	Benchmark	**Base**	**SFT**	**LoRA**	**OPD**	**VP-OPD**	**Base**	**SFT**	**LoRA**	**OPD**	**VP-OPD**	**Base**	**SFT**	**LoRA**	**OPD**	**VP-OPD**
Text-Centric	GPQA (avg@8)	32.95	**55.62**	42.49	44.95	45.01	27.21	**47.73**	47.66	45.45	46.34	32.13	53.91	51.52	**56.76**	55.93
	SuperGPQA	17.74	11.13	14.93	18.37	**19.93**	**19.77**	11.12	17.03	14.09	16.56	18.48	21.33	**22.42**	21.42	21.30
	MMLU-Pro	53.02	59.80	56.42	**67.26**	64.79	54.70	60.53	59.77	**64.57**	63.73	51.97	60.95	61.30	**71.26**	68.41
	AIME-25 (avg@16)	5.21	**36.88**	32.50	30.00	32.08	2.92	**21.46**	18.96	16.04	16.46	4.37	35.41	33.75	48.33	**49.17**
	MATH-500 (avg@8)	61.95	91.70	90.63	**92.60**	89.80	66.88	**85.75**	84.55	83.23	83.68	62.75	92.28	91.60	**93.68**	93.48
	$\text{IFEval}_{\text{loose}}$	81.65	61.51	75.23	**84.89**	84.41	**83.57**	64.87	80.82	79.74	83.33	**75.97**	30.31	73.20	70.93	73.01
	LiveCodeBench	25.21	30.81	32.42	**40.09**	37.25	27.11	25.12	23.32	**31.85**	**31.85**	28.82	45.97	43.41	**58.96**	55.92
	Avg.	39.68	49.64	49.23	**54.02**	53.32	40.31	45.23	47.44	47.85	**48.85**	39.21	48.59	53.89	**60.19**	59.60
	Δ	/	↑**9.96**	↑**9.56**	↑**14.35**	↑**13.65**	/	↑**4.92**	↑**7.14**	↑**7.54**	↑**8.54**	/	↑**9.38**	↑**14.68**	↑**20.98**	↑**20.39**
Vision-Centric	MMMU_{val}	52.70	49.44	52.78	53.11	**53.56**	**55.22**	52.33	53.22	46.67	54.78	53.33	28.56	**53.44**	26.00	46.56
	$\text{MMMU-Pro}_{\text{std}}$	35.30	33.12	35.09	35.38	**35.49**	37.46	**37.51**	37.28	36.59	37.05	**38.27**	13.64	37.40	14.62	34.16
	$\text{MMMU-Pro}_{\text{vis}}$	25.40	17.34	25.20	10.12	**26.60**	**28.21**	26.59	25.20	6.24	25.90	20.23	0.17	**23.41**	0.06	22.25
	MMStar	63.67	62.08	63.96	64.15	**65.47**	66.67	66.06	66.38	65.88	**66.86**	**63.28**	0.06	59.78	0.00	62.23
	DocVQA	94.40	97.28	97.08	97.44	**97.82**	**97.68**	97.51	97.51	97.10	97.52	**94.99**	88.98	94.92	76.66	94.40
	InfoVQA	76.10	76.45	**76.47**	76.43	76.45	78.30	77.86	77.74	77.48	**78.46**	**73.86**	71.21	71.12	57.33	68.69
	RealWorldQA	67.45	66.27	66.80	66.41	**68.50**	68.37	67.97	67.58	61.18	**68.50**	69.15	61.31	**69.54**	14.90	69.15
	AI2D	92.88	92.00	**93.04**	92.84	92.94	**93.65**	93.13	93.26	89.67	**93.65**	**92.94**	0.03	92.52	0.00	92.07
	MMBench	84.62	83.85	84.62	85.14	**85.22**	84.54	85.14	84.88	82.82	**85.31**	**85.22**	65.98	84.45	74.40	84.79
	PixMo-Count	65.36	65.73	64.79	65.73	**68.54**	62.17	61.99	61.24	62.92	**63.30**	**65.54**	50.94	65.36	11.42	**65.54**
	Avg.	65.79	64.36	65.98	64.68	**67.06**	**67.23**	66.61	66.43	62.66	67.13	**65.68**	38.09	65.19	27.54	63.98
	Δ	/	↓**1.43**	↑**0.20**	↓**1.11**	↑**1.27**	/	↓**0.62**	↓**0.80**	↓**4.57**	↓**0.09**	/	↓**27.59**	↓**0.49**	↓**38.14**	↓**1.70**



(a) Across training steps (b) LLaVA-OneVision-1.5-4B (c) LLaVA-OneVision-1.5-8B (d) LLaVA-OneVision-2.0-8B

Figure 3: **Stability of the offline importance ranking across three LLaVA-OneVision models. (a)** Spearman correlations with the base-model ranking remain high throughout training, with mean r_s values of 0.91, 0.96, and 0.94, respectively. **(b–d)** Rankings are consistent across five calibration sets (500 samples each), with mean correlations of 0.931, 0.939, and 0.916, respectively.

4 Experiments

4.1 Experimental Setup

We evaluate on LLaVA-OneVision-1.5 (An et al., 2025b) at 4B and 8B, and LLaVA-OneVision-2.0 (An et al., 2026) at 8B. The 4B student uses its base LLM Qwen3-4B-Instruct-2507 (Yang et al., 2025) as teacher. LLaVA-OneVision-2.0-8B also uses its base LLM, Qwen3-8B (Yang et al., 2025), as the teacher. For LLaVA-OneVision-1.5-8B, we use Qwen3-8B because its Qwen3-8B-Base backbone lacks stable instruction-following. VP-OPD uses the same trainable ratio, $\rho = 0.375$, for all three models. Training uses text-only prompts from Mixture-of-Thoughts (Hugging Face, 2025) (code, math, science). We compare VP-OPD with Base, SFT, LoRA (Hu et al., 2022), and full-parameter OPD. Details in Appendices C and D.

4.2 Main Results and Analysis

VP-OPD preserves vision while retaining most of OPD's language gains. Table 1 shows a consistent pattern across all three models. On text-centric benchmarks, VP-OPD remains close to full-parameter OPD in average performance. On vision-centric benchmarks, however, the difference

is much larger: VP-OPD stays near the base model, whereas full-parameter OPD degrades across all three models, more severely at 8B and most sharply on LLaVA-OneVision-2.0-8B, particularly on MMMU-Pro$_{\text{vision}}$. This is consistent with the mechanism in Sec. 2.2. Under text-only OPD, the text-mode policy remains relatively well behaved, but the unsampled vision-conditioned policy drifts much more strongly, producing cross-modal distillation interference. VP-OPD mitigates this failure mode by freezing vision-critical structures and concentrating language-side updates in a lower-importance subnetwork, thereby achieving a substantially better text–vision tradeoff.

VP-OPD outperforms both SFT and LoRA on the overall text–vision tradeoff. SFT on fixed teacher targets can degrade out-of-domain text performance, whereas OPD remains stable. This supports our earlier analysis that on-policy rollouts implicitly regularize the sampled text-mode policy. VP-OPD retains this stability while restricting updates to preserve vision-critical behavior (Table 1).

Additional analysis of LLaVA-OV-1.5-8B and LLaVA-OV-2.0-8B. On LLaVA-OV-1.5-8B, both OPD and VP-OPD yield smaller language gains than on 4B, while full-parameter OPD causes greater vision degradation. This may reflect the distribution gap between its Qwen3-8B-Base backbone and Qwen3-8B teacher. These results support our earlier argument that OPD benefits from closely aligned student and teacher output distributions, motivating recovery from the VLM's own base LLM. The larger drop on LLaVA-OV-2.0-8B may instead be associated with its broader vision-task alignment during continual pre-training, which text-only distillation can more strongly disrupt. VP-OPD nevertheless recovers substantial language capabilities while largely preserving vision performance on this model, further supporting its effectiveness.

4.3 Importance Ranking Stability

Stability of the Offline Importance Ranking. A key design choice in VP-OPD is to construct the sparse mask once before training and keep it fixed during distillation. Figure 3(a) compares training trajectories on a common display axis. The measured Spearman rank correlations r_s for OV-1.5-4B, OV-1.5-8B, and OV-2.0-8B remain high ($\bar{r}_s = 0.91$, $\bar{r}_s = 0.96$, and $\bar{r}_s = 0.94$, respectively). The measured results support the initial ranking as a reliable approximation to rankings recomputed during training across all three models.

Robustness to Calibration Sampling. To test how calibration sampling affects the ranking, we draw five independent sets of 500 samples (different random seeds; examples in Appendix H) and compute pairwise Spearman r_s across their induced rankings (Figure 3(b–d)). Using the same five sets for all three models, the mean pairwise correlations are 0.931 for OV-1.5-4B, 0.939 for OV-1.5-8B, and 0.916 for OV-2.0-8B. The ranking remains broadly consistent across calibration samples, with greater sampling sensitivity in OV-2.0-8B. Combined with the temporal stability above, this supports one-time offline mask construction and avoids the cost of online recomputation.

4.4 Ablation Study

Ablation analysis. We ablate four design choices of VP-OPD (Table 2), keeping ρ=0.375 on LLaVA-OneVision-1.5-4B

Table 2: **Component ablations of VP-OPD.** All variants use LLaVA-OneVision-1.5-4B with ρ=0.375. Rows (i)–(iv) correspond to the design choices discussed in the text.

Text-Centric Benchmarks								
Variant	**GPQA** avg@8	**Super GPQA**	**MMLU-Pro**	**AIME-25** avg@16	**MATH-500** avg@8	**IFEval loose**	**LiveCode Bench**	**Avg.**
VP-OPD	45.01	19.93	64.79	32.08	89.80	84.41	37.25	53.32
Does importance-guided masking matter?								
(i) Random mask	40.73	18.81	63.83	28.96	86.55	78.37	35.36	50.37
(i) Reversed mask	54.04	18.47	64.40	34.16	91.02	85.49	37.82	55.06
Trainable selection: low vision-importance vs. high text-importance								
(ii) Text importance	43.94	18.24	61.64	33.96	90.10	83.81	36.40	52.58
Scoring scope: all projections vs. input-side only (Q, gate, up)								
(iii) All projections	42.74	20.38	63.97	33.75	90.73	85.13	36.21	53.27
Ranking scope: per-layer vs. global								
(iv) Global ranking	33.33	17.54	59.70	20.41	82.48	82.97	32.33	46.97

Vision-Centric Benchmarks								
Variant	**MMMU val**	**MMMU-Pro std**	**MMMU-Pro vis**	**MMStar**	**RealWorld QA**	**MM Bench**	**PixMo Count**	**Avg.**
VP-OPD	53.56	35.49	26.60	65.47	68.50	85.22	68.54	57.63
Does importance-guided masking matter?								
(i) Random mask	52.78	35.95	12.89	64.36	66.06	84.97	65.54	54.65
(i) Reversed mask	52.33	34.68	9.25	64.30	66.01	85.14	64.61	53.76
Trainable selection: low vision-importance vs. high text-importance								
(ii) Text importance	52.11	35.15	17.46	65.16	67.32	85.31	65.92	55.49
Scoring scope: all projections vs. input-side only (Q, gate, up)								
(iii) All projections	52.33	35.78	24.68	65.32	67.06	85.14	65.54	56.55
Ranking scope: per-layer vs. global								
(iv) Global ranking	52.67	35.33	14.74	65.10	64.31	84.62	66.67	54.78

(trainable-ratio ablation in Appendix G.1).

(i) Importance-guided masking. Random masking degrades both text and vision, and reversed selection slightly improves text but causes a much larger vision drop, because vision-critical structures also carry language capability (consistent with the partial overlap in Appendix G) and absorb text-only distillation signal at the cost of substantial vision degradation; freezing high vision-importance structures and training only low vision-importance ones is therefore essential. **(ii) Vision- vs. text-importance ranking.** Selecting trainable units by high text importance underperforms VP-OPD on both text and vision averages, most visibly on MMMU-Pro$_{\text{vision}}$, because text importance is estimated without visual input and leaves many vision-critical units inside the trainable subnetwork. **(iii) Input-side vs. all-projection scoring.** Expanding scoring to all projections leaves text nearly unchanged but consistently lowers vision, most visibly on MMMU-Pro$_{\text{vision}}$ and PixMo-Count, indicating that input-side scoring is a more conservative identifier of units whose updates are less likely to disturb vision-conditioned behavior. **(iv) Per-layer vs. global ranking.** Global ranking substantially harms both text and vision, particularly on MMMU-Pro$_{\text{vision}}$ and RealWorldQA, since it over-concentrates trainable units in low-scoring layers and leaves other layers over-constrained or under-protected. **Overall,** the ablations support the mechanism underlying VP-OPD: vision preservation depends on identifying and freezing structures that are disproportionately important to the unsampled vision-conditioned policy.

More Analysis. (1) Generalization to additional VLM families is reported in Appendix E. **(2)** Comparisons with MOPD, OPD + Visual-KL, and related methods are reported in Appendix F. **(3)** Trainable-ratio sensitivity is analyzed in Appendix G.

5 Related Work

Catastrophic Forgetting in Multimodal Models. Catastrophic forgetting, the loss of prior knowledge when learning new tasks, is typically addressed by regularization (Kirkpatrick et al., 2017; Zenke et al., 2017), replay-based methods (Wang et al., 2024a; Huang et al., 2024a), and parameter-efficient methods (Hu et al., 2022; Dettmers et al., 2023). Recent work shows that on-policy training helps preserve a model's original capabilities (Shenfeld et al., 2025; Irawan et al., 2026; Zhu et al., 2025b; Chen et al., 2025a). We extend this to cross-modal OPD, asking whether distilling the base LLM recovers language without sacrificing vision.

Knowledge Distillation for Language Models. Knowledge distillation (Hinton et al., 2015) trains a student to match a teacher's output distribution, and on-policy distillation (OPD) (Agarwal et al., 2024; Lu & Thinking Machines Lab, 2025) extends this by training on the student's own rollouts, with broad adoption in recent LLM post-training (Yang et al., 2025; DeepSeek-AI, 2026; GLM-5-Team et al., 2026; Yang et al., 2026; Hübotter et al., 2026; Zhu et al., 2026; Shenfeld et al., 2026; Zhao et al., 2026). We extend this line to VLMs and identify cross-modal interference under text-only OPD.

Parameter Importance. Prior work measures parameter importance via curvature, sensitivity, activation-aware saliency, or output changes under perturbation (LeCun et al., 1989; Hassibi et al., 1993; Lee et al., 2019), and uses it to select low-importance parameters for offline fine-tuning or pruning (Ma et al., 2023; Frantar & Alistarh, 2023; Sun et al., 2023; Xia et al., 2023; Hwang et al., 2026). We instead use importance to constrain online updates in cross-modal on-policy distillation.

6 Conclusion

We studied how to recover the language capabilities of vision-language models without sacrificing vision, and showed that text-only on-policy distillation exhibits a failure mode we call *cross-modal distillation interference*. The core mechanism is a shared-parameter asymmetry in implicit regularization: text-only OPD regularizes the sampled text-mode policy, while the unsampled vision-conditioned policy shares the same updates without comparable protection and drifts much more severely. Motivated by this analysis, we proposed Vision-Preserving On-Policy Distillation (VP-OPD), a structure-aware and mechanism-driven approximation to constrained distillation that estimates vision-conditioned importance on a small multimodal calibration set, freezes vision-critical structures, and restricts updates to the remaining subnetwork. Experiments on LLaVA-OneVision-1.5 and LLaVA-OneVision-2.0

show that VP-OPD retains most of full-parameter OPD's language gains while substantially better preserving vision, yielding a markedly better text–vision tradeoff than full OPD, SFT, and LoRA.

AI USE STATEMENT

Generative AI tools were used to generate initial manuscript drafts, polish the writing, and automate the batch submission of experimental jobs. The authors reviewed all AI-assisted content and experimental outputs and take full responsibility for the final manuscript and reported results.

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 *The twelfth international conference on learning representations*, 2024.

Joshua Ainslie, James Lee-Thorp, Michiel De Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. Gqa: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 4895–4901, 2023.

Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Chunsheng Wu, Huajie Tan, Chunyuan Li, Jing Yang, Jie Yu, Xiyao Wang, Bin Qin, Yumeng Wang, Zizhen Yan, Ziyong Feng, Ziwei Liu, Bo Li, and Jiankang Deng. Llava-onevision-1.5: Fully open framework for democratized multimodal training, 2025a. URL https://arxiv.org/abs/2509.23661.

Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Didi Zhu, et al. Llava-onevision-1.5: Fully open framework for democratized multimodal training. *arXiv preprint arXiv:2509.23661*, 2025b.

Xiang An, Yin Xie, Feilong Tang, Yunyao Yan, Huajie Tan, Didi Zhu, Changrui Chen, Xiuwei Zhao, Bin Qin, Kaicheng Yang, Yifei Shen, Yuanhan Zhang, Kaichen Zhang, Wenkang Zhang, Zheng Cheng, Nansen Zhang, Chunsheng Wu, Chunjiang Ge, Zimin Ran, Dehua Song, Chunyuan Li, Shikun Feng, Ming Hu, Zhangquan Chen, Junbo Niu, Bo Li, Ziyong Feng, Ziwei Liu, Zongyuan Ge, and Jiankang Deng. Llava-onevision-2: Towards next-generation perceptual intelligence, 2026. URL https://arxiv.org/abs/2605.25979.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Walid Bousselham, Hilde Kuehne, and Cordelia Schmid. Vold: Reasoning transfer from llms to vision-language models via on-policy distillation. *arXiv preprint arXiv:2510.23497*, 2025.

Howard Chen, Noam Razin, Karthik Narasimhan, and Danqi Chen. Retaining by doing: The role of on-policy data in mitigating forgetting. *arXiv preprint arXiv:2510.18874*, 2025a.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024.

Shiqi Chen, Jinghan Zhang, Tongyao Zhu, Wei Liu, Siyang Gao, Miao Xiong, Manling Li, and Junxian He. Bring reason to vision: Understanding perception and reasoning through model merging. *arXiv preprint arXiv:2505.05464*, 2025b.

DeepSeek-AI. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 91–104, 2025.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1feb87871436031bdc0f2beaa62a049b-Abstract.html.

Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International conference on machine learning*, pp. 10323–10337. PMLR, 2023.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *arXiv preprint arXiv:2603.25562*, 2026.

Leo Gao, Achyuta Rajaram, Jacob Coxon, Soham V Govande, Bowen Baker, and Dan Mossing. Weight-sparse transformers have interpretable circuits. *arXiv preprint arXiv:2511.13653*, 2025.

GLM-5-Team, :, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, Huilong Chen, Jiajie Zhang, Jian Jiao, Jiaqi Guo, Jingsen Wang, Jingzhao Du, Jinzhu Wu, Kedong Wang, Lei Li, Lin Fan, Lucen Zhong, Mingdao Liu, Mingming Zhao, Pengfan Du, Qian Dong, Rui Lu, Shuang-Li, Shulin Cao, Song Liu, Ting Jiang, Xiaodong Chen, Xiaohan Zhang, Xuancheng Huang, Xuezhen Dong, Yabo Xu, Yao Wei, Yifan An, Yilin Niu, Yitong Zhu, Yuanhao Wen, Yukuo Cen, Yushi Bai, Zhongpei Qiao, Zihan Wang, Zikang Wang, Zilin Zhu, Ziqiang Liu, Zixuan Li, Bojie Wang, Bosi Wen, Can Huang, Changpeng Cai, Chao Yu, Chen Li, Chengwei Hu, Chenhui Zhang, Dan Zhang, Daoyan Lin, Dayong Yang, Di Wang, Ding Ai, Erle Zhu, Fangzhou Yi, Feiyu Chen, Guohong Wen, Hailong Sun, Haisha Zhao, Haiyi Hu, Hanchen Zhang, Hanrui Liu, Hanyu Zhang, Hao Peng, Hao Tai, Haobo Zhang, He Liu, Hongwei Wang, Hongxi Yan, Hongyu Ge, Huan Liu, Huanpeng Chu, Jia'ni Zhao, Jiachen Wang, Jiajing Zhao, Jiamin Ren, Jiapeng Wang, Jiaxin Zhang, Jiayi Gui, Jiayue Zhao, Jijie Li, Jing An, Jing Li, Jingwei Yuan, Jinhua Du, Jinxin Liu, Junkai Zhi, Junwen Duan, Kaiyue Zhou, Kangjian Wei, Ke Wang, Keyun Luo, Laiqiang Zhang, Leigang Sha, Liang Xu, Lindong Wu, Lintao Ding, Lu Chen, Minghao Li, Nianyi Lin, Pan Ta, Qiang Zou, Rongjun Song, Ruiqi Yang, Shangqing Tu, Shangtong Yang, Shaoxiang Wu, Shengyan Zhang, Shijie Li, Shuang Li, Shuyi Fan, Wei Qin, Wei Tian, Weining Zhang, Wenbo Yu, Wenjie Liang, Xiang Kuang, Xiangmeng Cheng, Xiangyang Li, Xiaoquan Yan, Xiaowei Hu, Xiaoying Ling, Xing Fan, Xingye Xia, Xinyuan Zhang, Xinze Zhang, Xirui Pan, Xu Zou, Xunkai Zhang, Yadi Liu, Yandong Wu, Yanfu Li, Yidong Wang, Yifan Zhu, Yijun Tan, Yilin Zhou, Yiming Pan, Ying Zhang, Yinpei Su, Yipeng Geng, Yong Yan, Yonglin Tan, Yuean Bi, Yuhan Shen, Yuhao Yang, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yurong Wu, Yutao Zhang, Yuxi Duan, Yuxuan Zhang, Zezhen Liu, Zhengtao Jiang, Zhenhe Yan, Zheyu Zhang, Zhixiang Wei, Zhuo Chen, Zhuoer Feng, Zijun Yao, Ziwei Chai, Ziyuan Wang, Zuzhou Zhang, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-5: from vibe coding to agentic engineering, 2026. URL https://arxiv.org/abs/2602.15763.

Babak Hassibi, David G Stork, and Gregory J Wolff. Optimal brain surgeon and general network pruning. In *IEEE international conference on neural networks*, pp. 293–299. IEEE, 1993.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

Jianheng Huang, Leyang Cui, Ante Wang, Chengyi Yang, Xinting Liao, Linfeng Song, Junfeng Yao, and Jinsong Su. Mitigating catastrophic forgetting in large language models with self-synthesized rehearsal. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1416–1428, 2024a.

Wenke Huang, Jian Liang, Zekun Shi, Didi Zhu, Guancheng Wan, He Li, Bo Du, Dacheng Tao, and Mang Ye. Learn from downstream and be yourself in multimodal large language model fine-tuning. *arXiv preprint arXiv:2411.10928*, 2024b.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Xu Tang, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

Hugging Face. Mixture-of-Thoughts. `https://github.com/huggingface/open-r1/Mixture-of-Thoughts`, 2025. GitHub repository.

Hyeontaek Hwang, Nguyen Dinh Son, and Daeyoung Kim. Model-dowser: Data-free importance probing to mitigate catastrophic forgetting in multimodal large language models. *arXiv preprint arXiv:2602.04509*, 2026.

Patrick Amadeus Irawan, Erland Hilman Fuadi, Shanu Kumar, Alham Fikri Aji, and Yova Kementchedjhieva. Lingudistill: Recovering linguistic ability in vision-language models via selective cross-modal distillation. *arXiv preprint arXiv:2604.00829*, 2026. URL `https://arxiv.org/abs/2604.00829`.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European conference on computer vision*, pp. 235–251. Springer, 2016.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114 (13):3521–3526, 2017.

Yann LeCun, John S. Denker, and Sara A. Solla. Optimal brain damage. In *Advances in Neural Information Processing Systems*, volume 2, 1989. URL `https://papers.nips.cc/paper_files/paper/1989/hash/6c9882bbac1c7093bd25041881277658-Abstract.html`.

Namhoon Lee, Thalaiyasingam Ajanthan, and Philip H. S. Torr. SNIP: Single-shot network pruning based on connection sensitivity. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=B1VZqjAcYX`.

Junlin Li, Guodong Du, Jing Li, Sim Kuan Goh, Wenya Wang, Yequan Wang, Fangming Liu, Ho-Kin Tang, Saleh Alharbi, Daojing He, et al. Multi-modality expansion and retention for llms through parameter merging and decoupling. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 30866–30887, 2025.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 26296–26306, 2024a.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pp. 216–233. Springer, 2024b.

Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. URL https://thinkingmachines.ai/blog/on-policy-distillation.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. Llm-pruner: On the structural pruning of large language models. *Advances in neural information processing systems*, 36:21702–21720, 2023.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 2200–2209, 2021.

Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1697–1706, 2022.

Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. *arXiv preprint arXiv:1611.06440*, 2016.

Sagnik Mukherjee, Lifan Yuan, Dilek Hakkani-Tur, and Hao Peng. Reinforcement learning finetunes small subnetworks in large language models. *arXiv preprint arXiv:2505.11711*, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less. *arXiv preprint arXiv:2509.04259*, 2025.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.

Cheng Shi, Yizhou Yu, and Sibei Yang. Vision function layer in multimodal llms. *arXiv preprint arXiv:2509.24791*, 2025.

Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. A simple and effective pruning approach for large language models. *arXiv preprint arXiv:2306.11695*, 2023.

ModelScope Team. EvalScope: Evaluation framework for large models, 2024. URL https://github.com/modelscope/evalscope.

P Team, X Du, Y Yao, K Ma, B Wang, T Zheng, K Zhu, M Liu, Y Liang, X Jin, et al. Supergpqa: Scaling llm evaluation across 285 graduate disciplines, 2025. *URL https://arxiv. org/abs/2502.14739*.

Bin Wang, Chunyu Xie, Dawei Leng, and Yuhui Yin. Iaa: Inner-adaptor architecture empowers frozen large language model with multimodal capabilities. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 21035–21043, 2025.

Yifan Wang, Yafei Liu, Chufan Shi, Haoling Li, Chen Chen, Haonan Lu, and Yujiu Yang. Inscl: A data-efficient continual learning paradigm for fine-tuning large language models with instructions. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 663–677, 2024a.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37: 95266–95290, 2024b.

Yana Wei, Liang Zhao, Jianjian Sun, Kangheng Lin, Jisheng Yin, Jingcheng Hu, Yinmin Zhang, En Yu, Haoran Lv, Zejia Weng, et al. Open vision reasoner: Transferring linguistic cognitive behavior for visual reasoning. *arXiv preprint arXiv:2507.05255*, 2025.

Miles Williams and Nikolaos Aletras. On the impact of calibration data in post-training quantization and pruning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10100–10118, 2024.

x.AI Corp. Grok-1.5 Vision Preview: Connecting the Digital and Physical Worlds with Our First Multimodal Model, 2024. URL https://x.ai/blog/grok-1.5v. Blog post.

Mengzhou Xia, Tianyu Gao, Zhiyuan Zeng, and Danqi Chen. Sheared llama: Accelerating language model pre-training via structured pruning. *arXiv preprint arXiv:2310.06694*, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled rlvr. *arXiv preprint arXiv:2604.03128*, 2026.

Zeping Yu and Sophia Ananiadou. Locate-then-merge: Neuron-level parameter fusion for mitigating catastrophic forgetting in multimodal llms. *arXiv preprint arXiv:2505.16703*, 2025.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9556–9567, 2024.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: a more robust multi-discipline multimodal understanding benchmark (2025). *URL https://arxiv. org/abs/2409.02813*, 2025.

Fanhu Zeng, Haiyang Guo, Fei Zhu, Li Shen, and Hao Tang. Robustmerge: Parameter-efficient model merging for mllms with direction robustness. *arXiv preprint arXiv:2502.17159*, 2025.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3987–3995. PMLR, 2017. URL https://proceedings.mlr.press/v70/zenke17a.html.

Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. Investigating the catastrophic forgetting in multimodal large language models. *arXiv preprint arXiv:2309.10313*, 2023.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, et al. Lmms-eval: Reality check on the evaluation of large multimodal models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 881–916, 2025.

Renrui Zhang, Jiaming Han, Chris Liu, Aojun Zhou, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. Llama-adapter: Efficient fine-tuning of large language models with zero-initialized attention. In *The Twelfth International Conference on Learning Representations*, 2024.

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.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025a.

Wenhong Zhu, Ruobing Xie, Weinan Zhang, and Rui Wang. Flexible realignment of language models. *arXiv preprint arXiv:2506.12704*, 2025b.

Wenhong Zhu, Ruobing Xie, Rui Wang, and Pengfei Liu. Hybrid policy distillation for llms. *arXiv preprint arXiv:2604.20244*, 2026.

A Positioning of Our Work

Our baselines cover the main post-training routes considered in this setting. SFT represents full-parameter supervised fine-tuning on teacher-generated data, LoRA represents adapter-style parameter-efficient tuning, and full-parameter OPD represents the direct text-only distillation baseline. We use LoRA as the representative adapter-style baseline, since many LoRA- and adapter-based methods share the same high-level recipe: they keep the pretrained weights fixed and train a small set of auxiliary parameters. Including many variants from this family would mainly test choices such as module parameterization, rank, and placement, rather than changing the comparison among SFT, adapter-style tuning, full-parameter OPD, and VP-OPD.

B Limitations

VP-OPD assumes access to the base LLM from which the VLM was derived, using it as both the distillation teacher and the reference for importance estimation. This restricts applicability to settings where the base LLM weights are available, excluding closed-source or API-only teacher models. Furthermore, our method is designed for the prevalent two-stage paradigm in which a pre-trained LLM is first built and then augmented with a vision encoder for multimodal fine-tuning. For VLMs trained end-to-end from scratch without a separable LLM stage, the notion of recovering degraded language capabilities via distillation from the base LLM does not directly apply.

C Additional Implementation Details

Common Setup. All methods train the LLM with AdamW in `bfloat16` on 8×NVIDIA H200 GPUs, on text-only data from Mixture-of-Thoughts (Hugging Face, 2025). Off-policy and on-policy configurations are summarized in Tables 3 and 4.

SFT. SFT fine-tunes the LLMs on the (prompt, reference response) pairs from Mixture-of-Thoughts.

LoRA. LoRA follows the same data as SFT, but freezes the LLM and trains only rsLoRA adapters (rank r=16, α=32, dropout 0.05) attached to the attention projections (`q_proj`, `k_proj`, `v_proj`, `o_proj`) and the MLP projections (`gate_proj`, `up_proj`, `down_proj`) in every transformer block.

OPD. OPD discards the reference responses and retains only the prompts: at each step the student generates on-policy rollouts, which the teacher scores to produce token-level distillation targets. We adopt the top-K-truncated reverse-KL objective in Eq. (2).

VP-OPD. VP-OPD shares the same rollout protocol, training budget, and objective (Eq. (2)) as OPD, and differs only in restricting updates to a vision-preserving subnetwork. Importance scores follow the first-order Taylor saliency $(w \cdot g)^2$ defined in Eq. (5), computed on 500 image-text samples drawn uniformly from the LLaVA-OneVision-1.5 mid-training (An et al., 2025a) and aggregated at structured granularity (per KV-group for attention, per channel for MLP). The least vision-important ρ=37.5% of components form the trainable subnetwork; this choice is justified by the ablation in Appendix G.1.

D Evaluation Protocol

Additional Experimental Settings. Text benchmarks: GPQA (Rein et al., 2023), SuperGPQA (Team et al.), MMLU-Pro (Wang et al., 2024b), AIME-25, MATH-500 (Hendrycks et al., 2021), IFEval (Zhou et al., 2023), LiveCodeBench (Jain et al., 2024). Vision benchmarks: MMMU (Yue et al., 2024), MMMU-Pro (std/vis) (Yue et al., 2025), MMStar (Chen et al., 2024), DocVQA (Mathew et al., 2021), InfoVQA (Mathew et al., 2022), RealWorldQA (x.AI Corp, 2024), AI2D (Kembhavi et al., 2016), MMBench (Liu et al., 2024b), PixMo-Count (Deitke et al., 2025). GPQA, AIME-25, and MATH-500 use temperature 0.7 with top-p=0.95 and report the average; others use greedy decoding.

Table 3: Off-policy training configuration (SFT and LoRA). Model-specific values follow the order LLaVA-OneVision-1.5-4B, LLaVA-OneVision-1.5-8B, and LLaVA-OneVision-2.0-8B.

Config	SFT	LoRA
Optimization		
Learning rate	5×10^{-5} / 5×10^{-5} / 5×10^{-5}	5×10^{-5} / 5×10^{-5} / 5×10^{-5}
LR schedule	cosine	cosine
Warmup ratio	0.05	0.05
Training budget		
Epochs	2	2
Global batch size	512	512
Max sequence length	32,768	32,768

Table 4: On-policy distillation configuration (OPD and VP-OPD). Model-specific values follow the order LLaVA-OneVision-1.5-4B, LLaVA-OneVision-1.5-8B, and LLaVA-OneVision-2.0-8B.

Config	OPD	VP-OPD
Optimization		
Learning rate	4×10^{-6} / 1×10^{-5} / 1×10^{-5}	4×10^{-6} / 1×10^{-5} / 1×10^{-5}
LR schedule	constant	constant
Warmup ratio	0.1	0.1
Training budget		
Epochs	1	1
Global batch size	256	256
Max prompt / response length	4,096 / 8,192	4,096 / 8,192
Distillation		
Rollouts per prompt (n)	1	1
Rollout temperature	1.0	1.0
Rollout top-p	0.95	0.95
Top-K truncation	256	256

We split evaluation across two open-source harnesses. GPQA, AIME-25, MATH-500, and LiveCodeBench are run under `evalscope`(Team, 2024); all other benchmarks are run under `lmms-eval`(Zhang et al., 2025), namely SuperGPQA, MMLU-Pro, IFEval for text, and the nine vision benchmarks (MMMU, MMMU-Pro standard and vision, MMStar, DocVQA, InfoVQA, RealWorldQA, AI2D, MMBench, PixMo-Count).

For GPQA, AIME-25, and MATH-500 we decode with temperature 0.7, top-p=0.95, top-k=20 and report the average over 8, 16, and 8 samples per problem, respectively. All other benchmarks use greedy decoding with a single pass (repeat $= 1$). Inference is served with vLLM in `bfloat16`: evalscope tasks use tensor-parallel size 2 and eval batch size 128; lmms-eval tasks run 8-way data parallel on a single node. We follow each harness's default generation length, chat template, and scoring rule for every benchmark, and adopt the official task splits (e.g., `gpqa_diamond`, `mmbench_en_dev`, `ai2d_no_mask`).

E GENERALIZATION ACROSS QWEN AND VLM FAMILIES

Table 5: **Results on Qwen2.5-VL-7B-Instruct with two language-model teachers.** We compare Base, full-parameter OPD, and VP-OPD across 7 text-centric and 10 vision-centric benchmarks. Avg. is the unweighted mean over benchmarks in each block. Δ reports the block-average change relative to Base.

		Teacher: OpenThinker3-7B			Teacher: Qwen2.5-7B-Instruct		
	Benchmark	**Base**	**OPD**	**VP-OPD**	**Base**	**OPD**	**VP-OPD**
Text-Centric	GPQA (avg@8)	33.46	**37.78**	36.59	**33.46**	33.05	32.97
	SuperGPQA	16.73	**18.89**	17.67	16.73	21.84	**22.94**
	MMLU-Pro	51.00	**53.88**	52.21	51.00	**52.07**	51.74
	AIME-25 (avg@16)	2.50	**17.29**	16.46	2.50	**3.04**	2.78
	MATH-500 (avg@8)	64.50	**75.28**	74.13	64.50	**70.23**	70.18
	$\text{IFEval}_{\text{loose}}$	**65.43**	64.15	64.52	65.43	64.78	**65.66**
	LiveCodeBench	15.92	**32.23**	30.37	15.92	**20.91**	19.75
	Avg.	35.65	**42.79**	41.71	35.65	37.99	**38.00**
	Δ	/	**↑7.14**	**↑6.06**	/	**↑2.34**	**↑2.35**
Vision-Centric	MMMU_{val}	51.00	35.11	**51.78**	**51.00**	44.89	50.89
	$\text{MMMU-Pro}_{\text{std}}$	36.42	35.65	**36.96**	36.42	36.29	**37.06**
	$\text{MMMU-Pro}_{\text{vis}}$	**33.06**	16.27	30.13	**33.06**	19.33	32.18
	MMStar	**62.07**	45.25	61.28	**62.07**	47.63	61.73
	DocVQA	**94.70**	94.21	94.11	94.70	94.74	**94.81**
	InfoVQA	80.21	**80.31**	80.16	**80.21**	77.96	79.27
	RealWorldQA	**69.15**	67.84	68.45	**69.15**	66.19	68.93
	AI2D	93.36	89.18	**94.05**	93.36	88.89	**93.71**
	MMBench	84.02	74.23	**84.34**	84.02	76.05	**85.22**
	PixMo-Count	57.30	57.49	**58.05**	57.30	56.04	**58.68**
	Avg.	**66.13**	59.55	65.93	66.13	60.80	**66.25**
	Δ	/	**↓6.58**	**↓0.20**	/	**↓5.33**	**↑0.12**

Setting. Qwen2.5-VL-7B-Instruct is distilled from either OpenThinker3-7B or Qwen2.5-7B-Instruct. We use the OPD rollout settings in Table 4, a learning rate of 10^{-5}, and $\rho = 0.25$ for VP-OPD.

Results and Analysis. Full-parameter OPD improves average text-centric performance but degrades vision performance with both teachers (Table 5). With OpenThinker3-7B, the average vision drop remains 5.44% even after excluding $\text{MMMU-Pro}_{\text{vis}}$. VP-OPD retains most of OPD's language gains while keeping average vision performance within 0.20% of Base. These results show that cross-modal distillation interference extends beyond LLaVA and $\text{MMMU-Pro}_{\text{vis}}$, and that VP-OPD remains effective in mitigating it.

F Comparison with Alternative Vision-Preservation Methods

Table 6: **Comparison with alternative vision-preservation methods on LLaVA-OV-1.5-4B.** We report 7 text-centric and 10 vision-centric benchmarks. Avg. is the mean within each block, and Δ is its change relative to Base, computed before rounding.

		LLaVA-OV-1.5-4B						
	Benchmark	**Base**	**OPD**	**OPD + Replay-SFT**	**EWC-style OPD**	**OPD + Visual-KL**	**MOPD**	**VP-OPD**
Text-Centric	GPQA (avg@8)	32.95	44.95	44.26	43.50	41.33	**47.68**	45.01
	SuperGPQA	17.74	18.37	18.79	18.32	17.52	**23.68**	19.93
	MMLU-Pro	53.02	67.26	65.86	65.05	65.79	**69.79**	64.79
	AIME-25 (avg@16)	5.21	30.00	**39.79**	30.83	23.54	12.92	32.08
	MATH-500 (avg@8)	61.95	92.60	**92.98**	91.55	89.63	80.72	89.80
	$\text{IFEval}_{\text{loose}}$	81.65	**84.89**	72.93	78.00	82.71	84.37	84.41
	LiveCodeBench	25.21	**40.09**	40.00	35.83	32.68	36.52	37.25
	Avg.	39.68	**54.02**	53.52	51.87	50.46	50.81	53.32
	Δ	/	**↑14.35**	**↑13.84**	**↑12.19**	**↑10.78**	**↑11.14**	**↑13.65**
Vision-Centric	MMMU_{val}	52.70	53.11	50.67	53.11	53.44	**54.98**	53.56
	$\text{MMMU-Pro}_{\text{std}}$	35.30	35.38	34.91	35.20	34.22	**38.72**	35.49
	$\text{MMMU-Pro}_{\text{vis}}$	25.40	10.12	2.66	12.75	14.10	**30.21**	26.60
	MMStar	63.67	64.15	63.62	**65.97**	64.52	63.71	65.47
	DocVQA	94.40	97.44	97.60	97.53	97.70	95.11	**97.82**
	InfoVQA	76.10	76.43	75.73	**76.67**	76.31	40.81	76.45
	RealWorldQA	67.45	66.41	66.41	67.45	66.54	**70.71**	68.50
	AI2D	92.88	92.84	92.71	92.75	92.52	92.48	**92.94**
	MMBench	84.62	85.14	84.88	**85.40**	85.31	81.92	85.22
	PixMo-Count	65.36	65.73	62.17	65.36	60.11	**69.21**	68.54
	Avg.	65.79	64.68	63.14	65.22	64.48	63.79	**67.06**
	Δ	/	**↓1.11**	**↓2.65**	**↓0.57**	**↓1.31**	**↓2.00**	**↑1.27**

Setting. We use LLaVA-OV-1.5-4B with Qwen3-4B-Instruct-2507 as the text teacher and $K = 256$ (Table 4). VP-OPD uses $\rho = 0.375$ and multimodal data only for calibration. Multimodal training examples are drawn from the public LLaVA-OneVision-1.5 dataset (An et al., 2025b), with Visual-KL and MOPD preserving the original type proportions and sampling uniformly within each type. Replay-SFT, Visual-KL, and MOPD use a 3:1 text-to-multimodal ratio.

- **OPD + Replay-SFT.** Adds multimodal cross-entropy to full-parameter OPD with $\lambda = 0.25$ and batch size 512.
- **EWC-style OPD.** Adds a Fisher-weighted parameter penalty with $\lambda_{\text{EWC}} = 10^4$, estimating Fisher weights on the same multimodal calibration set as VP-OPD.
- **OPD + Visual-KL.** Adds reverse KL from the current VLM to the frozen initial VLM on multimodal inputs, with $\beta = 0.8$ and batch size 512.
- **MOPD.** Distills text-only student rollouts from the LLM teacher and multimodal student rollouts from Qwen3-VL-8B-Instruct.

Results and Analysis. VP-OPD better preserves vision performance than EWC-style OPD, OPD + Replay-SFT, and OPD + Visual-KL while retaining most of full-parameter OPD's language gains (Table 6). OPD + Replay-SFT improves mathematics but causes a larger vision drop than full-parameter OPD, whereas EWC-style OPD and OPD + Visual-KL only partially recover performance on $\text{MMMU-Pro}_{\text{vis}}$ and reduce average text-centric performance. MOPD does not achieve a balanced text–vision tradeoff. Compared with full-parameter OPD, it improves knowledge benchmarks and recovers performance on $\text{MMMU-Pro}_{\text{vis}}$ but degrades mathematics and InfoVQA, leaving VP-OPD with a better overall text–vision tradeoff.

G Additional Experiments

G.1 Asymmetric Drift Sensitivity to the Trainable Ratio

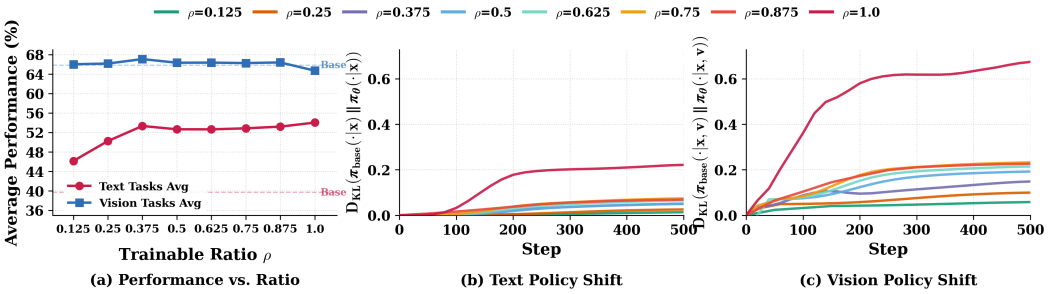


Figure 4: **Analysis of VP-OPD on LLaVA-OneVision-1.5-4B. (a)** Average text-centric and vision-centric accuracy as the trainable ratio ρ varies from 0.125 to 1.0; dashed lines mark the base model. **(b, c)** Forward KL between the base and current policy over training for each ρ.

The trainable ratio ρ governs the text–vision tradeoff in VP-OPD, and the three panels of Fig. 4 jointly characterize its shape. The finest attainable ρ is set by the coarser of the two structural units, the KV-group: since each layer contributes N_l^{attn} KV-groups to the ranking in Eq. (9), the per-layer attention mask admits at most N_l^{attn} distinct trainable fractions. Both LLaVA-OneVision-1.5-4B and LLaVA-OneVision-1.5-8B have $N_l^{\text{attn}}{=}8$ KV-groups per layer under GQA, which fixes the finest attainable ρ at $1/8$, the lower end of our sweep. In Fig. 4(a), text-centric accuracy rises sharply from $\rho{=}0.125$ to $\rho{=}0.375$ and largely saturates thereafter, while vision-centric accuracy peaks at $\rho{=}0.375$ (above the base), remains near the base for $\rho \in [0.5, 0.875]$, and collapses under full-parameter OPD ($\rho{=}1.0$). The two trends together favor $\rho{=}0.375$ as the joint optimum, which we adopt as the VP-OPD operating point. To visualize how policy shift evolves, Fig. 4(b, c) report the forward KL of the text-mode and vision-conditioned policies against the base policy over training steps. Once the vision-critical subnetwork is frozen, the vision-conditioned policy is tightly bounded. Even at $\rho = 0.875$, its KL against the base policy stays below 0.3, in sharp contrast to the uncontrolled growth under full-parameter OPD ($\rho = 1.0$). The text-mode policy KL is similarly suppressed at the same ρ, suggesting a partial overlap between the vision-critical and text-critical subnetworks, which we analyze and visualize in Appendix G.2.

G.2 Cross-Modal Importance Overlap

To examine the overlap at the component level, we compute the first-order Taylor importance scores in Eq. (5) on a text-only calibration set and on the multimodal calibration set used in VP-OPD, and compare the per-component rankings across all 36 layers.

Figure 5 reports the comparison at two structural granularities. MLP channels show high correlation between the two rankings (r=0.79), while attention KV-groups show substantially lower correlation (r=0.39). The gap is consistent with attention heads carrying more modality-specific importance than MLP channels, so text-only importance is a closer proxy for MLP than for attention. This supports using multimodal calibration in VP-OPD when the goal is to identify vision-critical structures.

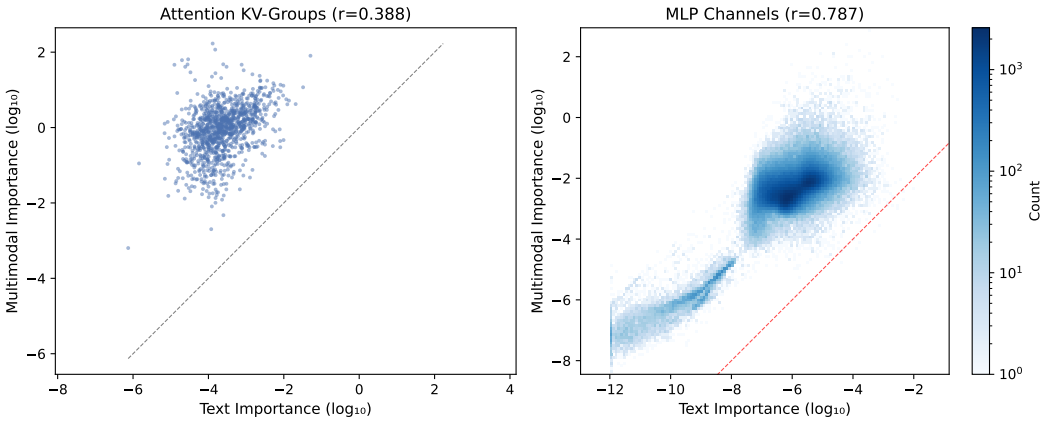
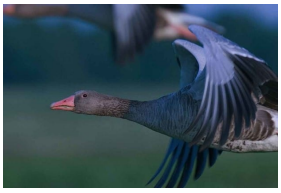


Figure 5: **Cross-modal importance correlation.** Comparison of per-component importance scores computed from text-only vs. multimodal calibration data (LLaVA-OneVision-1.5-8B, all 36 layers). *Left:* Attention KV-groups show moderate correlation (r=0.39). *Right:* MLP channels show high correlation (r=0.79). The diagonal indicates perfect agreement.

H CALIBRATION DATA EXAMPLES

VP-OPD estimates vision importance on a small multimodal calibration set sampled from the LLaVA-OneVision-1.5 mid-training data (An et al., 2025a). Each sample consists of an image paired with a fixed prompt and a descriptive caption used as the generation target. Table 7 shows three representative samples, illustrating the diversity of visual content. These samples are drawn uniformly at random from the 85M mid-training split; as shown in Section 4.3, the resulting importance ranking is robust to the specific calibration subset.

Table 7: **Representative calibration samples.** Three examples from the 500-sample calibration set used for vision importance estimation. Each sample pairs an image with the prompt `"Describe this image in detail."` and a descriptive caption as the generation target for computing gradient-based saliency (Eq. (5)).

Image	Caption (Generation Target)
	A gray goose with a pink beak and black-tipped wings is captured mid-flight in the foreground, its body streamlined and wings outstretched, showing detailed feathers. Another goose is blurred slightly behind it, suggesting a flock in motion. The background is a soft, out-of-focus mix of green and blue, likely a grassy field and sky, indicating an outdoor setting. The lighting is natural, with soft sunlight highlighting the goose's feathers.
	The image depicts Edradour Distillery, a historic Scottish whisky distillery known as one of the smallest traditional distilleries in Scotland. The main building features a white stone exterior with a black roof, accented by red window frames and doors. A large wooden sign reading "EDRADOUR DISTILLERY" spans the upper part of the wall. In front of the building, a blue trailer filled with brown grain is positioned near an open red door.
	The image shows a close-up of hands holding a smartphone with a trading app displaying green and red candlestick charts, positioned above a laptop keyboard and papers with financial graphs and the text "Should Learn MQL4 or MQL5." The scene is indoors, likely a workspace, with neutral tones creating a professional, focused mood. Sunlight streams from the left, casting soft shadows on the papers and laptop.

I BROADER IMPACT

VP-OPD targets a practical issue in vision-language models: the language degradation induced by multimodal fine-tuning. By restricting text-only distillation updates to parameters less critical for vision-conditioned behavior, it recovers most of the language gains of full-parameter OPD while substantially preserving vision performance, and does so with only a fraction of trainable parameters. This offers a cheaper and more accessible path to maintaining balanced capabilities in VLMs, which can benefit downstream applications such as education, assistive tools, and scientific analysis. As a training-time method, VP-OPD does not introduce new deployment-time capabilities; however, it inherits risks common to VLMs, including biases from pretraining data, hallucinations, and potential misuse in sensitive settings. We encourage responsible use and discourage any applications that may infringe upon personal privacy or be deployed for harmful purposes.

All base models, datasets, and benchmarks used in this work are publicly released for academic research, and we use them strictly under their original licenses. Our calibration set and rollouts are derived solely from publicly available samples, and all experimental results are generated for research purposes and follow standard ethical guidelines for the use of public data.