SEEING BEFORE DISTILLING: PREFILL VISUAL ALIGNMENT FOR MULTIMODAL ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) mitigates the distribution mismatch of offline distillation by training students on their own generated trajectories. However, we find that vanilla OPD is often insufficient for vision-language models, especially on visually dominated tasks, where students may attend to irrelevant image regions and weaken subsequent token-level distillation. To address this issue, we propose Prefill Visual Alignment On-Policy Distillation, which aligns the student's visual-token attention with the teacher during the prefill stage. This avoids task-specific decoded-token selection and provides a simple, unified supervision signal across multimodal tasks. Experiments on InternVL3 and Qwen3-VL show that our method improves on most benchmarks over vanilla OPD, e.g., from 44.8 to 51.5 on InfoVQA and from 69.1 to 71.8 on OCRBench. These results demonstrate that prefill-stage visual alignment strengthens visual grounding and enables more effective multimodal knowledge transfer.

1 INTRODUCTION

Large language models and vision-language models show strong reasoning and instruction-following abilities, but their deployment is constrained by high inference cost, memory usage, and slow decoding. Knowledge distillation Hinton et al. (2015) transfers knowledge from a powerful teacher to a smaller student, yet conventional offline distillation often relies on teacher-generated or external trajectories, causing a mismatch with the student's own inference-time distribution. This issue is severe in autoregressive generation Brown et al. (2020), where early errors can accumulate and push the student into unseen states. On-policy distillation Agarwal et al. (2024) mitigates this by training on the student's generated trajectories while using the teacher to supervise these student-induced states, making distillation better aligned with the student's actual behavior.

However, we find that on-policy distillation does not fully unlock the potential of multimodal students. This limitation becomes particularly evident in visually dominated tasks, such as chart understanding, document reasoning, OCR, and image captioning, where correct generation depends heavily on fine-grained visual grounding rather than only token-level distribution matching. In some cases, vanilla OPD even degrades performance; for example, on captioning tasks, InternVL3 Zhu et al. (2025) drops from 46.8 to 45.2, while Qwen3-VL Bai et al. (2025) decreases from 45.3 to 43.7. These results suggest that simply supervising the student on its own generated trajectories is insufficient for improving multimodal abilities. Through further visualization of attention maps, we observe that the student often allocates attention to irrelevant visual regions as shown in Figure 1, which reduces the signal-to-noise ratio of task-relevant visual evidence and weakens the effectiveness of subsequent token-level distillation.

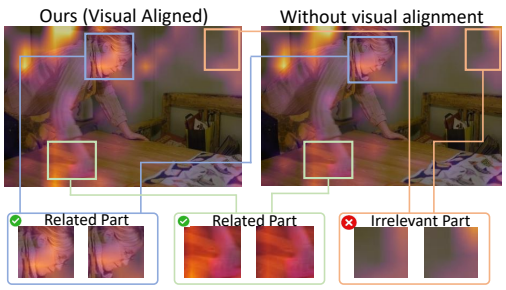

Figure 1: **Visual alignment improves token-level visual caption.** For the attention map of the word *"woman"*, our visual-aligned model focuses on semantically related regions such as the face and body, while the model without visual alignment attends to irrelevant background areas.

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.

To address this issue, we propose Prefill Visual Alignment On-Policy Distillation, which explicitly aligns the student's visual attention behavior before autoregressive decoding. Instead of identifying task-specific or semantically important decoded tokens, which would introduce additional annotation, parsing, or token-selection overhead, our method performs alignment directly over the visual-token attention during the prefill stage. This design is simple and unified across different multimodal tasks, since every sample naturally contains visual tokens before response generation. By guiding the student to attend to image regions more consistently with the teacher, our method strengthens task-relevant visual grounding at the source, thereby providing a cleaner visual context for subsequent on-policy token-level distillation.

The experimental results show that Visual-Align OPD improves over vanilla OPD across different model families and benchmarks. On InternVL3, our method brings clear gains on visually intensive tasks; for example, the performance on InfoVQA increases from 44.8 with vanilla OPD to 51.5 with Visual-Align OPD, indicating that explicit visual-token alignment helps the student better capture fine-grained document evidence. Similar improvements are observed on Qwen3-VL, where our method improves OCRBench from 69.1 to 71.8, showing stronger OCR-oriented visual grounding. These gains are not limited to a single benchmark or architecture, but appear across both InternVL3 and Qwen3-VL, suggesting that aligning visual attention during the prefill stage provides a more effective and robust supervision signal than token-level OPD alone. Overall, the results demonstrate that visual alignment can compensate for the visual grounding weakness of vanilla OPD and better transfer the teacher's multimodal ability to compact student models.

2 Related Work

Multimodal Large Language Models. Multimodal large language models (MLLMs) integrate visual encoders with large language models to enable image-conditioned reasoning and generation. Representative methods such as LLaVA Liu et al. (2023), InstructBLIP Dai et al. (2023), Qwen-VL Bai et al. (2025), and InternVL Zhu et al. (2025) have shown that visual features can be effectively connected to language models through projection layers, cross-attention modules, and multimodal instruction tuning. These models achieve strong performance on a wide range of vision-language tasks, including visual question answering Singh et al. (2019), captioning Plummer et al. (2015), OCR Liu et al. (2024), document understanding Zou et al. (2025), and chart reasoning Xu et al. (2023). However, multimodal generation differs from text-only generation because the decoding process is conditioned on an earlier visual prefill stage, where image tokens are integrated into the language model context. Errors in this stage, such as attending to irrelevant regions or missing fine-grained visual evidence, can bias the entire subsequent reasoning trajectory. While prior MLLM Singh et al. (2025); Achiam et al. (2023) studies mainly focus on architecture design, data scaling, and instruction tuning, our work studies how to transfer the teacher model's visual grounding behavior to the student during distillation. By aligning visual attention over image tokens, our method complements token-level on-policy distillation with prefill visual alignment.

Knowledge Distillation. Knowledge distillation Hinton et al. (2015) transfers teacher capabilities to smaller students by matching predictive distributions or internal representations. Classical approaches focus on logit-level supervision, while later methods further align hidden states or attention maps Zagoruyko & Komodakis (2016) to preserve richer intermediate knowledge. For large language models Gu et al. (2024), on-policy distillation Agarwal et al. (2024); Sang et al. (2026) has recently emerged as an effective alternative to offline distillation by letting the student generate its own responses and using the teacher to supervise the resulting trajectories Li et al. (2026), reducing the discrepancy between training and inference. Nevertheless, these methods are mostly developed for text-only models and mainly address errors during decoding. In vision-language models, visual prefill grounds prompts in image tokens; irrelevant attention or missing evidence can bias subsequent decoding Yang et al. (2024a). We therefore augment on-policy distillation with visual alignment to match the teacher's attention before and during generation.

3 METHOD

3.1 PRELIMINARIES: ON-POLICY DISTILLATION

Knowledge distillation provides a simple yet effective paradigm for transferring the capabilities of a large teacher model to a compact student model. Given an input x, the student is trained to match the teacher's predictive distribution rather than only the one-hot ground-truth label:

$$\mathcal{L}_{\mathrm{KD}} = \mathbb{E}_{x\sim\mathcal{D}} D_{\mathrm{KL}}\left(p_T(\cdot \mid x) \| p_\theta(\cdot \mid x)\right). \quad (1)$$

By preserving the soft probabilities over candidate outputs, distillation conveys richer information about the teacher's uncertainty, preference, and decision boundary, enabling the student to inherit stronger generalization ability while maintaining lower inference cost. On-policy distillation aims to transfer the behavior of a strong teacher model to a smaller student under the student's own induced generation distribution. Given an input $x \sim \mathcal{D}$, the student first samples an on-policy trajectory $y = (y_1, \ldots, y_T) \sim \pi_\theta(\cdot \mid x)$, where each prefix $y_{<t}$ represents a state actually visited by the student during decoding. The teacher then provides a target distribution $p_T(\cdot \mid x, y_{<t})$ at each visited state, and the student is optimized to match this distribution:

$$\mathcal{L}_{\mathrm{OPD}} = \mathbb{E}_{x,\, y\sim\pi_\theta} \sum_t D_{\mathrm{KL}}\left(p_T^t \,\|\, p_\theta^t\right), \quad (2)$$

where $p_T^t = p_T(\cdot \mid x, y_{<t})$ and $p_\theta^t = p_\theta(\cdot \mid x, y_{<t})$. Different from standard offline distillation, which supervises the student only on fixed reference trajectories y^*, on-policy distillation exposes the teacher to the student's own intermediate predictions and therefore provides corrective supervision on the states where the student is most likely to make mistakes. This makes the distillation objective better aligned with the test-time autoregressive process, reducing the discrepancy between training-time supervision and inference-time behavior.

Algorithm 1 Visual-Align On-Policy Distillation

Require: Student model π_θ; teacher model π_T; training dataset $\mathcal{D}$; selected layer set L; layer mapping $m(\cdot)$; visual-alignment weight λ_v; learning rate η.

for $x = (I, q) \sim \mathcal{D}$ **do**
  Sample $y \sim \pi_\theta(\cdot \mid x)$;
  $(\mathbf{z}_\theta^t, \{A_\theta^l\}) \leftarrow \pi_\theta(x, y_{<t})$;
  $(\mathbf{z}_T^t, \{A_T^l\}) \leftarrow \mathrm{sg}(\pi_T(x, y_{<t}))$;
  $\mathcal{V} \leftarrow \{\text{visual-token positions}\}$;
  $p_\theta^t \leftarrow \mathrm{softmax}(\mathbf{z}_\theta^t)$;
  $p_T^t \leftarrow \mathrm{softmax}(\mathbf{z}_T^t)$;
  $\mathcal{L}_{\mathrm{OPD}} \leftarrow D_{\mathrm{KL}}\left(p_T^t \,\|\, p_\theta^t\right)$;
  for $l \in L$ **do**
    $A_\theta^l \leftarrow \mathrm{Norm}_{\mathcal{V}}\left(\mathrm{MeanHead}(A_\theta^l)\right)$;
    $A_T^{m(l)} \leftarrow \mathrm{Norm}_{\mathcal{V}}\left(\mathrm{MeanHead}(A_T^{m(l)})\right)$;
    $\ell_{\mathrm{VA}}^l \leftarrow D_{\mathrm{KL}}\left(\mathrm{sg}(A_T^{m(l)}) \,\|\, A_\theta^l\right)$;
  end for
  $\mathcal{L}_{\mathrm{VA}} \leftarrow \frac{1}{|L|}\sum_{l\in L} \ell_{\mathrm{VA}}^l$;
  $\mathcal{L}_{\text{Visual-OPD}} \leftarrow \mathcal{L}_{\mathrm{OPD}} + \lambda_v \mathcal{L}_{\mathrm{VA}}$;
  $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{Visual-OPD}}$;
end for

3.2 PREFILL VISUAL-ALIGN ON-POLICY DISTILLATION

Problem Formulation. Given a multimodal input $x = (I, q)$, where I is an image and q is a textual question or instruction, a vision-language model first encodes the image into visual tokens and then performs autoregressive decoding conditioned on both visual and textual contexts. The student model π_θ generates an on-policy response as

$$y \sim \pi_\theta(\cdot \mid x), \quad (3)$$

where $y = (y_1, \ldots, y_T)$ denotes the response sampled from the current student policy, and T is the response length. At decoding step t, standard on-policy distillation matches the teacher and student predictive distributions under the same student-generated prefix:

$$\ell_{\mathrm{OPD}}^t = D_{\mathrm{KL}}\left(p_T^t \,\|\, p_\theta^t\right), \quad (4)$$

where $p_T^t = p_T(\cdot \mid x, y_{<t})$ and $p_\theta^t = p_\theta(\cdot \mid x, y_{<t})$ denote the teacher and student token distributions, respectively. The overall OPD objective is then defined as

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \mathbb{E}_{x,y}\left[\sum_{t=1}^{T} \ell_{\mathrm{OPD}}^t\right], \quad (5)$$

where $x \sim \mathcal{D}$, $y \sim \pi_\theta(\cdot \mid x)$, and $\mathcal{D}$ denotes the training data distribution.

This objective provides token-level supervision along the student's own generation trajectory, but is less direct for vision-language distillation. If the student selects wrong visual evidence during prefill,

such as wrong objects, background regions, or missed OCR areas, the token-level KL loss can only correct these grounding errors indirectly. To address this, we introduce Visual-Align OPD, which augments on-policy distillation with explicit visual attention alignment, encouraging the student to select teacher-consistent visual evidence under the same on-policy prefix.

Visual Attention Map. Let the self-attention matrix of the student model at layer l and head h be

$$W_\theta^{l,h} \in \mathbb{R}^{N\times N}, \tag{6}$$

where N is the token length of the current multimodal prefix, l indexes decoder layers, and h indexes attention heads. We denote the set of visual token positions by $\mathcal{V}$ and the set of query positions used for alignment by $\mathcal{Q}$. For a query token $i \in \mathcal{Q}$, the student's attention over visual tokens is extracted as

$$u_{\theta,i}^{l,h} = W_\theta^{l,h}[i, \mathcal{V}], \tag{7}$$

where $u_{\theta,i}^{l,h}$ is the unnormalized attention vector from query token i to all visual key tokens in $\mathcal{V}$. We normalize this vector into a probability distribution over visual tokens:

$$a_{\theta,i}^{l,h} = \mathrm{Norm}\left(u_{\theta,i}^{l,h}\right), \tag{8}$$

where $a_{\theta,i}^{l,h}$ denotes the normalized visual attention distribution for layer l, head h, and query token i. The normalization operator is defined as

$$\mathrm{Norm}(u)_j = \frac{u_j}{\sum_{k\in\mathcal{V}} u_k + \epsilon}, \tag{9}$$

where $j \in \mathcal{V}$ indexes a visual token, and ϵ is a small constant for numerical stability. To obtain a stable layer-wise visual attention map, we average over selected heads and query positions:

$$A_\theta^l = \frac{1}{|\mathcal{H}||\mathcal{Q}|} \sum_{h\in\mathcal{H}} \sum_{i\in\mathcal{Q}} a_{\theta,i}^{l,h}, \tag{10}$$

where $\mathcal{H}$ is the set of selected attention heads, and A_θ^l is the student's visual attention map at layer l. Similarly, the teacher visual attention map is computed as

$$A_T^l = \frac{1}{|\mathcal{H}||\mathcal{Q}|} \sum_{h\in\mathcal{H}} \sum_{i\in\mathcal{Q}} a_{T,i}^{l,h}, \tag{11}$$

where $a_{T,i}^{l,h}$ and A_T^l denote the corresponding teacher-side visual attention distribution and layer-wise attention map. When the teacher and student have different depths, we define a layer mapping:

$$m(l_s) = l_T, \tag{12}$$

where l_s is a selected student layer, l_T is its corresponding teacher layer, and $m(\cdot)$ specifies the teacher-student layer correspondence.

Visual Alignment Objective. As shown in Algorithm 1, for each selected student layer l, we align the student's visual attention map with the mapped teacher attention map:

$$\ell_{\mathrm{VA}}^l = D_{\mathrm{KL}}\left(\mathrm{sg}(A_T^{m(l)}) \,\|\, A_\theta^l\right), \tag{13}$$

where $\mathrm{sg}(\cdot)$ denotes stop-gradient, ensuring that the teacher serves only as a fixed supervisory signal. The KL divergence over visual tokens is computed as

$$D_{\mathrm{KL}}(A_T \,\|\, A_\theta) = \sum_{j\in\mathcal{V}} A_{T,j} \log \frac{A_{T,j} + \epsilon}{A_{\theta,j} + \epsilon}, \tag{14}$$

where $A_{T,j}$ and $A_{\theta,j}$ are the teacher and student attention probabilities assigned to visual token j.

The full visual alignment loss is

$$\mathcal{L}_{\mathrm{VA}}(\theta) = \mathbb{E}_{x,y}\left[\frac{1}{|\mathcal{L}|} \sum_{l\in\mathcal{L}} \ell_{\mathrm{VA}}^l\right], \tag{15}$$

where $\mathcal{L}$ is the set of selected student layers, $x \sim \mathcal{D}$, and $y \sim \pi_\theta(\cdot \mid x)$.

This objective encourages the student to allocate attention to the visual evidence emphasized by the teacher. Unlike token-level OPD, which supervises the final predictive distribution, the visual alignment term directly regularizes the intermediate visual grounding process.

Unified Objective. The final Visual-Align OPD objective combines token-level on-policy distillation with visual attention alignment:

$$\mathcal{L}_{\text{Visual-OPD}}(\theta) = \mathcal{L}_{\text{OPD}}(\theta) + \lambda_v \mathcal{L}_{\text{VA}}(\theta), \tag{16}$$

where λ_v controls the strength of visual attention alignment. When temperature scaling is used for token-level distillation, the token distribution is defined as

$$p_\theta^{t,\tau} = \text{softmax}\left(z_\theta^t/\tau\right), \tag{17}$$

where z_θ^t denotes the student logits at step t, and τ is the distillation temperature. The corresponding teacher distribution is

$$p_T^{t,\tau} = \text{softmax}\left(z_T^t/\tau\right), \tag{18}$$

where z_T^t denotes the teacher logits at step t. The temperature-scaled OPD loss becomes

$$\ell_{\text{OPD}}^{t,\tau} = \tau^2 D_{\text{KL}}\left(p_T^{t,\tau} \,\|\, p_\theta^{t,\tau}\right), \tag{19}$$

where the factor τ^2 follows the standard distillation scaling convention.

3.3 Why Visual Alignment Helps OPD

From a motivating analysis, the discrepancy between the teacher and student token distributions does not only arise from the language decoder itself, but also from how the two models aggregate visual evidence during the visual prefill stage. In a vision-language decoder, the prediction at each token is conditioned on an implicit visual context constructed through attention over visual tokens. Therefore, if the student attends to different visual regions from the teacher, the subsequent on-policy distillation loss may operate on a misaligned visual evidence path.

Formally, let $\mathcal{V}$ denote the set of visual token positions. At layer l, we write the visual context vector of the student as

$$c_\theta^l = \sum_{j \in \mathcal{V}} A_{\theta,j}^l v_{\theta,j}^l, \tag{20}$$

where $A_{\theta,j}^l$ denotes the attention weight assigned to the j-th visual token, and $v_{\theta,j}^l$ is its corresponding value representation. To isolate the effect of attention routing, we consider the case where the teacher and student share comparable visual value representations, denoted by v_j^l. Then the visual context mismatch can be approximated as

$$\left\|c_T^l - c_\theta^l\right\| \approx \left\| \sum_{j \in \mathcal{V}} \left(A_{T,j}^l - A_{\theta,j}^l\right) v_j^l \right\|. \tag{21}$$

If the visual value representations are bounded, i.e., $\|v_j^l\| \le C$ for all $j \in \mathcal{V}$, the above mismatch can be upper bounded by the attention discrepancy:

$$\left\|c_T^l - c_\theta^l\right\| \le C \left\|A_T^l - A_\theta^l\right\|_1. \tag{22}$$

Furthermore, by Pinsker's inequality, we have

$$\left\|c_T^l - c_\theta^l\right\| \le C\sqrt{2 D_{\text{KL}}\left(A_T^l \,\|\, A_\theta^l\right)}. \tag{23}$$

This bound shows that minimizing the visual attention KL term can indirectly control the mismatch between the teacher and student visual contexts. Therefore, Visual Alignment is not merely an auxiliary distillation regularizer. Instead, it explicitly constrains the visual evidence pathway used by the student during on-policy distillation. In this sense, the student is encouraged not only to imitate what the teacher predicts, but also to align with where the teacher grounds its prediction.

4 Experiments

4.1 Experimental Setup

Models and baselines. We conduct experiments on two representative vision-language student models, including InternVL3-2B Zhu et al. (2025) and Qwen3-VL-2B Bai et al. (2025). For on-policy distillation, each student model is distilled from a larger teacher model within the same model family: InternVL3-2B uses InternVL3-8B as the teacher, and Qwen3-VL-2B uses Qwen3-VL-8B as the teacher. We compare three methods: the original base model, standard on-policy distillation (OPD), and our proposed Visual-Align OPD.

Table 1: **Main results on seven multimodal benchmarks (test).** Bold marks the best score within each model family; shading highlights Visual-Align OPD (ours). All test portions are held out from distillation training.

Benchmark	Metric	InternVL3-2B			Qwen3-VL-2B		
		Base	OPD	**Ours**	Base	OPD	**Ours**
ChartQA (test)	Acc.	74.6	71.7	**76.7**	76.8	70.2	**79.2**
OCRBench (test)	EM	68.5	71.5	**71.8**	69.9	69.1	**71.8**
InfoVQA (test)	EM	39.0	44.8	**51.5**	64.1	54.9	**64.7**
DocVQA-2026 (test)	EM	3.8	**5.0**	**5.0**	5.0	2.5	**6.3**
TextVQA (test)	Acc.	59.1	**76.8**	66.3	78.6	78.4	**79.3**
Flickr30k (test)	METEOR	46.83	45.23	**48.20**	45.32	43.68	**45.73**
VizWiz-Caps (test)	METEOR	23.01	21.24	**23.35**	23.67	24.09	**24.27**

Benchmarks and data splits. We train on 5,000 examples sampled without replacement from InfoVQA (train), VizWiz-VQA (train) (Gurari et al., 2018), ChartQA (train), TextVQA (train), Flickr30k (train), and VizWiz-Caps (train). Evaluation uses the seven test benchmarks in Table 1; OCRBench and DocVQA-2026 supply no training examples. Training and evaluation may share a dataset name, but use disjoint portions: no test examples are included in the distillation training set. Appendix A details the training mixture and benchmark sources.

Implementation details. All models are trained using the ms-swift framework.[1] The training experiments are conducted on 4 NVIDIA A100 GPUs. For inference evaluation, we use an NVIDIA RTX 4090 under the same environment.

4.2 Main Results

Table 1 compares the base student, standard OPD, and Visual-Align OPD on held-out test data. We report accuracy on ChartQA (Masry et al., 2022) and TextVQA (Singh et al., 2019), exact match on OCRBench (Liu et al., 2024), InfoVQA (Mathew et al., 2022), and DocVQA-2026 (DocVQA 2026 Organizing Team, 2026), and METEOR on Flickr30k (Young et al., 2014) and VizWiz-Caps (Gurari et al., 2020). These benchmarks cover chart reasoning, OCR, document and text-based VQA, and image captioning.

As shown in Table 1, our method achieves stronger overall performance than both the base model and standard OPD. On InternVL3-2B, visual-align OPD improves ChartQA from 74.6 to 76.7, OCRBench from 68.5 to 71.8, InfoVQA from 39.0 to 51.5, and Flickr30k from 46.83 to 48.20. On Qwen3-VL-2B, our method improves on most benchmarks, increasing ChartQA from 76.8 to 79.2, OCRBench from 69.9 to 71.8, DocVQA-2026 from 5.0 to 6.3, TextVQA from 78.6 to 79.3, and VizWiz-Caps from 23.67 to 24.27. Visual-Align OPD improves overall robustness, but its task-agnostic grounding regularization may trade off with vanilla OPD's token-level fitting on short OCR-style answers such as TextVQA. In contrast, standard OPD can be unstable and even degrades performance on several benchmarks, such as ChartQA and captioning tasks. These results indicate that aligning visual attention during OPD helps the student model better preserve teacher-consistent visual grounding, leading to more reliable multimodal distillation.

Relative to vanilla OPD, visual alignment improves ChartQA by 5.0 points on InternVL3-2B and 9.0 points on Qwen3-VL-2B. However, InternVL3-2B reaches 66.3 on TextVQA, exceeding the base model's 59.1 but falling below OPD's 76.8. Stronger visual grounding therefore complements token-level supervision without dominating it on every task.

4.3 Analysis Study

Attention-Map Alignment Analysis. To examine whether visual alignment changes the internal grounding behavior of the student model trained by our method, we compare the generated-token

[1] Anonymized code is available at `https://anonymous.4open.science/r/851dfb8b922cf53b7296d7ce`.



Figure 2: **Visual alignment improves attention-map similarity.** Lines connect each sample before and after alignment; mean similarity increases on Flickr30k, TextVQA, and ChartQA.

Table 2: **Cosine similarity of generated-token visual attention maps.** We compare the attention-map closeness to the student and teacher models, where higher similarity indicates closer visual attention patterns.

Benchmark	vs Student		vs Teacher	
	w/o VA	Ours	*w/o VA*	Ours
Flickr30k	0.90	**0.95**	0.69	**0.70**
TextVQA	0.90	**0.95**	0.35	**0.36**
ChartQA	0.93	**0.99**	0.55	**0.60**

visual attention maps before and after applying our visual-alignment objective. For each sample, we average the attention maps over generated tokens and compute the cosine similarity between the student attention map and the corresponding teacher attention map. The same sample is connected by a line in Figure 2, allowing us to directly observe how visual alignment shifts the student toward or away from the teacher on instance-by-instance basis.

As shown in Figure 2, visual alignment increases the average student and teacher attention-map similarity across Flickr30k, TextVQA, and ChartQA. The improvement is especially clear on TextVQA and ChartQA, where accurate answering requires fine-grained visual grounding over text and structured visual evidence. Although a small number of samples show decreased similarity, the overall upward shift of the mean bars indicates that our method makes the student attend to visual regions more similarly to the teacher. This suggests that the gains of our approach do not only come from token-level distribution matching, but also from correcting the student's visual grounding behavior during generation.

Visual attention score analysis. To analyze whether visual alignment changes the model's internal visual grounding behavior, we compare the cosine similarity of mean generated-token visual attention maps. Specifically, for each generated response, we average the attention maps over generated tokens and measure their similarity to the attention maps produced by the original student model and the teacher model. This comparison allows us to examine whether our visual-alignment OPD shifts the student's visual attention toward more reliable visual grounding patterns, rather than only improving the final output scores.

As shown in Table 2, our method increases the attention-map similarity to both the student and teacher models across all evaluated benchmarks. Compared with w/o VA, the similarity to the student model improves from 0.90 to 0.95 on Flickr30k, from 0.90 to 0.95 on TextVQA, and from 0.93 to 0.99 on ChartQA. Meanwhile, the similarity to the teacher model also increases from 0.69 to 0.70, 0.35 to 0.36, and 0.55 to 0.60, respectively. Surprisingly, visual alignment does not simply pull the student away from its original attention pattern toward the teacher; instead, it improves closeness to both sides. This suggests that our method regularizes the visual routing process by suppressing noisy or unstable attention while preserving the student's useful grounding structure. Such a phenomenon indicates that visual alignment may provide a more stable prefill-stage visual basis for OPD, allowing token-level distillation to operate on a better grounded generation trajectory. The teacher-similarity gains are modest in absolute magnitude: 0.01 on Flickr30k and TextVQA, and 0.05 on ChartQA. Meanwhile,

Table 3: **Ablation on the visual-alignment weight λ_v.** We report OCRBench and InfoVQA accuracy/EM, and METEOR scores on Flickr30k and VizWiz-Caps.

Method	λ_v	OCRBench Acc.	InfoVQA Acc.	Flickr30k METEOR	VizWiz-Caps METEOR
Base model	–	68.5	39.0	46.83	23.01
OPD baseline	0	71.5	44.8	45.23	21.24
Visual-Align OPD	0.1	71.8	**51.5**	48.20	23.35
Visual-Align OPD	0.5	**72.8**	**51.5**	**48.39**	23.33
Visual-Align OPD	1.0	72.2	50.6	48.21	**23.50**
Visual-Align OPD	2.0	71.8	50.7	48.21	23.44

similarity to the original student increases by 0.05, 0.05, and 0.06, respectively. These measurements are consistent with preserving useful student attention while correcting some teacher-discrepant patterns, rather than replacing the student's attention wholesale. They also caution against treating attention similarity as a substitute for task accuracy: a higher average similarity need not improve every prediction, and its value depends on which visual evidence the task requires.

Effect of visual-alignment weight. Table 3 studies the effect of the visual-alignment weight λ_v in Visual-Align OPD. Setting $\lambda_v = 0$ corresponds to the OPD baseline, where the student is distilled only on its on-policy token trajectory without explicit visual-attention alignment. We vary λ_v from 0.1 to 2.0 to examine whether the proposed visual alignment is sensitive to the strength of the auxiliary alignment objective.

The results show that Visual-Align OPD is better than the original OPD baseline across a wide range of λ_v values, demonstrating the robustness of the proposed alignment objective. For OCRBench, InfoVQA, Flickr30k, and VizWiz-Caps, all tested visual-alignment settings outperform the OPD baseline, indicating that the gain does not rely on a narrowly tuned hyperparameter. In particular, $\lambda_v = 0.5$ achieves the best overall performance, improving OCRBench from 71.5 to 72.8, InfoVQA from 44.8 to 51.5, and Flickr30k from 45.23 to 48.39. Even when λ_v is increased to 1.0 or 2.0, the model still maintains clear advantages over OPD. This suggests that visual alignment provides a stable complementary signal to on-policy distillation, helping the student improve visual grounding without being overly sensitive to the choice of λ_v.

Effect of the number of visual-alignment layers. Table 4 studies the effect of using different numbers of layers for visual alignment. We fix the visual-alignment weight to $\lambda_v = 0.5$ and vary the number of aligned decoder layers. Specifically, #Layers $= 1$, 3, and 6 denote applying the alignment loss to the last 1, 3, and 6 decoder layers, respectively. This experiment examines whether this method is sensitive to the scope of layer-wise visual alignment.

Overall, Visual-Align OPD remains effective across all tested layer numbers. All three settings outperform vanilla OPD on InfoVQA, Flickr30k, and VizWiz-Caps, indicating benefits across multiple layer configurations. Among them, aligning only the last layer achieves the best results on OCRBench (72.8) and InfoVQA (51.5), while aligning 6 layers obtains the best Flickr30k score (48.45), and aligning 3 layers performs best on VizWiz-Caps (23.58). These results suggest that upper-layer visual alignment is generally beneficial, but increasing the number of aligned layers does not lead to monotonic gains. A compact alignment on the final layers is often sufficient, likely because later decoder layers contain more task-relevant and semantically refined visual grounding information, consistent with prior findings in feature-based ViT distillation Yang et al. (2024b). On OCRBench, six layers yield 71.2, below OPD's 71.5, while one layer yields 72.8. This trade-off suggests selecting alignment depth for the task rather than assuming that more constraints always improve recognition.

PPL analysis on human captions. We evaluate caption likelihood by feeding the first sentence of each ground-truth human caption to each model and computing its log perplexity. This measures compatibility with human annotations rather than generated-caption quality; lower log-PPL indicates

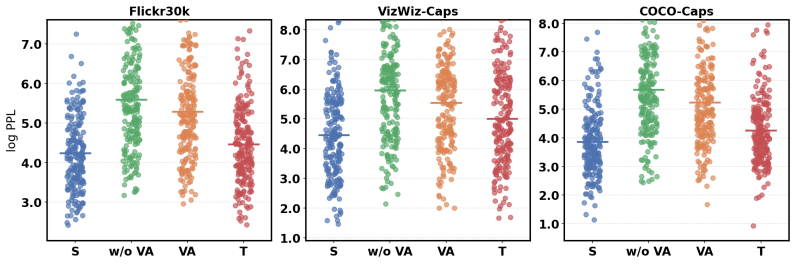

Figure 3: **Log PPL on human captions.** Points represent samples and bars denote means. Visual alignment preserves caption likelihood on Flickr30k, VizWiz-Caps, and COCO-Caps.

Table 4: **Ablation on the number of visual-alignment layers.** We set $\lambda_v = 0.5$ and align the last 1, 3, or 6 decoder layers. Metrics follow Table 3.

Method	#Layers	OCRBench Acc.	InfoVQA Acc.	Flickr30k METEOR	VizWiz-Caps METEOR
Base model	–	68.5	39.0	46.83	23.01
Vanilla OPD	–	71.5	44.8	45.23	21.24
Visual-Align OPD	1	**72.8**	**51.5**	48.39	23.33
Visual-Align OPD	3	71.8	50.2	48.36	**23.58**
Visual-Align OPD	6	71.2	50.9	**48.45**	23.36

a better fit. In Figure 3, S, w/o VA, VA, and T denote the student, OPD without visual alignment, visual-aligned model, and teacher, respectively.

As shown in Figure 3, vanilla OPD without visual alignment tends to increase log-PPL on ground-truth captions across captioning benchmarks, suggesting that token-level on-policy distillation alone may shift the student away from the human caption distribution. In contrast, Visual-Align OPD reduces log-PPL compared with vanilla OPD on Flickr30k, VizWiz-Caps, and COCO-Caps (Chen et al., 2015). This indicates that visual attention alignment provides a stabilizing signal during OPD, helping the student maintain better compatibility with human captions while benefiting from teacher supervision. The improvement across multiple captioning datasets further supports that visual alignment makes OPD more robust rather than only improving a specific benchmark. Caption likelihood and attention similarity provide complementary diagnostics of output compatibility and internal visual behavior. Their agreement with task-level gains supports our interpretation, but does not establish that attention changes alone cause these gains.

5 Conclusion

In this paper, we identify that on-policy distillation for VLMs should not only correct token-level predictions, but also address visual grounding errors that emerge during the visual prefill stage. We propose Visual-Align OPD, which combines standard on-policy distillation with a visual attention alignment objective over selected decoder layers. Across multiple VLM backbones and multimodal benchmarks, our method improves over both the base student model and vanilla OPD. Further ablations show that Visual-Align OPD is robust to different alignment weights and layer numbers, with few later layers generally providing stronger gains. Our attention-map and perplexity analyses further suggest that visual alignment improves the student's visual behavior, making OPD more suitable for multimodal distillation.

Limitations

The proposed visual alignment introduces extra attention extraction and matching costs during training. However, it doesn't change the model architecture or introduce extra inference-time overhead.

AI USE STATEMENT

AI tools were used for wording refinement and structural formatting.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

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 *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

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.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. URL https://arxiv.org/abs/1504.00325.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in neural information processing systems*, 36:49250–49267, 2023.

DocVQA 2026 Organizing Team. ICDAR2026 competition on multimodal reasoning over documents in multiple domains. Official benchmark website, 2026. URL https://site.docvqa.org/challenges/2026.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024.

Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3608–3617, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Gurari_VizWiz_Grand_Challenge_CVPR_2018_paper.html.

Danna Gurari, Yinan Zhao, Meng Zhang, and Nilavra Bhattacharya. Captioning images taken by people who are blind. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII 16*, pp. 417–434. Springer, 2020.

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

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.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12):220102, 2024.

Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the association for computational linguistics: ACL 2022*, pp. 2263–2279, 2022.

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.

Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pp. 2641–2649, 2015.

Hejian Sang, Yuanda Xu, Zhengze Zhou, Ran He, Zhipeng Wang, and Jiachen Sun. On-policy self-distillation for reasoning compression. *arXiv e-prints*, pp. arXiv–2603, 2026.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8317–8326, 2019.

Zhengzhuo Xu, Sinan Du, Yiyan Qi, Chengjin Xu, Chun Yuan, and Jian Guo. Chartbench: A benchmark for complex visual reasoning in charts. *arXiv preprint arXiv:2312.15915*, 2023.

Chuanguang Yang, Zhulin An, Libo Huang, Junyu Bi, Xinqiang Yu, Han Yang, Boyu Diao, and Yongjun Xu. Clip-kd: An empirical study of clip model distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15952–15962, 2024a.

Zhendong Yang, Zhe Li, Ailing Zeng, Zexian Li, Chun Yuan, and Yu Li. Vitkd: Feature-based knowledge distillation for vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1379–1388, 2024b.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. doi: 10.1162/tacl_a_00166. URL https://aclanthology.org/Q14-1006/.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

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*, 2025.

Anni Zou, Wenhao Yu, Hongming Zhang, Kaixin Ma, Deng Cai, Zhuosheng Zhang, Hai Zhao, and Dong Yu. Docbench: A benchmark for evaluating llm-based document reading systems. In *Proceedings of the 4th International Workshop on Knowledge-Augmented Methods for Natural Language Processing*, pp. 359–373, 2025.

A APPENDIX

Train/test separation. The training mixture in Table 6 contains only the train portions of the listed datasets. Main results and analyses use test portions held out from distillation training. Thus, a dataset appearing in both the training and evaluation lists denotes different, non-overlapping portions, not reuse of training examples for testing. OCRBench and DocVQA-2026 are used only for evaluation.

Benchmarks (test). As shown in Table 5, we evaluate our method on a diverse set of multimodal benchmarks covering chart reasoning, OCR-centric understanding, document question answering, text-based VQA, and image captioning. Specifically, ChartQA (test) (Masry et al., 2022) evaluates visual and logical reasoning over charts, requiring models to answer questions based on chart contents. OCRBench (test) (Liu et al., 2024) focuses on OCR-related multimodal understanding, including text recognition, scene-text VQA, document-oriented VQA, key information extraction, and handwritten mathematical expression recognition. InfoVQA (test) (Mathew et al., 2022) evaluates question answering over infographic images, where models need to jointly understand textual, graphical, and visual elements. DocVQA-2026 (test) (DocVQA 2026 Organizing Team, 2026) further tests document reasoning ability across diverse document domains, such as reports, papers, slides, posters, maps, comics, infographics, and engineering drawings. TextVQA (test) (Singh et al., 2019) measures whether models can read and reason about text appearing in natural images. For captioning evaluation, Flickr30k (test) (Young et al., 2014) assesses general image captioning on natural images, while VizWiz-Caps (test) (Gurari et al., 2020) focuses on images taken by blind users and is therefore more challenging due to diverse image quality and real-world assistive scenarios.

Table 5: **Overview of evaluation benchmarks (test).** We evaluate multimodal understanding across chart reasoning, OCR, document understanding, text-based VQA, and image captioning.

Benchmark	Task Type	Description
ChartQA (test)	Chart QA	Evaluates visual and logical reasoning over charts, requiring models to answer questions based on chart contents and numerical relations.
OCRBench (test)	OCR-centric understanding	Evaluates OCR-related multimodal abilities, including text recognition, scene-text VQA, document-oriented VQA, key information extraction, and handwritten mathematical expression recognition.
InfoVQA (test)	Infographic QA	Evaluates question answering over infographic images, requiring joint understanding of textual, graphical, and visual elements.
DocVQA-2026 (test)	Document QA	Evaluates multimodal reasoning over document images from diverse domains, such as reports, papers, slides, posters, maps, comics, infographics, and engineering drawings.
TextVQA (test)	Text-based VQA	Tests whether models can read and reason about text appearing in natural images to answer visual questions.
Flickr30k (test)	Image captioning	Evaluates general image captioning ability on natural images with human-written captions.
VizWiz-Caps (test)	Image captioning	Evaluates captioning on images taken by blind users, making it challenging due to real-world image quality, unusual viewpoints, and assistive-use scenarios.

Training dataset composition. As shown in Table 6, our training set is constructed by uniformly mixing samples from six multimodal datasets, including InfoVQA (train) (Mathew et al., 2022), VizWiz-VQA (train) (Gurari et al., 2018), ChartQA (train) (Masry et al., 2022), TextVQA (train) (Singh et al., 2019), Flickr30k (train) (Young et al., 2014), and VizWiz-Caps (train) (Gurari et al., 2020). Although these datasets have different original sizes, we assign each dataset the same sampling weight of 1.0 and sample approximately the same number of instances from each source. This results in a balanced training set of $5{,}000$ samples, where each dataset contributes about 16.7% of the final data. Among them, VQA-style datasets account for $3{,}334$ samples, while captioning

datasets account for 1,666 samples. This balanced composition exposes the model to diverse multimodal abilities, including infographic understanding, chart reasoning, scene-text VQA, general visual question answering, and image captioning.

Table 6: **Composition of the training dataset (train portions only).** All sources use sampling weight 1.0. These training examples are separate from the test portions used for evaluation.

Dataset	Sampled	Orig. Size	Ratio
InfoVQA (train)	833	23,946	3.48%
VizWiz-VQA (train)	833	4,319	19.29%
ChartQA (train)	833	2,500	33.32%
TextVQA (train)	835	34,602	2.41%
Flickr30k (train)	833	29,000	2.87%
VizWiz-Caps (train)	833	7,750	10.75%
Total	5,000	–	–

Evaluation prompts. Table 7 summarizes the prompt templates used in our evaluation. For multiple-choice tasks, we adopt a direct-answer prompt that requires the model to output only the option letter in a unified XML-style format. For open-ended, document understanding, and chart reasoning tasks, we similarly constrain the model to provide only the final answer without explanations, which simplifies answer parsing and reduces evaluation ambiguity. For captioning tasks, we use a concise instruction that asks the model to describe the image in one sentence. Overall, these task-specific templates ensure consistent output formats across different benchmarks while avoiding additional reasoning text during evaluation.

Table 7: **Prompt templates used for evaluation.** We use task-specific prompts for multiple-choice, open-ended, document, chart, and captioning tasks.

Prompt Type	Template
CoT Prompt	`Think step by step, then give your final answer as a single option letter. Use the format: <answer>A</answer>.`
Direct Prompt / MCQ Prompt	`Answer with the option letter only. Use the format: <answer>A</answer>.`
Open-ended Prompt	`Answer the question with a short phrase. Do not explain. Use the format: <answer>ANSWER</answer>.`
Document Prompt	`Answer the question based on the document image. Give only the final answer. Use the format: <answer>ANSWER</answer>.`
ChartQA Prompt	`Answer the question based on the chart. Give only the final answer, such as a number, percentage, or short phrase. Do not explain. Use the format: <answer>ANSWER</answer>.`
Caption Prompt	`Describe this image in one concise sentence.`

Evaluation metrics. For accuracy-based tasks, we report accuracy (Acc.), defined as

$$\text{Acc.} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\left(\hat{y}_i = y_i\right), \tag{24}$$

where N is the number of samples, $\hat{y}_i$ is the predicted answer, y_i is the ground-truth answer, and $\mathbb{I}(\cdot)$ is the indicator function.

For exact-match evaluation, we use

$$\text{EM} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\left(\text{norm}(\hat{y}_i) = \text{norm}(y_i)\right), \tag{25}$$

where $\text{norm}(\cdot)$ denotes the standard answer normalization function.

For captioning tasks, we report METEOR. Given precision P and recall R computed from unigram matches between the generated caption and reference caption, METEOR first computes a weighted harmonic mean:

$$F_{\text{mean}} = \frac{10PR}{R + 9P}. \tag{26}$$

It then applies a fragmentation penalty:

$$\text{Penalty} = \gamma \left(\frac{c}{m}\right)^{\beta}, \tag{27}$$

where m is the number of matched unigrams and c is the number of matched chunks. The final METEOR score is

$$\text{METEOR} = (1 - \text{Penalty}) F_{\text{mean}}. \tag{28}$$

Higher Acc., EM, and METEOR scores indicate better performance.

Training hyperparameters. As shown in Table 8, for Visual-Align OPD training, we use the same hyperparameter setting for both model families, including Qwen3-VL-8B-Instruct $\rightarrow$ Qwen3-VL-2B-Instruct and InternVL3-8B $\rightarrow$ InternVL3-2B. We perform full-parameter training with `bfloat16` precision for one epoch. The learning rate is set to 2×10^{-6} with a warmup ratio of 0.05. The per-device batch size is 1, and the gradient accumulation step is 2, resulting in an effective batch size of 6 under three training GPUs. We set the maximum input length to 4096 and the maximum completion length to 2048. For distillation, we disable sequential KD and SFT loss by setting `seq_kd=false` and $\alpha_{\text{sft}} = 0.0$. The OPD loss weight and KL coefficient are both set to 1, and the visual-alignment weight is set to $\alpha_v = 1.0$. Visual alignment is applied to the last decoder layer, corresponding to `num_align_layers=1`. During training, we use vLLM to generate student rollouts and apply DeepSpeed ZeRO-3 for both the student and teacher models.

Table 8: **Training hyperparameters.** Both model families use the same training settings; their teacher and student models are listed separately.

Hyperparameter	**Qwen3-VL**	**InternVL3**
Teacher model	Qwen3-VL-8B-Instruct	InternVL3-8B
Student model	Qwen3-VL-2B-Instruct	InternVL3-2B
Training type	Full fine-tuning	
Training epochs	1	
Learning rate	2×10^{-6}	
Warmup ratio	0.05	
Precision	bfloat16	
Per-device batch size	1	
Gradient accumulation steps	2	
Number of training GPUs	3	
Effective batch size	6	
Maximum sequence length	4096	
Maximum completion length	2048	
Sequential KD	Disabled	
SFT loss weight α_{sft}	0.0	
OPD loss weight λ_v	1.0	
KL coefficient β	1.0	
Visual-alignment weight α_v	1.0	
Number of aligned layers	1	
Rollout engine	vLLM	
DeepSpeed stage	ZeRO-3	