Uncertainty-Aware Listwise Reinforcement Fine-Tuning for Fine-Grained Visual Classification

Zhi Tan[1,2], **Hulingxiao He**[2], **Yuxin Peng**[2,*]
[1]School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University
[2]Wangxuan Institute of Computer Technology, Peking University
tanzhi@stu.pku.edu.cn hehulingxiao@stu.pku.edu.cn pengyuxin@pku.edu.cn

Abstract

Fine-grained visual classification (FGVC) with Multimodal Large Language Models (MLLMs) remains challenging, as visually similar subordinate categories inherently induce classification uncertainty due to subtle inter-class differences. Existing reinforcement learning methods typically require the model to generate a single class name and optimize it with sparse correctness rewards. However, a seemingly incorrect prediction can still provide highly informative signals if it falls within the same local confusion set as the ground-truth class. By treating all non-exact matches as equally wrong, the rigid single-answer paradigm overlooks these valuable near-misses. Consequently, it fails to capture the model's intrinsic uncertainty and provides limited learning signals. To address these limitations, we propose List-GRPO, an uncertainty-aware listwise Group Relative Policy Optimization framework. Rather than enforcing a single prediction, our method prompts the model to adaptively generate a ranked candidate list, explicitly capturing its predictive uncertainty. Furthermore, to mitigate the advantage vanishing when online exploration completely misses the target, we introduce a dynamic ground-truth anchored response strategy. We synthesize an off-policy anchor by prepending the ground truth to the most frequent hard negatives mined from online samples. To prevent the anchored response from dominating learning, we isolate the advantage estimation of online samples from the anchored response and use a gating coefficient to modulate its contribution. Extensive experiments on multiple fine-grained benchmarks demonstrate that List-GRPO significantly improves Top-1 accuracy while maintaining high ground-truth coverage with only about three candidate predictions on average.

1 Introduction

Fine-grained visual classification (FGVC) aims to distinguish subordinate categories that are visually similar but semantically different, such as bird species or flower types [47, 36]. Unlike generic visual recognition, FGVC requires models to localize subtle discriminative cues, including part shape, texture, color patterns, and domain-specific attributes, and further map these cues to precise category names. While multimodal large language models (MLLMs) [3, 12, 22, 44, 20, 46] have demonstrated remarkable capabilities across general multimodal benchmarks [28, 26, 27, 49, 29, 53, 24], adapting them to fine-grained recognition remains challenging [15, 52, 10]. By formulating visual classification as language generation, MLLMs can directly produce open-ended category names [51]. However, their recognition performance is often limited by the misalignment between the discriminative visual evidence captured by the vision encoder and the fine-grained semantic knowledge stored in the language decoder [10, 15].

Reinforcement learning with verifiable rewards (RLVR) using Group Relative Policy Optimization (GRPO) [39] has emerged as a promising approach to improve the performance of MLLMs on

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

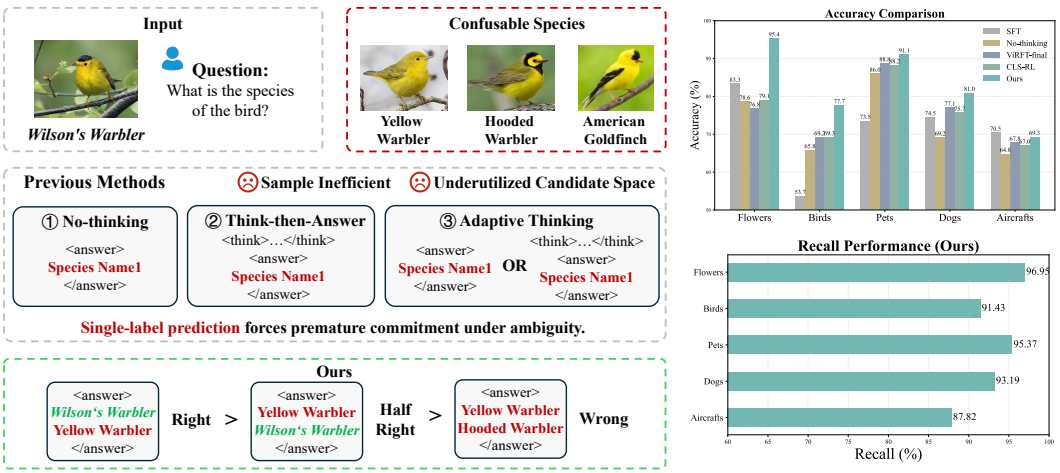

Figure 1: **Comparison of Optimization Paradigms for FGVC.** Existing approaches optimize MLLMs for FGVC as a hard generation task, where the model is required to produce a single class name and is judged as entirely correct or entirely wrong. In contrast, our **List-GRPO** framework allows the model to generate an adaptive candidate set of category names and optimizes it with recall and precision rewards. This approach provides more effective learning signals, improves sample efficiency, and achieves a better balance between coverage and top-1 accuracy, as illustrated by quantitative improvements across multiple fine-grained datasets.

fine-grained classification tasks [48, 18, 50, 23]. However, as illustrated in Figure 1, most existing approaches formulate FGVC as an exact-match generation problem, where the model is required to output a single class name and is rewarded only when this prediction exactly matches the ground truth. This formulation creates a mismatch with the intrinsic structure of FGVC. Fine-grained categories often form local confusion sets, in which multiple subordinate classes share highly similar visual appearances and differ only in subtle attributes. As a result, semantically meaningful predictions are treated as entirely wrong, even though they may indicate that the model has already captured relevant fine-grained evidence and narrowed its prediction to the correct visual neighborhood.

In FGVC, resolving visual ambiguity inherently requires evaluating multiple confusable categories before making a final decision [8, 13]. Collapsing this uncertainty into a single label discards useful structure in the model's belief space and provides coarse binary feedback during optimization. To address this limitation, we argue that FGVC with MLLMs should be reformulated from forced single-answer generation to uncertainty-aware adaptive candidate list generation. Even when the top-ranked prediction is incorrect, including the ground-truth category in the candidate list suggests that the model has identified the correct fine-grained neighborhood. This provides an opportunity for reinforcement learning to exploit not only exact matches, but also informative near-miss predictions that are otherwise discarded by standard exact-match rewards.

To this end, we propose **List-GRPO**, an uncertainty-aware listwise reinforcement learning framework for FGVC with MLLMs. In our formulation, uncertainty awareness refers to the model's ability to explicitly express visual ambiguity through an adaptive ranked candidate list. Instead of producing a single overconfident prediction, the model generates a ranked list of plausible categories. This listwise formulation exposes the local confusion set considered by the model and provides richer supervision than a solitary answer. To optimize such outputs, we design recall and precision rewards. The recall reward encourages the ground-truth category to be included in the candidate list, while the precision reward promotes concise lists and accurate top-ranked predictions. By decoupling candidate recall from top-rank precision, List-GRPO converts sparse exact-match feedback into structured listwise supervision while preserving Top-1 accuracy as the primary objective. To address the advantage vanishing when early answers fail to include the ground truth, we introduce a ground-truth anchored response construction strategy. For each training instance, we construct an auxiliary positive response by placing the ground-truth category at the first position and appending hard-negative categories mined from online rollouts. This anchor provides a high-quality positive hypothesis within the GRPO group and encourages the model to distinguish the ground truth from its most confusing alternatives. To prevent the constructed response from dominating on-policy learning,

we further propose isolated advantage estimation, which separates constructed anchors from online responses during advantage computation. This design allows synthetic positive guidance to stabilize optimization without distorting the relative preference structure among on-policy samples.

In summary, our contributions are threefold: (1) We propose **List-GRPO**, a listwise reinforcement learning framework that reframes FGVC with MLLMs as uncertainty-aware adaptive candidate list generation. Rather than forcing the model to produce a single prediction, List-GRPO allows it to express fine-grained visual ambiguity by adaptively generating a ranked list of plausible categories according to its predictive uncertainty. (2) To facilitate this listwise formulation, we introduce a ground-truth anchored response construction strategy that synthesizes positive responses by prioritizing the ground truth and appending hard-negative categories mined via online sampling. Furthermore, we propose an isolated GRPO optimization scheme that decouples on-policy from constructed responses during advantage estimation, ensuring that synthetic guidance augments learning without distorting the model's inherent preference structure. (3) Extensive experiments across multiple benchmarks demonstrate that our List-GRPO significantly outperforms single-label baselines, yielding substantial gains in Top-1 accuracy while maintaining high ground-truth candidate recall with an average list length of only about three candidates.

2 Related Work

Fine-Grained Visual Classification with MLLMs Fine-grained visual classification (FGVC) aims to distinguish subordinate categories that share highly similar global appearances but differ in subtle local cues, such as part shapes, color patterns, or domain-specific anatomical attributes. While recent multimodal large language models (MLLMs), including LLaVA [22, 1, 21], Qwen-VL [3–5], and InternVL [6, 56], have demonstrated remarkable open-ended visual understanding and instruction-following capabilities, their success on general vision-language tasks does not translate to FGVC. In fine-grained scenarios, a model must not only pinpoint highly localized discriminative regions but also associate them with precise, expert-level names. Consequently, these models remain highly susceptible to visual grounding errors and label-space ambiguities.

Recent studies have analyzed the limitations of MLLMs on fine-grained recognition from different perspectives. FINER [15] attributes this limitation to a modality grounding gap, where the model may possess relevant textual knowledge but fails to reliably ground it in subtle image evidence. Finedefics [10] incorporates attribute descriptions and contrastive objectives to enforce stricter alignment between visual details and category semantics. Other works explore output space reformulations: Nlg2choice [17] suggests that while free-form responses contain valuable cues, relying on one-step generation for final prediction is unreliable; In addition, hierarchical visual recognition emphasizes taxonomic consistency rather than treating labels as isolated class names. TARA [11] injects taxonomic knowledge into LMMs by aligning intermediate representations with biology foundation models, improving both hierarchical consistency and leaf-node recognition. Beyond method-oriented studies, recent benchmarks further expose the difficulty of fine-grained and long-tail visual knowledge in MLLMs. FROW [33] studies open-vocabulary, open-set fine-grained recognition, showing that performance drops as the semantic search space grows. WorldVQA [55] evaluates MLLMs on fine-grained visual knowledge by decoupling entity recognition from reasoning. Results indicate that even state-of-the-art MLLMs achieve below 50% accuracy, highlighting their difficulty in identifying fine-grained visual entities.

Reinforcement Fine-Tuning Reinforcement learning (RL) has emerged as a pivotal post-training paradigm for aligning large language and multimodal models. While early alignment strategies heavily relied on human preference feedback optimized via PPO-style algorithms [57, 42, 32, 38], recent advancements like DPO [37] and GRPO [39] streamline this pipeline by bypassing explicit reward models or value networks. GRPO, in particular, estimates relative advantages directly within a group of sampled responses, proving highly effective for fine-tuning scenarios equipped with automatically verifiable rewards.

This verifiable reward paradigm (RLVR) has rapidly expanded from text-only reasoning to multimodal contexts [19, 34, 23, 40, 30, 7]. For instance, Visual-RFT [23] introduces task-specific GRPO rewards, utilizing exact-match criteria for classification and IoU metrics for detection. Contemporaneously, frameworks like VLM-R1 [40], MM-Eureka [30], and OpenVLThinker [7] demonstrate that rule-based verification significantly boosts visual reasoning and out-of-distribution generalization.

Recently, RLVR has also been explored for fine-grained classification. CLS-RL [18] investigates few-shot image classification and introduces an adaptive thinking mechanism to dynamically modulate reasoning lengths. Fine-R1 [9] employs Triplet Augmented Policy Optimization (TAPO) to enhance discrimination across visually confusing negatives, while SpeciaRL [2] formulates a specificity-aware reward to penalize overly broad predictions in open-world settings. DiVE-k [16] constructs discriminative multiple-choice signals from top-k predictions to encourage closed-set comparison among visually similar candidates. However, these methods predominantly optimize single-label exact-match objectives, which can be sample-inefficient and fail to exploit the informative value of near-misses within the candidate space. In contrast, our proposed List-GRPO explicitly generates an adaptive candidate set, leveraging the inherent uncertainty of fine-grained confusions to deliver a denser, more efficient policy learning signal. Compared to DiVE-k, which constructs candidate options offline for multiple-choice policy optimization and uses candidate generation followed by answer selection at inference. List-GRPO directly generates and optimizes a ranked list in one response. Its training anchors are constructed from the current policy's lists, allowing the confusable categories used for supervision to evolve during optimization.

3 Method

3.1 Overview

In fine-grained visual classification (FGVC), visually similar subcategories often share local attributes and differ only in subtle details, inherently leading to classification uncertainty. Under such conditions, a seemingly incorrect prediction can still provide highly informative signals if it falls within the same local confusion set as the ground-truth class. Therefore, rather than penalizing all incorrect responses equally, we argue that the optimization process should evaluate two distinct capabilities: whether the model successfully recalls the ground-truth class into a candidate list, and whether it ranks the ground truth as the top prediction.

To instantiate this intuition, we propose **List-GRPO**, a framework for fine-grained classification with Multimodal Large Language Models (MLLMs). As illustrated in Figure 2, for each training instance, List-GRPO proceeds in three steps. First, it samples $N-1$ online listwise responses from the current policy to explore the decision space. Second, it dynamically constructs a ground-truth anchored response by combining the target category with the most frequent confusing categories mined from the online samples. Third, it updates the policy using an isolated GRPO objective, ensuring that the off-policy anchored response guides the decision boundary without contaminating the on-policy advantage estimation. In the following sections, we will elaborate on each of these steps in detail.

3.2 Uncertainty-Aware Candidate List Online Sampling

During the online exploration phase, given an input image x, the policy model is explicitly prompted to structure its response in a listwise format:

$$\texttt{<answer>}\ c_1, c_2, \ldots, c_m\ \texttt{</answer>}. \tag{1}$$

This formulation enables the model to output a single category when highly confident, or to expand the candidate set when uncertain. Each generated response o_i is parsed into a normalized list $L_i = [c_{i,1}, c_{i,2}, \ldots, c_{i,m_i}]$, where $c_{i,j}$ denotes the j-th predicted category in the i-th response.

A fundamental challenge in standard GRPO for classification tasks is reward sparsity: a strict top-1 exact-match reward treats all incorrect answers equally, failing to acknowledge when the model successfully restricts the visual evidence to the correct semantic neighborhood. To address this, we compute the ground-truth rank ρ_i within L_i and decouple the reward into recall and precision components:

$$r_i^{\text{rec}} = \mathbb{I}[\rho_i < \infty], \quad r_i^{\text{pre}} = \mathbb{I}[\rho_i = 1]. \tag{2}$$

The recall reward r_i^{rec} encourages the model to include the ground-truth class within the top L predictions, while the precision reward r_i^{pre} incentivizes the model to rank the ground truth as the top prediction. The final composite reward is defined as:

$$r_i = \lambda_{\text{rec}} r_i^{\text{rec}} + \lambda_{\text{pre}} r_i^{\text{pre}}. \tag{3}$$

Here, the recall term r_i^{rec} densifies the learning signal by encouraging the inclusion of the target class, while the precision term r_i^{pre} aggressively optimizes for the final accuracy. To limit reward gains from excessively long responses, lists with more than $L_{\max}$ candidates receive zero reward.

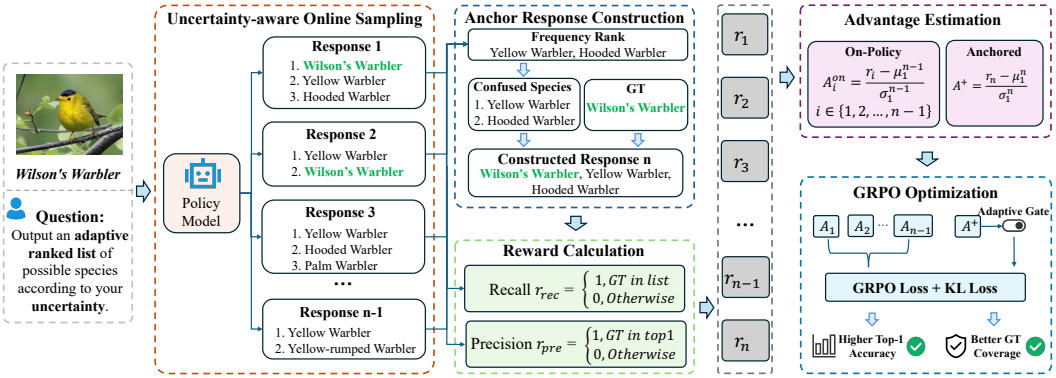

Figure 2: Overview of our List-GRPO framework. Our method allows the model to generate adaptive ranked candidate lists, capturing its uncertainty over confusable classes. To guide learning toward correct local decisions, a ground-truth anchored response is dynamically constructed from the most frequent confusions and optimized with isolated advantage estimation, shaping the decision boundary without interfering with online sample comparisons.

3.3 Dynamic Anchored Response Construction

Although the recall-precision decomposition effectively densifies the reward, pure online sampling $\mathcal{O} = \{o_1, o_2, \ldots, o_{N-1}\} \sim \pi_{\theta_{\text{old}}}$ may suffer from degenerate exploration. Specifically, in early training stages or under severe fine-grained difficulty, all samples might completely miss the ground truth, leading to the vanishing advantage problem. To anchor the optimization space and guarantee a positive learning signal, we synthesize one off-policy positive response $o^\dagger$ per instance. Crucially, instead of appending arbitrary negative distractors, we dynamically construct this anchor to reflect the model's current cognitive state. We aggregate the non-ground-truth categories from the online sample lists and compute their occurrence frequencies:

$$f(c) = \sum_{i=1}^{N-1} \mathbb{I}[c \in L_i], \quad c \neq y. \tag{4}$$

By extracting the top-M most frequent confusable predictions $\{\tilde{c}_1, \ldots, \tilde{c}_M\}$, filtered by a minimum occurrence count of 2, we capture the policy's primary local confusions. The anchored response is then constructed by prepending the ground truth to these hard negatives:

$$o^\dagger = \langle y, \tilde{c}_1, \ldots, \tilde{c}_M \rangle. \tag{5}$$

This design not only guarantees a positive reward but explicitly pits the ground truth against the current hardest negative distractors. Consequently, it encourages the model to refine the local decision boundary between the ground truth and its nearest confusable categories, rather than rigidly committing to an isolated label.

3.4 Isolated Advantage Estimation

While $o^\dagger$ provides a crucial boundary-shaping signal, directly incorporating it into the online group $\mathcal{O}$ for advantage estimation violates on-policy assumptions. The off-policy anchor would artificially distort the reward mean and variance, thereby contaminating the relative preference signals among the genuine online explorations.

To maintain the integrity of the advantage estimation, we decouple the process. The $N-1$ online samples compute their advantages strictly within their empirical distribution:

$$\mu_{\text{on}} = \frac{1}{N-1} \sum_{i=1}^{N-1} r_i, \quad \sigma_{\text{on}} = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N-1} (r_i - \mu_{\text{on}})^2}, \quad A_i^{\text{on}} = \frac{r_i - \mu_{\text{on}}}{\sigma_{\text{on}} + \epsilon}. \tag{6}$$

Conversely, the anchored response $o^\dagger$ computes its reward $r^\dagger$ and is normalized against the mixed group statistics $(\mu_{\text{mix}}, \sigma_{\text{mix}})$ derived from $\mathcal{O} \cup \{o^\dagger\}$, yielding an asymmetric advantage $A^\dagger$:

$$\mu_{\text{mix}} = \frac{1}{N}\left(\sum_{i=1}^{N-1} r_i + r^\dagger\right), \sigma_{\text{mix}} = \sqrt{\frac{1}{N}\left(\sum_{i=1}^{N-1}(r_i - \mu_{\text{mix}})^2 + (r^\dagger - \mu_{\text{mix}})^2\right)}, A^\dagger = \frac{r^\dagger - \mu_{\text{mix}}}{\sigma_{\text{mix}} + \epsilon}. \tag{7}$$

If the current policy already predicts the ground-truth class at the top position in any online sample (i.e., $\exists i, r_i^{\text{pre}} = 1$), enforcing the anchored response becomes redundant and may lead to over-regularization. To address this, we introduce a dynamic gating coefficient $\eta^\dagger$:

$$\eta^\dagger = \begin{cases} 0, & \exists i \in [1, N-1],\ r_i^{\text{pre}} = 1, \\ \eta, & \text{otherwise.} \end{cases} \tag{8}$$

The overall GRPO-style training objective combines the online samples, the anchored response, and KL regularization:

$$\mathcal{L} = \underbrace{\mathcal{L}_{\text{on}}}_{\text{online GRPO}} + \underbrace{\mathcal{L}_{\text{anc}}}_{\text{anchored GRPO}} + \underbrace{\beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})}_{\text{policy regularization}}. \tag{9}$$

The online GRPO loss $\mathcal{L}_{\text{on}}$ is computed over the $N-1$ sampled responses with the standard clipped advantage:

$$\mathcal{L}_{\text{on}} = -\frac{1}{N-1}\sum_{i=1}^{N-1} \min\left(q_i(\theta)A_i^{\text{on}}, \text{clip}(q_i(\theta), 1-\epsilon, 1+\epsilon)A_i^{\text{on}}\right), \tag{10}$$

where $q_i(\theta) = \frac{\pi_\theta(o_i|x)}{\pi_{\theta_{\text{old}}}(o_i|x)}$ is the importance sampling ratio. The anchored loss $\mathcal{L}_{\text{anc}}$ is computed similarly but is modulated by the gating coefficient

$$\mathcal{L}_{\text{anc}} = -\eta^\dagger \min\left(q^\dagger(\theta)A^\dagger, \text{clip}(q^\dagger(\theta), 1-\epsilon, 1+\epsilon)A^\dagger\right). \tag{11}$$

This decomposition makes explicit that $\mathcal{L}_{\text{on}}$ handles the GRPO update from online samples, $\mathcal{L}_{\text{anc}}$ enforces boundary shaping via the anchor, and the KL term regularizes the policy relative to the reference model. This gating mechanism ensures that the constructed anchor acts as a safety net, contributing to the gradient only when the online samples fail to achieve top-1 precision. It adaptively calibrates the strength of the boundary-aware supervision, avoiding redundant updates and allowing the standard on-policy rollouts to naturally refine the predictive distribution once the target is reliably captured.

4 Experiments

4.1 Experimental Settings

Datasets We evaluate our method on five widely used fine-grained visual classification (FGVC) benchmarks: **CUB-200-2011** [45] (200 bird species), **Stanford Dogs** [14] (120 dog breeds), **Oxford Flowers** [31] (102 flower categories), **FGVC-Aircraft** [25] (100 aircraft variants), and **Oxford-IIIT Pet** [35] (37 pet breeds). Since acquiring large-scale, expert-annotated data for fine-grained domains is often prohibitively expensive in practice, we focus on evaluating the model's sample efficiency. Specifically, we conduct experiments under few-shot settings by sampling a limited number of instances per class ($k \in \{1, 2, 4\}$) for training, while reporting the performance on the complete standard test sets.

Baselines for Comparison We benchmark our approach under an **Open-Ended QA** evaluation following ViRFT [23], where the model is prompted to directly generate the answer without being given a candidate set. Our comparison includes the following baselines: (1) **Supervised Fine-Tuning (SFT)**: The MLLM is fine-tuned on answer-only targets using standard cross-entropy loss, without step-by-step reasoning annotations, and trained with the same number of optimization steps as our reinforcement learning framework. (2) **No-Thinking RL**: A direct-generation baseline optimized

via GRPO using a strict exact-match reward, which produces a single label without intermediate reasoning or candidate list expansion. (3) **ViRFT** [23]: A method that employs GRPO to fine-tune the MLLM with a think-then-answer framework. (4) **CLS-RL** [18]: This GRPO approach prompts the MLLM to adaptively decide whether to generate an explicit thinking process before producing the final answer. Our method and all baselines are trained and evaluated under the same open-ended format without restricting to a predefined candidate set, ensuring a fair comparison across approaches.

Table 1: Performance comparison of fine-grained classification accuracy (%) across five datasets under open-ended question answering setting. "Ours" denotes our proposed framework. The best results in each setting are highlighted in bold, and ↑ indicates the absolute performance gain over the strongest baseline.

Model	Oxford Flowers	CUB Birds	Oxford Pets	Stanford Dogs	FGVC Aircrafts	Average
Qwen2.5-VL-7B	63.54	51.00	70.65	64.94	52.54	60.53
1 Sample per Class						
SFT	51.24	43.82	70.46	56.61	59.47	56.32
No-Thinking RL	64.76	56.83	80.59	65.76	65.02	66.59
ViRFT [23]	62.20	59.91	79.56	72.20	63.10	67.39
CLS-RL [18]	63.34	64.00	76.91	70.69	63.91	67.77
Ours	**67.80** ↑ 3.04	**67.07** ↑ 3.07	**83.81** ↑ 3.22	**77.32** ↑ 5.12	**70.29** ↑ 5.27	**73.26** ↑ 5.49
2 Samples per Class						
SFT	58.95	46.30	61.98	66.44	65.64	59.86
No-Thinking RL	65.45	59.13	84.82	67.22	62.74	67.87
ViRFT [23]	72.79	64.15	82.53	72.49	64.57	71.31
CLS-RL [18]	74.26	68.45	83.10	71.88	67.51	73.04
Ours	**86.76** ↑ 12.50	**73.92** ↑ 5.47	**88.12** ↑ 3.30	**79.86** ↑ 7.37	**69.27** ↑ 1.76	**79.59** ↑ 6.55
4 Samples per Class						
SFT	83.96	55.28	75.93	74.90	**70.45**	72.10
No-Thinking RL	78.64	65.81	86.02	69.15	64.84	72.89
ViRFT [23]	76.82	69.23	88.77	77.06	67.81	75.94
CLS-RL [18]	79.09	69.26	88.25	75.71	67.03	75.87
Ours	**95.37** ↑ 12.02	**77.70** ↑ 8.44	**91.11** ↑ 2.34	**80.96** ↑ 3.90	69.34 ↓ 1.11	**82.90** ↑ 6.96

Implementation Details **Training.** We adopt **Qwen2.5-VL-7B** [5] as the base model. All RL pipelines are implemented via the `EasyR1` framework [54, 41] for fair comparison. During GRPO, we utilize a normalization group size of $N = 8$. Uniquely, our method samples $N - 1 = 7$ online candidate responses and injects 1 constructed anchored response for isolated advantage estimation. We set the number of appended negative anchors to $M = 2$. The reward weights are balanced at $\lambda_{\text{rec}} = 0.5$ and $\lambda_{\text{pre}} = 0.5$. **Inference and Evaluation.** At inference, each model receives a test image and a prompt combining a recognition question with an output-format instruction. Predictions are parsed according to the specified format, with List-GRPO generating a ranked candidate list in a single response. We normalize category names by lowercase and removing non-alphanumeric characters. A predicted category is considered correct if its normalized string contains the normalized ground-truth label following ViRFT [23]. For List-GRPO, Top-1 accuracy evaluates the first candidate, while recall checks whether any candidate matches the ground truth. Additional experimental settings are provided in Appendix A.

4.2 Main Results

Table 1 summarizes the performance of our method compared with other baselines across five fine-grained visual classification datasets under an open-ended question answering setting. Across all training regimes, our method consistently outperforms standard supervised fine-tuning (SFT), No-Thinking RL, ViRFT [23], and CLS-RL [18], demonstrating its effectiveness in leveraging limited supervision and handling visual ambiguity. Under the limited 1-shot setting, our approach achieves an average accuracy of 73.26%, yielding a substantial absolute gain of 5.49% over the strongest baseline (CLS-RL at 67.77%). Furthermore, as the number of samples per class increases, the overall performance gap further widens. The average accuracy improvement scales from +5.49% in the 1-shot setting to +6.55% (79.59% average) in the 2-shot setting, and reaches +6.96% (82.90% average) in the 4-shot regime. These results underscore the robustness and scalability of our ap-

proach in effectively utilizing limited data to enhance fine-grained visual classification performance, successfully navigating the inherent visual ambiguities among subordinate categories.

Table 2: Ablation study on the core components of our framework under the 1-shot setting. The best results in each row are highlighted in bold.

Method	Flowers	Birds	Pets	Dogs	Aircrafts	Average
No-Thinking RL (Single Answer)	64.76	56.83	80.59	65.76	65.02	66.59
+ Uncertainty-Aware List	65.37	64.05	**85.25**	72.74	66.94	70.87
+ Ground-Truth Anchor	66.46	60.10	80.76	72.98	67.51	69.60
Full Model (List + Anchor)	**67.80**	67.07	83.81	**77.32**	**70.29**	**73.26**
Full Model w/o Isolated	66.63	**67.88**	83.15	75.02	69.19	72.37

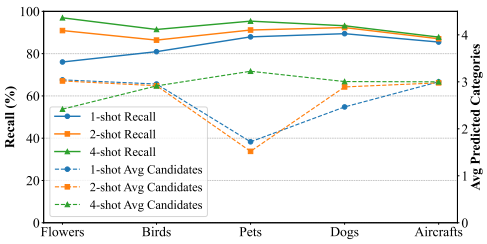

Figure 3: Recall and average predicted candidate categories of our method under varying training data sizes.

Table 3: Ablation on GRPO rollout size (N) using Qwen2.5-VL-7B (1-shot setting). Doubling the number of rollouts for the No-Thinking baseline ($N = 16$) yields only marginal gains and still falls significantly short of our method at $N = 8$.

Method	N	Birds	Dogs	Aircrafts
No-Thinking RL	8	56.83	65.76	65.02
No-Thinking RL	16	59.22	67.26	65.47
Ours w/o Anchor	8	64.05	72.74	66.94
Ours	8	**67.07**	**77.32**	**70.29**

Analysis of Adaptive Candidate Lists To further understand the impact of our uncertainty-aware list prediction mechanism, we analyze both the recall and the average number of predicted candidate categories of our method on the test sets. The recall is computed as the percentage of test samples for which the ground-truth category is included in the predicted candidate list, while the average number of predicted candidates is the mean length of the candidate lists generated by the model. As shown in Figure 3, our method predicts 2-3 candidate categories on average, while achieving a consistently high recall across all datasets. For instance, in the 4-shot setting, our method achieves 96.95% recall on Oxford Flowers with an average of 2.42 predicted candidates, and 91.43% recall on CUB Birds with an average of 2.91 predicted candidates. This confirms that our framework does not blindly inflate the candidate list but instead intelligently bounds the semantic confusion set to balance uncertainty with precision. By effectively capturing multiple plausible categories under visual ambiguity, our approach enhances overall system robustness and yields a compact, highly relevant candidate pool ideal for downstream expert verification or automated disambiguation.

4.3 Ablation Studies

Effect of Core Components To evaluate the individual contributions of our proposed components, we conduct an ablation study under the 1-shot setting. As shown in Table 2, starting from a baseline that generates a single answer (equivalent to No-Thinking RL), transitioning from a single-answer baseline to the Uncertainty-Aware List provides +4.28% average boost, validating our premise that allowing the model to express uncertainty is critical for fine-grained learning. Moreover, integrating our Ground-Truth Anchor Construction (using the ground truth as the constructed response) and Isolated Advantage Estimation mechanism into the No-Thinking framework improves the average accuracy by +3.01%, confirming that our anchored reinforcement learning design effectively guides the model towards more accurate predictions. Finally, the full model that combines both components achieves the best performance, demonstrating that the uncertainty-aware list formulation and ground-truth anchored learning synergistically enhance the model's ability to handle visual ambiguity and learn from limited data. Without isolated advantage estimation ("Full Model w/o Isolated"), the performance drops by -1.17% on Oxford Flowers and -2.30% on Stanford Dogs, indicating that this component is necessary to prevent the synthetic response from artificially distorting on-policy preference signals.

Table 4: Ablation on the number of hard-negative categories appended to the ground-truth anchor. "Append M" denotes an anchor response constructed using the ground truth alongside the top-M most frequent confusions.

Append M	Dogs		Birds	
	Accuracy	Recall	Accuracy	Recall
0 (GT Only)	76.26	80.99	64.03	64.03
2	**77.32**	89.44	**67.07**	80.93
4	76.35	**91.14**	66.24	**84.88**

Table 5: Base-to-Novel generalization performance (Top-1 Accuracy %). Models are fine-tuned on the Base classes under a 4-shot setting and evaluated on the unseen Novel classes.

Method	Birds	Dogs
Qwen2.5-VL-7B	44.99	53.50
No-Thinking RL	47.16	52.32
Ours	**55.88**	**63.88**

Table 6: Fine-grained classification performance (%) of Qwen2.5-VL-3B across five datasets under 1-shot setting. The best results in each row are highlighted in bold.

Method	Pets	Flowers	Dogs	Aircraft	Birds	Average
No-Thinking	72.47	54.85	62.00	50.71	48.15	57.64
ViRFT [23]	69.42	52.13	51.19	52.63	52.71	55.62
CLS-RL [18]	74.00	59.72	66.85	54.91	57.15	62.53
Ours	**76.04**	**62.72**	**70.99**	**56.86**	**60.39**	**65.40**
Ours-Recall	82.75	69.87	85.69	74.44	73.94	77.34

Effect of Appended Confusing Categories We further investigate the influence of the anchor's composition by varying the number of appended hard-negative categories (M). As reported in Table 4, relying solely on the ground-truth class (Append 0) yields suboptimal results. Without any contrasting context, this setup inadvertently trains the model to output a single label, thus limiting its ability to provide listwise supervision. Conversely, appending the top-2 most frequent confusions (Append 2) leads to a marked increase in both Top-1 accuracy and candidate recall (e.g., recall jumps from 64.03% to 80.93% on CUB200). This confirms that explicitly contrasting the target class with the model's current hard negatives effectively sharpens the local decision boundary. However, further expanding the anchor to include four negatives introduces a trade-off: while candidate recall reaches its peak, Top-1 accuracy begins to degrade. This indicates that an overly broad set of distractors introduces noise, which dilutes the learning signal needed for precise top-1 predictions.

Effect of Reward Components. To investigate the individual contributions of precision and recall rewards, we conduct an ablation study on the Uncertainty-Aware List (UA-list) variant, which generates ranked candidate lists without dynamic anchoring, using Qwen2.5-VL-7B with 1-shot setting on the CUB200 and Dogs datasets. Table 7 shows that both single-reward variants outperform No-Thinking RL, while their combination achieves the highest accuracy on both datasets, demonstrating that precision and recall provide complementary supervision.

Effect of List-GRPO on image representations. We further investigate whether List-GRPO improves visual representations beyond merely optimizing category generation and ranking. Following Fine-R1 [9], we mean-pool final-layer image-token hidden states into 3,584-dimensional features. We conduct 4-shot and full-shot linear probing on Qwen2.5-VL-7B before and after List-GRPO, evaluating both on the full test sets. The improvements in Table 8 suggest enhanced fine-grained feature separability. Further details and retrieval results are provided in Appendix A.7.

Effect of Vision-Encoder Updates. We compare full-model fine-tuning with a variant that freezes the vision encoder under the 1-shot setting. The results in Table 9 support the benefit of updating the vision encoder during fine-grained adaptation.

Analysis of Rollout Number To assess whether the performance gains of our method are primarily driven by the uncertainty-aware list formulation or simply by increasing the number of rollouts (i.e., candidate responses) during GRPO, we conduct an ablation study on the rollout size (N) for the No-Thinking baseline. As shown in Table 3, doubling the number of rollouts from 8 to 16 for the No-Thinking baseline yields only marginal improvements (e.g., +2.39% on Birds and +1.50% on Dogs), and still falls significantly short of our method's performance at $N = 8$ ('ours w/o Anchor'

Table 7: Reward ablation (1-shot). Results are Top-1 accuracy (%).

Setting	Birds	Dogs
No-Thinking RL	56.8	65.8
UA-list (precision)	62.6	69.8
UA-list (recall)	60.7	69.2
UA-list (both)	**64.1**	**72.7**

Table 8: Linear probing of frozen image-token features. Results are peak test accuracy (%).

Probe	Model	Birds	Dogs	Mean
4-shot	Base	87.26	69.14	78.20
	List-GRPO	**88.77**	**73.30**	**81.04**
Full-shot	Base	92.91	88.85	90.88
	List-GRPO	**94.38**	**89.78**	**92.08**

Table 9: Vision-encoder ablation of List-GRPO with Qwen2.5-VL-7B under 1-shot training. Results are Top-1 accuracy (%) on the full test sets.

Training configuration	Birds	Dogs
Frozen vision encoder	64.9	73.5
Full-model fine-tuning	**67.1**	**77.3**

means only using the uncertainty-aware list without constructing the ground-truth anchored response). This demonstrates that the performance improvements of List-GRPO are fundamentally driven by its uncertainty-aware list formulation, rather than merely scaling up exploration compute.

Generalization Analysis To verify that our framework learns transferable discriminative skills, we evaluate it under a Base-to-Novel setting. We partition the classes into a disjoint Base set (50%) and Novel set (50%). Under the 4-shot condition, models are optimized exclusively on the Base classes and evaluated zero-shot on the unseen Novel classes. As shown in Table 5, standard exact-match RL (No-Thinking RL) struggles significantly with out-of-distribution generalization. In contrast, our List-GRPO framework achieves +8.72% and +11.56% gains over No-Thinking RL on CUB-200 and Stanford Dogs, respectively, demonstrating that our method not only excels at recognizing fine-tuned categories but also preserves the model's zero-shot robustness on novel fine-grained concepts.

Backbone Scalability Table 6 reports the 1-shot performance of Qwen25-VL-3B [5] across five FGVC datasets. Our method achieves substantial improvements over other RL baselines in terms of top-1 accuracy. Additionally, the corresponding recall values ('Ours-Recall') indicate that the model consistently includes the ground-truth class in its candidate list. These results confirm that our framework generalizes beyond a specific backbone and can effectively enhance fine-grained classification performance across different model scales.

5 Conclusion and Limitation

We propose List-GRPO, an uncertainty-aware listwise GRPO framework for fine-grained visual classification. By generating uncertainty-aware candidate lists and leveraging anchored reinforcement signals, our method improves both top-1 accuracy and recall, providing robust coverage for visually confusable categories. The "uncertainty-aware" refers to response-level ambiguity handling, where the model expresses multiple plausible fine-grained hypotheses through a ranked candidate list. This formulation does not perform calibrated uncertainty estimation or explicitly model probabilistic confidence. Evaluating calibration and selective prediction therefore remains an important direction for future work. In addition, our approach follows the No-Thinking paradigm, generating candidates in a single response without explicit intermediate reasoning. Incorporating multi-step or internal reasoning to iteratively refine candidate lists may further improve accuracy and interpretability.

Acknowledgments and Disclosure of Funding

This work was supported by the grants from the National Natural Science Foundation of China (62432001, 62525201, 62132001) and Beijing Natural Science Foundation (L247006, L257005).

References

[1] X. An, Y. Xie, K. Yang, W. Zhang, X. Zhao, Z. Cheng, Y. Wang, S. Xu, C. Chen, C. Wu, H. Tan, C. Li, J. Yang, J. Yu, X. Wang, B. Qin, Y. Wang, Z. Yan, Z. Feng, Z. Liu, B. Li, and J. Deng. Llava-onevision-1.5: Fully open framework for democratized multimodal training. In *arXiv*, 2025.

[2] S. Angheben, D. Berasi, A. Conti, E. Ricci, and Y. Wang. Specificity-aware reinforcement learning for fine-grained open-world classification. *arXiv preprint arXiv:2603.03197*, 2026.

[3] J. Bai, S. Bai, S. Yang, S. Wang, S. Tan, P. Wang, J. Lin, C. Zhou, and J. Zhou. Qwen-VL: A Versatile Vision-Language Model for Understanding, Localization, Text Reading, and Beyond. *arXiv preprint arXiv:2308.12966*, 2023.

[4] S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge, W. Ge, Z. Guo, Q. Huang, J. Huang, F. Huang, B. Hui, S. Jiang, Z. Li, M. Li, M. Li, K. Li, Z. Lin, J. Lin, X. Liu, J. Liu, C. Liu, Y. Liu, D. Liu, S. Liu, D. Lu, R. Luo, C. Lv, R. Men, L. Meng, X. Ren, X. Ren, S. Song, Y. Sun, J. Tang, J. Tu, J. Wan, P. Wang, P. Wang, Q. Wang, Y. Wang, T. Xie, Y. Xu, H. Xu, J. Xu, Z. Yang, M. Yang, J. Yang, A. Yang, B. Yu, F. Zhang, H. Zhang, X. Zhang, B. Zheng, H. Zhong, J. Zhou, F. Zhou, J. Zhou, Y. Zhu, and K. Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[5] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, H. Zhong, Y. Zhu, M. Yang, Z. Li, J. Wan, P. Wang, W. Ding, Z. Fu, Y. Xu, J. Ye, X. Zhang, T. Xie, Z. Cheng, H. Zhang, Z. Yang, H. Xu, and J. Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[6] Z. Chen, J. Wu, W. Wang, W. Su, G. Chen, S. Xing, M. Zhong, Q. Zhang, X. Zhu, L. Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198, 2024.

[7] Y. Deng, H. Bansal, F. Yin, N. Peng, W. Wang, and K.-W. Chang. Openvlthinker: An early exploration to complex vision-language reasoning via iterative self-improvement. *ArXiv*, abs/2503.17352, 2025.

[8] A. Dubey, O. Gupta, P. Guo, R. Raskar, R. Farrell, and N. Naik. Pairwise confusion for fine-grained visual classification. In *Proceedings of the European conference on computer vision (ECCV)*, pages 70–86, 2018.

[9] H. He, Z. Geng, and Y. Peng. Fine-r1: Make multi-modal llms excel in fine-grained visual recognition by chain-of-thought reasoning. *arXiv preprint arXiv:2602.07605*, 2026.

[10] H. He, G. Li, Z. Geng, J. Xu, and Y. Peng. Analyzing and boosting the power of fine-grained visual recognition for multi-modal large language models. *arXiv preprint arXiv:2501.15140*, 2025.

[11] H. He, Z. Tan, and Y. Peng. Taxonomy-aware representation alignment for hierarchical visual recognition with large multimodal models. *arXiv preprint arXiv:2603.00431*, 2026.

[12] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[13] S. Kadavath, T. Conerly, A. Askell, T. Henighan, D. Drain, E. Perez, N. Schiefer, Z. Hatfield-Dodds, N. DasSarma, E. Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

[14] A. Khosla, N. Jayadevaprakash, B.-P. Yao, and L. Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011.

[15] J. Kim and H. Ji. Finer: Investigating and enhancing fine-grained visual concept recognition in large vision language models. *arXiv preprint arXiv:2402.16315*, 2024.

[16] R. Kumar, A. Sadhu, and R. Nevatia. Dive-k: Differential visual reasoning for fine-grained image recognition. *arXiv preprint arXiv:2511.18305*, 2025.

[17] L. Lawrence, O. Saha, M. Wei, C. Sun, S. Maji, and G. Van Horn. You may speak freely: Improving the fine-grained visual recognition capabilities of multimodal large language models with answer extraction. *arXiv preprint arXiv:2510.14885*, 2025.

[18] M. Li, J. Zhong, S. Zhao, Y. Lai, H. Zhang, W. B. Zhu, and K. Zhang. To think or not to think: A study of thinking in rule-based visual reinforcement fine-tuning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[19] Y. Li, J. Cheng, S. Jia, H. Kuang, S. Jiao, Q. Hou, and M.-M. Cheng. Tempsamp-r1: Effective temporal sampling with reinforcement fine-tuning for video llms. *arXiv preprint arXiv:2509.18056*, 2025.

[20] X. Linhui, C. Guiping, G. Mingyue, G. Xianchao, Y. Fan, T. Ming, L. Xin, P. Yuxin, and W. Yaowei. A survey on the green development of large models: From resource-efficient architectures to hardware–software co-design. *Chinese Journal of Electronics*, 35(5):1817–1840, 2026.

[21] H. Liu, C. Li, Y. Li, B. Li, Y. Zhang, S. Shen, and Y. J. Lee. LLaVA-NeXT: Improved reasoning, ocr, and world knowledge, January 2024.

[22] H. Liu, C. Li, Q. Wu, and Y. J. Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

[23] Z. Liu, Z. Sun, Y. Zang, X. Dong, Y. Cao, H. Duan, D. Lin, and J. Wang. Visual-rft: Visual reinforcement fine-tuning. *arXiv preprint arXiv:2503.01785*, 2025.

[24] P. Lu, H. Bansal, T. Xia, J. Liu, C. Li, H. Hajishirzi, H. Cheng, K.-W. Chang, M. Galley, and J. Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023.

[25] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[26] A. Masry, X. L. Do, J. Q. Tan, S. Joty, and E. Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the association for computational linguistics: ACL 2022*, pages 2263–2279, 2022.

[27] M. Mathew, V. Bagal, R. Tito, D. Karatzas, E. Valveny, and C. Jawahar. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1697–1706, 2022.

[28] M. Mathew, D. Karatzas, and C. Jawahar. DocVQA: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021.

[29] B. McKinzie, Z. Gan, J.-P. Fauconnier, S. Dodge, B. Zhang, P. Dufter, D. Shah, X. Du, F. Peng, A. Belyi, et al. Mm1: methods, analysis and insights from multimodal llm pre-training. In *European Conference on Computer Vision*, pages 304–323. Springer, 2025.

[30] F. Meng, L. Du, Z. Liu, Z. Zhou, Q. Lu, D. Fu, T. Han, B. Shi, W. Wang, J. He, et al. Mm-eureka: Exploring the frontiers of multimodal reasoning with rule-based reinforcement learning. *arXiv preprint arXiv:2503.07365*, 2025.

[31] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008.

[32] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[33] C. Pang, H. Yu, Z. Chen, L. Lu, and X. Lou. Towards fine-grained recognition with large visual language models: Benchmark and optimization strategies. *arXiv preprint arXiv:2512.10384*, 2025.

[34] J. Park, J. Na, J. Kim, and H. J. Kim. Deepvideo-r1: Video reinforcement fine-tuning via difficulty-aware regressive grpo. *arXiv preprint arXiv:2506.07464*, 2025.

[35] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

[36] Y. Peng, Z. Wang, G. Li, X. Zheng, S. Yin, and H. He. A survey on fine-grained multimodal large language models. *Chinese Journal of Electronics*, 35(2):771–803, 2026.

[37] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

[38] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[39] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. Li, Y. Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[40] H. Shen, P. Liu, J. Li, C. Fang, Y. Ma, J. Liao, Q. li Shen, Z. Zhang, K. Zhao, Q. Zhang, R. Xu, and T. Zhao. Vlm-r1: A stable and generalizable r1-style large vision-language model. *ArXiv*, abs/2504.07615, 2025.

[41] G. Sheng, C. Zhang, Z. Ye, X. Wu, W. Zhang, R. Zhang, Y. Peng, H. Lin, and C. Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

[42] N. Stiennon, L. Ouyang, J. Wu, D. M. Ziegler, R. J. Lowe, C. Voss, A. Radford, D. Amodei, and P. Christiano. Learning to summarize from human feedback. *ArXiv*, abs/2009.01325, 2020.

[43] G. Team. Gemma 3 technical report, 2025.

[44] G. Team, P. Georgiev, V. I. Lei, R. Burnell, L. Bai, A. Gulati, B. O. Batsaikhan, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

[45] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. *Tech. Report CNS-TR-2011-001*, 2011.

[46] Y. Wang, Y. Lu, L. Meng, X. Li, F. Lin, and X. Meng. Generative text-to-video retrieval via a chain-of-thought reasoning paradigm. *Chinese Journal of Electronics*, 35(6):2591–2603, 2026.

[47] X.-S. Wei, Y.-Z. Song, O. Mac Aodha, J. Wu, Y. Peng, J. Tang, J. Yang, and S. Belongie. Fine-grained image analysis with deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(12):8927–8948, 2021.

[48] X. Wen, Z. Liu, S. Zheng, S. Ye, Z. Wu, Y. Wang, Z. Xu, X. Liang, J. Li, Z. Miao, et al. Reinforcement learning with verifiable rewards implicitly incentivizes correct reasoning in base llms. *arXiv preprint arXiv:2506.14245*, 2025.

[49] H. Wu, Z. Zhang, E. Zhang, C. Chen, L. Liao, A. Wang, C. Li, W. Sun, Q. Yan, G. Zhai, et al. Q-bench: A benchmark for general-purpose foundation models on low-level vision. *arXiv preprint arXiv:2309.14181*, 2023.

[50] W. Wu, C. Gao, J. Chen, K. Qinghong Lin, Q. Meng, Y. Zhang, Y. Qiu, H. Zhou, and M. Z. Shou. Reinforcement learning in vision: A survey. *arXiv e-prints*, pages arXiv–2508, 2025.

[51] Q. Yu, X. Shen, and L.-C. Chen. Towards open-ended visual recognition with large language models. In *European Conference on Computer Vision*, pages 359–376. Springer, 2024.

[52] Y. Zhang, A. Unell, X. Wang, D. Ghosh, Y. Su, L. Schmidt, and S. Yeung-Levy. Why are visually-grounded language models bad at image classification? *Advances in Neural Information Processing Systems*, 37:51727–51753, 2024.

[53] Z. Zhao, H. Lu, Y. Huo, Y. Du, T. Yue, L. Guo, B. Wang, W. Chen, and J. Liu. Needle in a video haystack: A scalable synthetic framework for benchmarking video mllms. *arXiv preprint*, 2024.

[54] Y. Zheng, J. Lu, S. Wang, Z. Feng, D. Kuang, and Y. Xiong. Easyr1: An efficient, scalable, multi-modality rl training framework, 2025.

[55] R. Zhou, Y. Shao, H. Lu, B. Xing, T. Bai, Y. Chen, J. Zhao, L. Sui, H. Yao, Z. Zhao, H. Yang, H. Wu, Z. Zhou, J. Zhu, Z. Huang, Y. Bao, Y. Liu, Y. Charles, and X. Zhou. Worldvqa: Measuring atomic world knowledge in multimodal large language models, 2026.

[56] J. Zhu, W. Wang, Z. Chen, Z. Liu, S. Ye, L. Gu, H. Tian, Y. Duan, W. Su, J. Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

[57] D. M. Ziegler, N. Stiennon, J. Wu, T. B. Brown, A. Radford, D. Amodei, P. Christiano, and G. Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

A Experimental Details

A.1 Dataset Statistics

We evaluate our framework across five widely adopted fine-grained visual classification (FGVC) benchmarks. The core statistics for these datasets are detailed in Table 10. To simulate real-world low-resource scenarios, we construct few-shot training splits by randomly sampling $k \in \{1, 2, 4\}$ instances per class from the original training sets. All evaluations are performed on the full, standard test sets.

Table 10: Detailed statistics of the fine-grained visual classification benchmarks.

Dataset	Number of Classes	Test Set Size
CUB-200-2011	200	5,794
Stanford Dogs	120	8,580
Oxford Flowers	102	2,463
FGVC-Aircraft	100	3,333
Oxford-IIIT Pet	37	3,669

A.2 Implementation Details

Prompt Design. For all RL fine-tuning experiments, the prompt templates comprise two components: (1) a *base prompt* that explicitly instructs the model to address the recognition task, and (2) a *format prompt* that strictly constrains the output structure. Following Nlg2choice [17], we adopt multiple semantically equivalent base prompt variants for each dataset to encourage robust instruction following, as listed in Table 11.

The placeholder `{type}` is replaced with dataset-specific terms (e.g., "species" for CUB-200, "variant" for FGVC-Aircraft), while `{domain}` is replaced with the corresponding broader category (e.g., "bird", "aircraft").

For the format prompt, we utilize the following constraints depending on the baseline methods:

- **No-Thinking RL:** *"Answer only the {type} name, no other text."*
- **ViRFT:** We directly adopt the original instructional prompt from ViRFT [23], which requires a strict tag format: *"Output the thinking process in <think>...</think> and final answer in <answer>...</answer> tags. The output answer format should be as follows: <think>...</think> <answer>{type} name</answer>. Please strictly follow the format."*

Table 11: Base prompt variants for each dataset. `{type}` and `{domain}` are dynamically replaced with dataset-specific terms during training and evaluation.

No.	Prompt Variant
1	What {type} is this {domain}?
2	What is the {type} of this {domain}?
3	What is the {type} of the {domain}?
4	What is the {type} of the {domain} in this image?
5	What is the {type} of the {domain} in the image?
6	Identify this {domain}'s {type}.
7	Name the {type} shown in the image.
8	Which {domain} {type} is pictured here?
9	Classify the {type} of this {domain}.
10	What {domain} {type} does the photo depict?
11	Determine the {type} of the {domain} in view.
12	Provide the common name of this {domain}.
13	To which {type} does this {domain} belong?
14	Label the {type} of the {domain} shown.
15	Recognize and state this {domain}'s {type}.

- **CLS-RL:** We adopt the original prompt from CLS-RL [18]: *"Please first identify whether this problem requires intermediate thinking or calculation. If the problem requires thinking or calculation, output the thinking and calculation process inside <think>...</think> tags and the final answer inside <answer>...</answer> tags. If no thinking or calculation is required, directly output the final answer inside <answer>...</answer> tags. Your output should follow one of two cases: (1) <answer> {type} name </answer>, (2) <think>...</think> <answer> {type} name </answer>."*

- **Ours:** *"Output an adaptive number of possible species names according to your uncertainty, ranked from most likely to least likely, inside <answer>...</answer> tags. Use the following format only: <answer> {type} name 1, ... </answer>"*

Training Details. Our framework is implemented using the `easyr1` codebase, an efficient and scalable multimodal RL training framework built upon the `veRL` project. We utilize the Qwen2.5-VL-7B-Instruct [5] as our foundational model. During GRPO training, we employ a rollout group size of $G = 8$ and regularize the policy using a KL penalty coefficient of $\beta = 0.01$. Specifically, we sample 7 online responses per prompt during the rollout phase and synthetically construct 1 anchor response by prepending the ground-truth category to the top-2 most frequently occurring hard negatives (with frequency no less than 2) mined from the current online group. The $L_{\max}$ is set to 6. The generation temperature is set to 1.0 with top-p at 1.0. The policy is optimized via AdamW with a learning rate of 1×10^{-6} and a weight decay of 0.01. Following ViRFT [23], all experiments are trained under 4 epochs and executed on 8 NVIDIA A6000 GPUs (48G). In our experiments, the few-shot samples are randomly selected from the training set using a random seed, without any manual selection. All methods are trained on the same sampled few-shot sets and evaluated on the test set, ensuring a fair comparison under each setting.

ViRFT Implementation. To guarantee a fair and rigorous comparison, we meticulously reproduce the ViRFT baseline within our `easyr1` pipeline, applying identical hyperparameters. Following DiVE-k [16], we rectify a reward verification logic present in the original ViRFT implementation. The original ViRFT implementation utilizes a bidirectional substring matching criterion (i.e., triggering a positive reward if either `ground_truth in prediction` or `prediction in ground_truth`). As demonstrated in DiVE-k, this bidirectional logic creates algorithmic shortcuts; for example, generating the broad term "gull" would incorrectly yield a positive reward for the specific target "California Gull". To eliminate this metric inflation, we enforce a strict unidirectional string-matching rule, requiring the exact ground-truth string to be present within the model's prediction.

A.3 Visualization of Constructed Responses

Our framework introduces a dynamic ground-truth anchored response strategy to mitigate advantage collapse when online exploration completely misses the target. In Table 12, we visualize examples of these constructed off-policy anchors across different datasets during training.

Typically, an anchored response is synthesized by explicitly placing the ground-truth category before the most frequently occurring hard negatives mined from the current rollout group. For instance, if the ground truth is "California Gull" but the policy confidently predicts "Ring-billed Gull" and "Herring Gull", the synthesized listwise anchor explicitly contrasts the truth against these specific confusions. These examples demonstrate that the model actively navigates genuine local confusion sets. By incorporating these off-policy anchored responses, our algorithm effectively captures the evolving confusion sets and provides dense training signals throughout different stages of learning.

Table 12: Visualization of constructed off-policy anchored responses during training. Each response is synthesized by prepending the ground-truth category to the most frequent hard negatives mined from the online rollout group.

Dataset	Ground Truth	Constructed Anchor Response
CUB-200-2011	Lazuli Bunting	`<answer> Lazuli Bunting, Indigo Bunting, Painted Bunting </answer>`
	Canada Warbler	`<answer> Canada Warbler, Magnolia Warbler, Louisiana Warbler </answer>`
	Horned Puffin	`<answer> Horned Puffin, Tufted Puffin </answer>`
	Nashville Warbler	`<answer> Nashville Warbler, Tennessee Warbler </answer>`
Stanford Dogs	Tibetan Terrier	`<answer> Tibetan Terrier, Havanese, Shih Tzu </answer>`
	Eskimo Dog	`<answer> Eskimo Dog, Siberian Husky, Alaskan Malamute </answer>`
	Silky Terrier	`<answer> Silky Terrier, Yorkshire Terrier </answer>`
	Sussex Spaniel	`<answer> Sussex Spaniel, English Cocker Spaniel </answer>`
Oxford Flowers	Gazania	`<answer> Gazania, Sun flower </answer>`
	Californian poppy	`<answer> Californian poppy, Mexican poppy </answer>`
	Globe-flower	`<answer> Globe-flower, Buttercup </answer>`
	Tree poppy	`<answer> Tree poppy, Iceland Poppy </answer>`
FGVC-Aircraft	Saab 340	`<answer> Saab 340, Saab 2000 </answer>`
	737-800	`<answer> 737-800, 737-900, 737-400 </answer>`
	Tu-134	`<answer> Tu-134, Tu-154, Yak-42 </answer>`
	CRJ-700	`<answer> CRJ-700, CRJ-900 </answer>`
Oxford-IIIT Pet	Russian blue	`<answer> Russian blue, Domestic Cat </answer>`
	Shiba inu	`<answer> Shiba inu, Husky </answer>`
	Egyptian mau	`<answer> Egyptian mau, Domestic Cat, African Wildcat </answer>`
	Boxer	`<answer> Boxer, Boston Terrier, American Bulldog </answer>`

Table 13: Detailed performance metrics across various few-shot settings. "Avg. Cand." denotes the average number of categories explicitly generated within the candidate list.

Setting	Metric	Flowers	Birds	Pets	Dogs	Aircrafts	Average
1-shot	Precision (%)	67.80	67.07	83.81	77.32	70.29	73.26
	Recall (%)	76.05	80.93	87.93	89.44	85.48	83.97
	Avg. Cand.	3.04	2.95	1.72	2.47	3.00	2.64
2-shot	Precision (%)	86.76	73.92	88.12	79.86	69.27	79.59
	Recall (%)	90.91	86.40	91.17	92.33	87.04	89.57
	Avg. Cand.	3.02	2.92	1.52	2.89	2.98	2.67
4-shot	Precision (%)	95.37	77.70	91.11	80.96	69.34	82.90
	Recall (%)	96.95	91.43	95.37	93.19	87.82	92.95
	Avg. Cand.	2.42	2.91	3.22	3.01	3.00	2.91

A.4 Detailed Performance

To further analyze the behavior of List-GRPO under various data constraints, we report the detailed Precision, Recall, and the average number of predicted candidates across all five benchmarks for $k \in \{1, 2, 4\}$. As shown in Table 13, our framework maintains a high recall-to-precision ratio while adaptively adjusting the candidate list size to capture uncertainty appropriately, achieving high recall even when outputting only around three candidate classes.

Table 14: Novel-class Top-1 accuracy (%) under 1-shot base-to-novel adaptation. The best results are highlighted in bold.

Method	Flowers	Birds	Pets	Dogs	Aircraft
Qwen2.5-VL-7B	46.0	51.3	59.6	58.7	31.4
+ ViRFT	59.2	54.1	62.9	59.4	30.5
+ List-GRPO	**69.7**	**62.3**	**71.9**	**61.9**	**52.6**

Table 15: Top-1 accuracy (%) of additional backbones on randomly sampled 40% category subsets of six datasets. List-GRPO uses 1-shot training. The best results for each backbone are highlighted in bold.

Model	Flowers	Birds	Pets	Dogs	Aircraft	Cars
Gemma3-4B-IT	47.9	16.1	54.3	33.4	16.1	53.2
Gemma3-4B-IT + List-GRPO	**67.2**	**36.1**	**85.4**	**58.1**	**41.7**	**79.6**
Qwen3-VL-4B	46.7	38.8	65.8	53.9	30.3	69.0
Qwen3-VL-4B + List-GRPO	**66.4**	**69.0**	**85.1**	**74.5**	**50.1**	**87.9**

Notably, even in the 1-shot setting, List-GRPO achieves substantial recall across all benchmarks while keeping the average number of predicted candidates relatively low (around 2–3), demonstrating that the model effectively prioritizes the most likely classes without excessively expanding the candidate list. As the number of shots increases, the precision improves steadily while the recall remains high, indicating that additional data helps the model resolve ambiguity and refine top-ranked predictions.

A.5 Additional Base-to-Novel Generalization

We further evaluate base-to-novel generalization on five fine-grained classification datasets. For each dataset, we randomly select 40% categories with a 1:1 base-novel split. Using Qwen2.5-VL-7B as the backbone, we fine-tune with one labeled image per base class and evaluate on novel classes that are unseen during fine-tuning. The results in Table 14 support List-GRPO's ability to generalize to fine-grained categories unseen during fine-tuning.

A.6 Additional Backbone Experiments

We further evaluate List-GRPO using Gemma3-4B-IT [43] and Qwen3-VL-4B [4]. For each dataset, we randomly select 40% of the categories, use 1-shot training image per selected category for adaptation, and evaluate on test images from the selected categories. As shown in Table 15, List-GRPO consistently improves recognition accuracy for both backbones across all six datasets, supporting its applicability beyond Qwen2.5-VL.

A.7 Additional Feature Evaluation

Linear-Probe Settings. We freeze each checkpoint and train a linear classifier using either four images per class or the full training set. The base and adapted checkpoints use identical classes, data splits, probe architectures, and optimization settings. Each probe is trained for 5,000 epochs with Adam, a learning rate of 10^{-4}, and a batch size of 512.

Nearest-Neighbor Retrieval. We further evaluate the image representations using leave-one-out kNN retrieval. For each test image, we extract its mean-pooled representation and retrieve the nearest neighbors from the remaining test images based on feature similarity. Each test image queries all remaining test images, excluding itself. Hit@k measures the percentage of queries with at least one same-class image among the k nearest neighbors. As shown in Table 16, List-GRPO improves both Hit@1 and Hit@5 on CUB200 and Dogs datasets, indicating more class-consistent local feature neighborhoods.

Table 16: Leave-one-out retrieval using frozen image-token features. Results are Hit@k (%) with all other test images from the same dataset forming the gallery.

Metric	Model	CUB	Dogs	Mean
Hit@1	Base	77.84	59.36	68.60
	List-GRPO	**80.43**	**64.13**	**72.28**
Hit@5	Base	93.87	81.16	87.52
	List-GRPO	**94.99**	**84.37**	**89.68**



Figure 4: Qualitative results across various fine-grained visual classification benchmarks. Subtle distinctions among sub-categories highlight the limitations of rigid single-label exact-match paradigms and demonstrate the necessity of our uncertainty-aware listwise prediction framework.

A.8 Qualitative Results

To qualitatively evaluate our trained List-GRPO model, we visualize examples from the test sets across different fine-grained benchmarks in Figure 4.

The results illustrate that subordinate categories within the same overarching family often exhibit extremely subtle inter-class differences. For example, distinguishing between aviation variants (e.g., Boeing 737-200 vs. 737-300), specific bird species (e.g., Cape May Warbler vs. Prairie Warbler), or closely related pet breeds (e.g., Siamese vs. Tonkinese cats) remains highly challenging due to severe visual ambiguity.

Standard exact-match RL methods tend to collapse this ambiguity into a single, overconfident prediction, penalizing near-misses that are semantically reasonable. In contrast, our List-GRPO model produces a ranked candidate list, effectively capturing the predictive uncertainty in these local confusion sets. This enables robust recognition of fine-grained categories without being overly penalized for not selecting the exact ground truth.

These qualitative results demonstrate that List-GRPO can successfully capture local confusion patterns, and maintain high recall even when outputting only a few predictions, highlighting the practical effectiveness of our uncertainty-aware listwise prediction framework.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the contributions of the List-GRPO framework.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our approach in the Conclusion and Limitation section.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: [N/A]

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We clearly define our listwise reward formulations, anchor construction strategies, and advantage estimation logic in the main text. Complete hyperparameter settings and implementation details are provided in the Appendix. Researchers can reproduce our results with the provided information.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We attach our code as supplementary material.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental settings, including the few-shot sampling protocol ($k \in \{1, 2, 4\}$) and dataset details, are described in Section 4. Full fine-tuning details (e.g., learning rates, optimizer) are in the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to the substantial computational cost of language-model experiments, we report results from a single run. We use fixed random seeds for both few-shot training and evaluation to ensure reproducibility. In our experiments, both our method and the baseline methods exhibit stable performance.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We describe the compute resources we use for training and inference in Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This research complies with the NeurIPS Code of Ethics. It does not include human subjects. The datasets are publicly available and do not have ethic concerns. The study develops algorithms tested on standard, publicly available academic benchmarks without sensitive or harmful content.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

Justification: This work focuses on a general reinforcement learning framework for fine-grained visual classification using standard academic datasets. It is foundational research and does not have direct, specific foreseeable negative applications.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: [N/A]

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All datasets (CUB-200, Stanford Dogs, etc.) and the pre-trained MLLM (Qwen2.5-VL) utilized in this work are properly cited.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: We explicitly outline the usage of the Qwen2.5-VL-7B model, which serves as the core Multimodal Large Language Model (MLLM) optimized within our List-GRPO reinforcement learning framework.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.