A. Related Work of Efficient VLMs

With the advent of visual instruction tuning [30, 92] and the scaling of large language models (LLMs) [24, 27, 66], both large-scale open-source [8, 156, 162, 189] and closed-source [7, 28, 62] vision–language models (VLMs) have emerged. However, these large-scale VLMs impose substantial computational demands in real-world scenarios, such as on-device or edge processing. Consequently, there is a growing demand for lightweight VLMs that can be efficiently deployed on resource-constrained devices while maintaining fast inference, driving active research in efficient VLM design. Early efforts have mainly focused on integrating additional visual encoders [35, 70, 116, 178], multiple computer vision backbones [22, 33, 113, 165], or rational embeddings [49, 78, 157] into LLMs [39, 79, 80, 98, 132, 190]. In addition, a growing body of research [75, 76, 95, 112, 142] has explored architectural strategies—such as shared or repetitive feed-forward network (FFN) structures and expanded hidden dimensions—to enhance efficiency without significant performance degradation. Furthermore, several studies [25, 26, 89, 120, 184, 187] propose vision–text aligned training strategies, adopt Mamba architecture [42], or incorporate the mixture-of-experts paradigm [10, 34, 63, 123, 130] to achieve scalable model capacity.

B. The Objective of GRPO

For a question x and its multiple generated responses $\{\hat{y}_j\}_{j=1}^{G}$, the RL objective of GRPO [128] (Generalized Reinforcement Policy Optimization) is defined as:

$$\mathcal{L}_{\text{GRPO}} = -\mathbb{E}_j \left[\mathbb{E}_t \left[\min \left(r_{j,t} A_j,\ \text{clip}(r_{j,t}, 1-\varepsilon, 1+\varepsilon) A_j \right) - \beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}) \right] \right], \tag{9}$$

$$\text{where} \quad r_{j,t} = \frac{\pi_\theta(\hat{y}_{j,t} \mid x, \hat{y}_{j,<t})}{\pi_{\theta_{\text{old}}}(\hat{y}_{j,t} \mid x, \hat{y}_{j,<t})} \quad \text{and} \quad A_j = \frac{\mathcal{R}_j - \text{mean}\left(\{\mathcal{R}_j\}_{j=1}^{G}\right)}{\text{std}\left(\{\mathcal{R}_j\}_{j=1}^{G}\right)}. \tag{10}$$

Here, $r_{j,t}$ denotes the policy ratio for new policy π_θ and old policy $\pi_{\theta_{\text{old}}}$ for each token t, and A_j indicates the advantage computed by normalized rewards $\mathcal{R}$. This objective encourages the new policy π_θ to improve upon the old policy $\pi_{\theta_{\text{old}}}$ according the advantage A. The clipped surrogate objective limits the policy update ratio $r_{j,t}$ to the range $[1-\varepsilon,\ 1+\varepsilon]$, preventing excessively large updates. In addition, KL divergence term $D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})$ penalizes deviation from a reference policy π_{ref}, ensuring regularization for stable training.

In our **Masters** training setup, the total reward is computed as the sum $\mathcal{R} = \mathcal{R}_{\text{acc}} + \mathcal{R}_{\text{distill}}$, from which the advantage is directly derived. Since the updating model is the student, the policy π_θ corresponds to the student's logit-softmax output $P_{\mathcal{S}}$, and the parameter θ represents the student's weight set $\mathbf{W}_{\mathcal{S}}$. In our setup, the policy ratio $r_{j,t}$ is always one because the student is updated only once per training iteration i; hence, the old policy $\pi_{\theta_{\text{old}}}$ and the new policy π_θ are identical. Therefore, the clipped surrogate term becomes redundant, and the objective of GRPO simplifies to

$$\mathcal{L}_{\text{GRPO}} = -\mathbb{E}_j \left[r_{j,t} A_j - \beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}) \right], \tag{11}$$

where $r_{j,t} = 1$. Technically, we still keep the ratio term in the expression to ensure the gradient properly flows to the student parameters during training. Additionally, we set $\beta = 0.1$ to prevent the student from being updated excessively, providing stable regularization.

C. Visual Instruction Tuning Data

We assemble a 1.5M-sample visual instruction tuning dataset that encompasses both real-world and synthetic sources: COCO-ReM [135], iNaturalist2018 [149], VQA-v2 [41], Super-CLEVR [88], MAVIS [181], Geometry3K [99], SQA [101], AI2D [68], SA-1B [70], LLaVAR [183], VSR [91], TallyQA [3], TabMWP [102], KonIQ [56], InternVL [155]-filtered synthetic knowledge dataset covering politics, math, physics, chemistry, RLAI-F [171], CLEVR-Math [90], SROIE [60], ChartQA [109], DocVQA [110], FigureQA [65], GQA [61], InfoVQA [111], M3CoT [19], MapQA [15], OK-VQA [108], TextVQA [134], WildVision [105], DVQA [64], GeoQA+ [13], GeOS [127], IconQA [100], UniGEO [16], GeomVerse [67], Geo170K [38], MathV360K [133], and RAM++ [59]-filtered synthetic data of Infinity-MM [43] covering coarse and fine-grained perception, relation, attribute, and logic reasoning.

D. Additional Parsing Prompts for Accuracy Reward

Prediction Evaluation Prompt

```
System:
You are an evaluation assistant that gives accuracy scores
compared with Ground Truth and Generated Text from AI.
Question is in <question> </question> tag.
Ground Truth is in <ground truth> </ground truth> tag.
Generated Text in <generated text> </generated text> tag.
After reading the Question, compare the Generated Text
against the Ground Truth summary:
  - If the Generated Text fully and correctly captures the core point → 1
  - If it is incorrect or irrelevant → 0
  - If it has repetitive response → 0
  - If it has empty response → 0
Output the numerical evaluation score (0 or 1) after giving a brief explanation.
  - The evaluation score should be wrapped in <answer> </answer> tag.

User:
<question>
{}
</question>

<ground truth>
{}
</ground truth>

<generated text>
{}
</generated text>

Provide the numerical evaluation score after giving a brief explanation.
The evaluation score should be wrapped in <answer> </answer> tag.
```

Accuracy Reward Parsing Prompt

```
System:
You are an evaluation assistant that gives binary accuracy scores (0 or 1)
based on the provided overall summary.
The summary will be wrapped inside <overall_summary>
and </overall_summary> tag.
After reading the summary, briefly output the integer score (0 or 1)
without any text.
Your final output must include only the integer value.

User:
<overall_summary>
{}
</overall_summary>

Please output your integer accuracy score (0 or 1)
based on the summary above without any text.
```

E. Comparing Masters-applied Small VLMs with Closed-source and Large Open-source VLMs.

VLMs	AI2D	ChartQA	MathVista	MMB	MM-Vet	MMMU	MMMU-Pro	MMStar	BLINK	SEED	SEED2+	RWQA
Claude-3.5-Sonnet [7]	81.2	90.8	67.7	82.6	70.1	68.3	51.5	65.1	60.1	61.7	71.7	65.8
Claude-3.7-Sonnet [7]	82.5	92.2	66.8	84.8	70.0	71.0	56.5	65.1	56.6	74.3	67.6	55.4
Claude-4-Sonnet [7]	83.0	-	74.6	-	-	74.4	61.6	69.4	60.4	-	-	69.8
Gemini-1.5-Pro [140]	79.1	87.2	63.9	73.9	64.0	62.2	46.9	59.1	59.1	76.0	70.8	67.5
Gemini-2.0-Flash [140]	83.1	-	70.4	90.0	73.6	69.9	54.4	69.4	64.0	-	-	72.3
Gemini-2.5-Pro [28]	89.5	-	80.9	-	83.3	74.7	-	73.6	-	-	-	-
GLM-4.5V [55]	88.1	86.6	84.6	-	75.2	75.4	65.2	75.3	65.3	-	74.0	
GPT-4o [115]	84.6	85.7	63.8	83.4	69.1	69.1	51.9	64.7	68.0	77.1	72.0	75.4
GPT-4.1 [115]	85.9	-	70.4	-	78.8	74.0	-	69.8	**68.5**	78.0	73.1	78.7
GPT-5-Mini [115]	88.2	-	79.1	-	-	79.0	**67.3**	74.1	-	-	-	**79.0**
GPT-5 [115]	89.5	-	81.9	-	77.6	**84.2**	-	75.7	-	-	-	-
NVLM-72B [31]	85.2	86.0	66.6	-	58.9	59.7	-	63.7	48.0	75.5	68.4	69.9
LLaVA-OneVision-72B [86]	85.6	83.7	67.5	85.8	60.6	56.8	31.0	65.8	55.4	77.5	-	71.9
Molmo-72B [32]	83.4	87.3	58.6	-	61.1	54.1	-	63.3	49.7	74.6	67.6	73.7
Qwen2.5-VL-72B [8]	88.7	89.5	74.2	88.6	76.9	68.2	61.2	70.8	64.4	79.5	73.0	75.7
Qwen3-VL-32B [162]	89.5	89.8	83.8	**90.6**	79.4	76.0	65.3	77.7	67.3	79.9	72.8	**79.0**
InternVL3-78B [189]	**89.7**	89.7	79.0	89.0	81.3	72.2	62.3	72.5	66.3	78.7	71.9	78.0
InternVL3.5-38B [156]	87.8	88.8	81.9	90.3	82.2	76.9	66.0	75.3	60.9	79.1	71.0	75.9
Qwen2.5-VL-7B-**Masters**	88.6	95.6	78.8	89.1	81.7	71.3	60.6	74.9	67.2	81.8	**75.9**	77.3
Qwen3-VL-8B-**Masters**	88.5	**95.9**	81.8	89.5	79.4	72.9	61.4	79.7	72.3	81.7	75.1	77.5
InternVL3-8B-**Masters**	88.9	94.8	82.3	90.1	83.8	74.0	58.8	82.0	68.0	**82.6**	75.0	74.8
InternVL3.5-8B-**Masters**	87.2	95.1	**85.0**	88.2	**85.6**	72.7	58.1	**80.8**	67.8	81.4	75.5	74.9

F. Detailed Comparison for Masters

In this section, we provide extended analyses and comparisons that further validate the design choices of Masters. We cover curriculum learning baselines, the effect of offline versus semi-online RL, the impact of masking on reasoning quality, the behavior of the distillation reward, teacher trajectory usage, reward judge reliability, difficulty measurement via distributional divergence, and the interplay between masking and RL components.

F.1. Comparison with Curriculum Learning

A natural question is whether curriculum learning (CL) alone can address the capacity gap between teacher and student without explicit masking. To investigate this, we employ Qwen3-VL-32B to estimate the difficulty of all 1.5M training samples on a 1–10 difficulty scale and conduct distillation to Qwen3-VL-8B in an easy-to-hard ordering. As shown in the table below, CL yields an improvement over the vanilla baseline; however, Masters remains beneficial even on top of the CL ordering. This suggests that CL alone cannot fully resolve the representational mismatch between teacher and student, as it only controls the *data presentation order* without modifying the teacher signal itself. In contrast, Masters explicitly addresses this gap at the model-capacity level by progressively simplifying and restoring the teacher's representations.

Method	MathVista	MM-Vet	MMMU	BLINK	Avg
Naive	78.4	75.8	70.4	69.9	73.6
+Curriculum	78.5	75.8	70.6	70.1	73.7
Masters w/o Mask-Progressive	79.0	76.5	71.1	70.5	74.3
+Curriculum	79.2	76.7	71.2	70.7	74.5
Masters	81.8	79.4	72.9	72.3	76.6
+Curriculum	81.9	79.4	73.0	72.4	76.7

Online Percentage	MathVista	MM-Vet	MMMU	BLINK	Avg	Wall Time
0%	81.8	79.4	72.9	72.3	76.6	49 hours
5%	81.9	79.4	72.9	72.4	76.7	73 hours
10%	81.8	79.6	73.0	72.4	76.7	97 hours
15%	81.9	79.5	73.0	72.4	76.8	119 hours
20%	82.1	79.6	73.1	72.6	76.9	148 hours

F.2. Offline versus Semi-Online RL

While online RL generally offers stronger exploration, the goal of Masters is not exploration-heavy policy improvement but rather scalable and stable knowledge transfer under limited compute budgets. Nonetheless, we investigate a semi-online variant using Qwen3-VL-32B and Qwen3-VL-8B, where a fraction of training iterations involve online rollouts. Specifically, 0% corresponds to the fully offline setting and 10% means that online rollouts are performed during 10% of total training iterations. We find that increasing the online percentage can improve performance, but the training time grows rapidly, reinforcing the practical advantage of the offline formulation adopted by Masters.

F.3. Impact of Masking on Reasoning Quality

A potential concern is that masking teacher weights may disrupt the coherence of the teacher's reasoning. We clarify that Masters does not follow a strict think-answer paradigm; instead, all parts of the response except the final answer are treated

as the reasoning portion. To evaluate whether masking degrades reasoning quality, we introduce an additional reasoning reward (R_{rea}) that measures whether the reasoning part is supportive of the final answer. We then evaluate the masked teacher model's response quality through both the accuracy reward and the reasoning reward across varying masking ratio schedules (0.4→0 and 0.2→0). As shown in Table **??**, the rewards do not change substantially under masking. Furthermore, training Masters with the reasoning reward yields only marginal improvement, suggesting that masking does not introduce significant reasoning degradation.

Qwen3-VL-32B	0.4	0.3	0.2	0.1	0
R_{acc}	0.54	0.57	0.58	0.61	0.62
R_{rea}	0.79	0.80	0.82	0.83	0.83

InternVL3.5-38B	0.2	0.15	0.1	0.05	0
R_{acc}	0.58	0.60	0.62	0.63	0.64
R_{rea}	0.72	0.73	0.75	0.75	0.76

	MathVista	MM-Vet	MMMU	BLINK	Avg
Masters	81.8	79.4	72.9	72.3	76.60
Masters (w/ R_{rea})	81.9	79.5	73.0	72.2	76.65

F.4. Masking and Representational Complexity

The goal of masking is to simplify the teacher's representational complexity to facilitate alignment with the student. Consider a simple illustrative example: suppose the teacher's feature mapping is $y = 0.01x^3 + 3x^2 - 0.03x + 5$ and the student's is $y = w_1x^2 + w_2$. Due to the difference in capacity, directly fitting the student to the teacher is suboptimal. By removing small-magnitude weights in the teacher, we reduce its complexity to $y = 3x^2 + 5$, which the student can now match exactly ($w_1 = 3, w_2 = 5$). This principle underlies our masking strategy: pruning low-magnitude weights in the teacher reduces its representational complexity for better alignment with the student.

To empirically validate that masking controls difficulty, we generate student responses and masked teacher responses for all training samples across varying masking ratios. We then compute the Jensen–Shannon Divergence (JSD) between the masked teacher and student response distributions. As shown in table, decreasing the masking ratio (i.e., restoring the teacher) increases JSD, confirming that difficulty increases as the teacher becomes more expressive. Interestingly, over-masking also increases JSD, suggesting that excessive masking can distort the teacher signal rather than simplify it.

Qwen3-VL-(32B→8B)	0.5	0.4	0.3	0.2	0.1	0
JSD	0.36	**0.11**	0.19	0.30	0.39	0.47
InternVL3.5-(38B→8B)	0.3	0.2	0.15	0.1	0.05	0
JSD	0.39	**0.16**	0.29	0.36	0.48	0.56

α	AI2D	MathVista	MMB	MM-Vet	MMStar	Avg
0	86.5	82.3	88.1	85.0	75.8	83.5
0.3	86.9	83.8	**88.6**	85.4	78.5	84.6
0.5	87.0	84.1	88.4	**85.8**	79.6	85.0
0.7	87.1	84.6	88.3	85.5	80.5	85.2
1	**87.2**	**85.0**	88.2	85.6	**80.8**	**85.4**

F.5. Behavior of the Distillation Reward

One might worry that combining accuracy and distillation rewards linearly could reinforce incorrect responses. However, this concern stems from a misunderstanding of how the distillation reward operates. When the teacher considers a response to be incorrect, it naturally assigns low confidence to the corresponding tokens. In this case, high teacher–student alignment means that the student also assigns low confidence to those incorrect tokens. Therefore, a high R_{distill} does not reinforce incorrect responses but rather reflects that the student correctly matches the teacher's low-confidence distribution on erroneous parts. To empirically verify this, we scale the distillation reward via $R_{\text{acc}} + \alpha R_{\text{distill}}$ and vary α. We observe that increasing α consistently improves performance, confirming that the distillation reward provides a complementary and beneficial training signal.

F.6. Role of Teacher Trajectories

We examine the effect of mixing teacher and student trajectories during training. Using teacher trajectories leads to better performance than using student trajectories alone; however, the two should be balanced by maintaining the same total number of generated responses. Under fixed sampling budgets, using pure teacher trajectories is at least not worse than using pure student trajectories, and its primary benefit lies in stabilizing alignment when combined with student responses. All experiments in this comparison are conducted with both accuracy and distillation rewards active.

Source	AI2D	MathVista	MMB	MM-Vet	MMMU	Avg	$\Delta(\%p)$
$\mathcal{S}(\#8)$	86.4	83.8	86.3	85.0	71.0	82.5	-
$\mathcal{S}(\#4) + \mathcal{T}(\#4)$	87.2	85.0	88.2	85.6	72.7	83.7	+1.2%p
$\mathcal{T}(\#8)$	87.0	84.8	86.3	85.3	72.6	83.2	+0.7%p

Response	Student Judge	Teacher Judge	AI2D	MathVista	MMB	MM-Vet	MMStar	Avg	$\Delta(\%p)$
Student Teacher	✓ ✗	✗ ✓	87.2	85.0	88.2	85.6	80.8	85.36	-
Student Teacher	✗ ✗	✓ ✓	87.3	85.1	88.2	85.6	80.7	85.38	+0.02%p
Student Teacher	✓ ✓	✗ ✗	87.2	85.1	88.1	85.5	80.8	85.34	-0.02%p

F.7. Reliability of the Student Model as Judge

To reduce computational cost and GPU memory usage, Masters adopts a smaller student model as the reward judge. To evaluate the reliability of this choice, we compare student and teacher models as judges. We observe that switching between student and teacher judges results in negligible performance differences, indicating that the reward signals are consistent across model scales. This is because the accuracy reward involves comparing model responses against ground-truth labels, which is a relatively straightforward evaluation task. Therefore, adopting a small model as the judge provides comparable reward reliability while significantly reducing resource requirements.

F.8. Interplay between Masking and RL

To disentangle the contributions of masking and RL, we conduct an ablation where each component is applied independently. As shown in table, RL with only an accuracy reward yields a limited improvement, while masking alone provides a moderate gain. The largest improvement is achieved when masked distillation and RL are combined. This indicates that masking primarily facilitates capacity alignment by simplifying teacher behaviors, which in turn enables RL with the distillation reward to achieve substantially larger gains. The two components are thus complementary: masking creates favorable conditions for effective reinforcement learning.

InternVL3.5-8B (InternVL3.5-38B)	AI2D	MathVista	MM-Vet	MMStar	BLINK	RWQA	Avg	$\Delta(\%p)$
Naive	84.8	78.9	83.5	69.8	60.0	67.9	74.2	-
Naive + Mask-Progressive	86.0	80.1	84.6	71.0	61.2	69.1	75.3	+1.1%p
Naive + RL (R_{acc})	85.3	79.2	83.9	70.4	60.5	68.6	74.7	+0.5%p
Naive + RL ($R_{acc} + R_{distill}$)	86.3	80.6	84.9	72.6	61.4	69.6	75.9	+1.7%p
Naive + Mask-Progressive + RL ($R_{acc} + R_{distill}$)	**87.2**	**85.0**	**85.6**	**80.8**	**67.8**	**74.9**	**80.2**	**+6.0%p**

Qwen3-VL-8B (Qwen3-VL-32B)	AI2D	MathVista	MM-Vet	MMStar	BLINK	RWQA	Avg	$\Delta(\%p)$
Naive	86.4	78.4	75.8	73.1	69.9	71.8	75.9	-
Naive + Mask-Progressive	87.3	79.8	77.3	76.0	70.9	74.3	77.6	+1.7%p
Naive + RL (R_{acc})	86.9	79.0	76.5	74.2	70.3	72.9	76.6	+0.7%p
Naive + RL ($R_{acc} + R_{distill}$)	87.8	80.3	77.7	76.6	71.5	74.9	78.1	+2.2%p
Naive + Mask-Progressive + RL ($R_{acc} + R_{distill}$)	**88.5**	**81.8**	**79.4**	**79.7**	**72.3**	**77.5**	**79.9**	**+4.0%p**

F.9. Masking Ratio and Pre-Generation Cost

Across different model families, effective masking ratios are consistently around 0.2 or 0.4, as reported in the main paper. In practice, 0.2 serves as a robust default, since over-masking beyond the optimal ratio leads to significant performance degradation. Regarding pre-generation cost, using 256 A100 GPUs with 8 generated responses for 1.5M samples, pre-generation takes approximately 1.5 days for 72B/78B models, about 1 day for 32B/38B models, 12–16 hours for 7B/8B/14B models, and 6–9 hours for 2B/3B/4B models. This one-time overhead remains significantly lower than fully online RL training. For reference, training Masters with eight online rollout samples for the student and teacher on 1k training samples takes around 0.5 hours on 256 A100 80GB GPUs, which corresponds to approximately 31 days for the full 1.5M sample dataset.