SEEING BUT STRUGGLING TO SAY: ORGANIZING DISPERSED EVIDENCE IN MULTI-MODAL LARGE LANGUAGE MODELS FOR FINE-GRAINED RECOGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

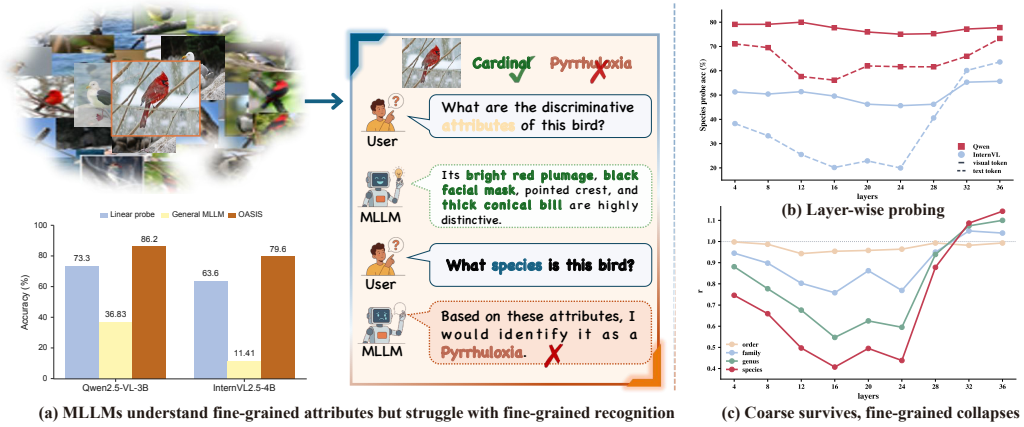
Multi-modal large language models (MLLMs) have shown remarkable general capabilities, yet still struggle with fine-grained visual recognition (FGVR), a long-standing task that demands distinguishing subordinate-level categories through subtle visual cues. While existing efforts focus on improving MLLMs for FGVR by introducing external supervision, we revisit this challenge by asking whether MLLMs already possess fine-grained understanding intrinsically. Through layer-wise linear probing, we find that the fine-grained evidence is indeed present across the hidden states, but can hardly be decoded by the native generation. Furthermore, the evidence disperses away from the answer position at intermediate layers, and this dispersion hurts fine-grained recognition far more than coarse-grained one. What MLLMs lack is thus not the fine-grained knowledge, but its organization into a discriminative structure that the readout can exploit. Motivated by this, we propose **O**rganizing **A**ggregated fine-grained evidence with **S**elf-el**I**cited attribute**S** (OASIS), a simple yet effective method that aggregates the dispersed evidence at the middle layer and organizes it by attributes self-elicited from the MLLM itself. With the backbone entirely frozen, OASIS unlocks the intrinsic fine-grained capability of MLLMs across six benchmarks and four backbones, while mitigating the intermediate-layer dispersion and instilling fine-grained discriminative structure. The code is available at `https://anonymous.4open.science/r/OASIS_MLLM`.

1 INTRODUCTION

Fine-grained visual recognition (FGVR) aims to distinguish subordinate-level categories within a super-category, such as bird species or dog breeds, which requires a precise understanding of subtle visual cues (Wei et al., 2022). Recent studies have paid increasing attention to adapting Multi-modal Large Language Models (MLLMs) for this task, mainly improving them with external fine-grained supervision, *e.g.*, aligning objects with category names via attribute-augmented learning (He et al., 2025), or distilling chain-of-thought reasoning over discriminative cues from stronger models (He et al., 2026). Despite the impressive results of MLLMs on general tasks (Liu et al., 2024a; Bai et al., 2025b), FGVR remains far from solved (Yu et al., 2026).

Differently, we revisit this challenging task by asking whether the prior of MLLMs already contains the knowledge for fine-grained understanding. Interestingly, we observe that an MLLM can accurately describe the discriminative attributes of different parts, yet still fails to name the fine-grained category (cf. Fig. 1(a)). Through layer-wise linear probing on CUB (Wah et al., 2011), a widely used benchmark of fine-grained bird species, we empirically find that the species information is readily decodable from the hidden states (cf. Fig. 1(b)), yet the native generation can hardly read the category out. Tracing across layers, we further observe that the fine-grained evidence remains stable in the visual tokens but disperses away from the answer position on the text side at intermediate layers, and the re-aggregation at final layers remains suboptimal for the readout, this dispersion barely affects coarse-grained recognition, which relies on a few salient cues, but severely hurts fine-grained recognition that requires combining multiple subtle cues to reason out the category (cf. Fig.1(c)). What MLLMs lack is therefore not the fine-grained knowledge, but the organization of the dispersed evidence into a fine-grained discriminative structure that the readout can exploit.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) MLLMs understand fine-grained attributes but struggle with fine-grained recognition (c) Coarse survives, fine-grained collapses

Figure 1: **MLLMs can understand the fine-grained evidence, but struggle to say the category.** (a) The MLLM accurately describes the discriminative attributes, yet names a wrong species; quantitatively, a linear probe recognizes the species far better than the native generation, and OASIS closes this gap. (b) Layer-wise probing on CUB shows that the evidence stays stable at the visual tokens (solid) yet disperses away from the answer position (dashed) at intermediate layers. (c) The retention ratio r between the two positions degrades with finer granularity, and the dispersion hits fine-grained recognition the hardest.

Motivated by this finding, we propose **OASIS**, *i.e.*, **O**rganizing **A**ggregated fine-grained evidence by **S**elf-el**I**cited attribute**S**, a simple yet effective method that deploys the intrinsic fine-grained capability of MLLMs by aggregating the dispersed evidence and organizing it by self-elicited attributes to instill fine-grained discriminative structure. OASIS intervenes exactly where the dispersion occurs. At the middle layer of the frozen MLLM by default, a learnable summary token adaptively gathers the evidence scattered across visual and textual tokens through a lightweight adapter. The aggregated representation is then organized by a contrastive objective guided by fine-grained attributes, where the MLLM itself annotates the visual attributes of each image, and the attribute similarity among categories softly shapes the contrast toward fine-grained structure. OASIS trains only $\sim$3% parameters in a single stage with the backbone entirely frozen, requiring no external priors.

Extensive experiments on six fine-grained benchmarks across four MLLMs validate the effectiveness of OASIS. With only $\sim$3% trainable parameters, OASIS consistently surpasses the compared methods in all settings, and the advantage becomes more pronounced on the more challenging iNaturalist subsets, with gains up to 36.2 on iNat-Plantae covering 4,271 fine-grained species. Moreover, the injection restores the fine-grained evidence at the answer position from the injection layer onward, and both saliency and representation visualizations confirm that OASIS concentrates the visual evidence onto discriminative regions and organizes the latent space by category. In summary, our contributions are threefold:

- We empirically reveal that fine-grained evidence is already present in MLLMs, and further find that it disperses at intermediate layers, hurting fine-grained recognition far more than the coarse-grained counterparts.
- We propose OASIS, a simple yet effective method that aggregates the dispersed evidence and organizes it by self-elicited attributes, with the MLLMs entirely frozen and no external priors required.
- OASIS consistently outperforms full fine-tuning and parameter-efficient methods across six benchmarks, while probing and visualization confirm that it mitigates the intermediate-layer dispersion and instills fine-grained discriminative structure.

2 Seeing but Struggling to Say: An Empirical Study

In this section, we first reveal that MLLMs already possess fine-grained knowledge from large-scale pretraining, yet fail to organize such evidence to reason out the correct category. We then further explore why such evidence fails to translate into fine-grained recognition, and the analysis motivates the design of OASIS.

Fine-grained evidence is already inside MLLMs To examine whether MLLMs encode fine-grained evidence, we train linear probes on the hidden states across layers at two kinds of positions, the answer position, *i.e.*, the last prompt token that conditions the generation, and the visual token positions, to classify the species on CUB.

Taking Qwen2.5-VL-3B and InternVL2.5-4B as examples, the species information is decodable across layers and positions, as shown in Fig. 1(b). On Qwen2.5-VL-3B, the probe on visual tokens stays at 75–80% throughout the network, and the answer position reaches 73.3% at the final layer, in line with prior findings that classification-related information is encoded in the latent space of MLLMs (Zhang et al., 2024b). InternVL2.5-4B exhibits the same pattern with a lower absolute level (55.7% and 63.6% at the final layer). In stark contrast, the native generation on the same models reaches only 36.8% and 11.4% on CUB. The fine-grained information is thus already present in MLLMs, yet cannot be decoded by the language model's generation. What is missing is not the evidence itself, but its organization into a form the readout can use.

Fine-grained evidence disperses at intermediate layers Fig. 1(b) further reveals where the evidence goes. On Qwen2.5-VL-3B, the probe on visual tokens stays stable at 75–80% across layers, indicating that the visual evidence is well preserved throughout the network. The answer position, however, starts close to the visual level at early layers, drops sharply to 55.9% at the middle layer (a gap of 22.4% at layer 12). InternVL2.5-4B shows the same trajectory with the deeper drop, where the answer position falls to 20.2% at layer 16 while the visual tokens still remain around 50%.

This reveals a clear dispersion, where the fine-grained evidence stays in the visual tokens but drifts away from the position that conditions the generation at intermediate layers. Although the final layers partially re-aggregate it, such late and passive re-aggregation is suboptimal, as the recovered representation is neither complete nor in a form the native readout can exploit, consistent with the observation that final-layer representations shift toward next-token computation rather than maintaining the abstract information (Hernandez et al., 2024).

Dispersion hits fine-grained recognition harder than coarse-grained The dispersion of evidence appears to be a general property of MLLMs, yet it is far more damaging for fine-grained recognition, which relies on the joint of multiple subtle cues rather than a few salient ones.

To verify this, we exploit the taxonomy of CUB and evaluate the same probes at four granularities (Chen et al., 2022), from order down to species, measuring at each layer the ratio r_ℓ of the answer-position decodability to the visual-token decodability, which reflects how much evidence survives from the visual tokens to the answer position. As the ratio is computed within each granularity, it isolates the effect of dispersion from the absolute difficulty induced by the number of categories.

As shown in Fig. 1(c), coarse-grained recognition is nearly immune to the dispersion, with r_ℓ staying above 0.96 at the order level across all layers, whereas the ratio degrades monotonically with finer granularity and collapses to 0.41 at the species level on InternVL2.5-4B. Furthermore, the native generation on Qwen2.5-VL-3B exhibits the same trend, reaching 69.5% at the order level but only 36.8% at the species level. Coarse-grained categories can be resolved from a single salient cue that survives the dispersion, while fine-grained categories demand combining several cues, exactly what the dispersion breaks apart. The general dispersion thus turns into a fine-grained-specific failure, and a targeted intervention at the dispersing layers is where the remedy pays off.

3 METHODOLOGY

In this section, we first present an overview of the proposed OASIS, and then elaborate its two key steps, *i.e.*, evidence aggregation and attribute-guided organization.

3.1 OVERVIEW

OASIS aggregates dispersed fine-grained evidence, then organizes it by self-elicited attributes to instill fine-grained discriminative structure. As illustrated in Fig. 2, given an image x_i and its textual prompt p_i, the frozen MLLM performs a standard forward pass, and we take the hidden sequence $\mathbf{H}_i^\ell \in \mathbb{R}^{T_i \times d}$ after the middle layer $\ell = \lfloor N/2 \rfloor$, where N is the number of decoder layers, T_i the

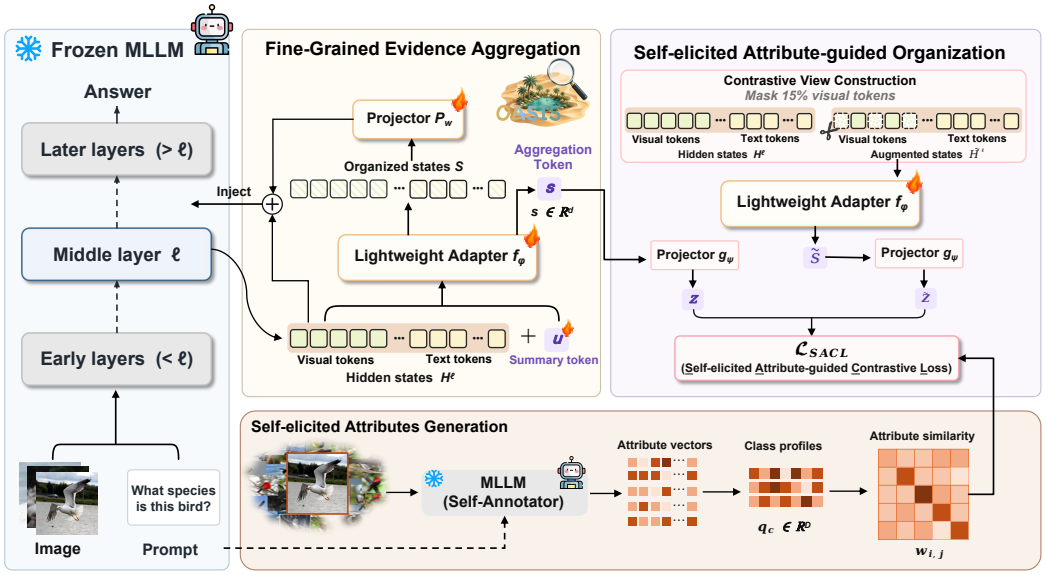


Figure 2: **Overview of OASIS.** At the middle layer of the frozen MLLM, a learnable summary token $\mathbf{u}$ and a lightweight adapter f_ϕ aggregate the dispersed evidence into the summary $\mathbf{s}$. A masked view yields the counterpart $\tilde{\mathbf{s}}$, and the embeddings $(\mathbf{z}, \tilde{\mathbf{z}})$ are optimized by SACL, whose soft weights w_{ij} derive from class-level attribute profiles annotated by the MLLM itself.

number of prompt tokens, and d the hidden dimension. A learnable summary token $\mathbf{u} \in \mathbb{R}^d$ is concatenated to the sequence, and a lightweight adapter f_ϕ transforms them into $[\mathbf{S}_i, \mathbf{s}_i]$, where $\mathbf{s}_i$ aggregates the fine-grained evidence from the entire sequence and $\mathbf{S}_i$ denotes the transformed sequence. The summary $\mathbf{s}_i$ is projected by a head g_ψ into an embedding $\mathbf{z}_i$ for the organization objective, while $\mathbf{S}_i$ is injected back into the frozen residual stream through a zero-initialized projection P_ω, after which the remaining layers and the native LM head proceed unchanged to produce the answer. OASIS is trained in a single end-to-end stage with the joint objective

$$\mathcal{L} \;=\; \mathcal{L}_{\mathrm{CE}} \;+\; \lambda\, \mathcal{L}_{\mathrm{SACL}}, \tag{1}$$

where $\mathcal{L}_{\mathrm{CE}}$ is the standard cross-entropy over the answer tokens, and $\mathcal{L}_{\mathrm{SACL}}$ is the proposed Self-elicited Attribute-guided Contrastive Loss that organizes the aggregated representation toward fine-grained discriminative structure, and λ is a trade-off weight balancing the two terms.

3.2 Aggregating Dispersed Fine-grained Evidence

As analyzed previously, fine-grained evidence remains inside the frozen MLLMs, yet disperses across token positions at intermediate layers. In this subsection, we detail the first step of OASIS, *i.e.*, how the dispersed evidence is aggregated from the frozen hidden states. To adaptively gather fine-grained evidence distributed across visual and textual tokens into a compact representation, we first introduce a learnable summary token $\mathbf{u} \in \mathbb{R}^d$ appended to the frozen hidden sequence, and process them jointly with a lightweight adapter f_ϕ:

$$[\,\mathbf{S}_i,\ \mathbf{s}_i\,] \;=\; f_\phi\big(\mathbf{H}_i^\ell \oplus \mathbf{u}\big), \tag{2}$$

where $\oplus$ denotes concatenation along the sequence dimension, f_ϕ is instantiated as a two-layer Transformer encoder (Dosovitskiy et al., 2021), $\mathbf{S}_i$ is the transformed counterpart of the original hidden sequence, and $\mathbf{s}_i$ is the output at the summary position—the result of $\mathbf{u}$ acting as a learnable query that retrieves evidence across visual and textual tokens. Notably, the aggregated $\mathbf{s}_i$ is subsequently organized toward fine-grained discriminative structure under the guidance of self-elicited attributes, making the evidence the MLLMs already possess directly usable for fine-grained recognition. And the aggregation introduces nothing beyond what the MLLMs already encode, as $\mathbf{u}$ lives only in the hidden space and adds no entry to the tokenizer.

3.3 Organizing Fine-grained Evidence with Self-elicited Attributes

The aggregated evidence is then organized into a form better suited for fine-grained recognition within the MLLMs. We shape the summary representation through a contrastive objective to make the features discriminative, while the guidance of self-elicited attributes further instills fine-grained semantic structure.

Specifically, to encourage $\mathbf{u}$ to capture fine-grained cues, we construct a contrastive augmentation pair for each representation, where a contiguous span covering $\rho = 15\%$ of the visual tokens in $\mathbf{H}_i^\ell$ is masked out, and the masked sequence passes through the same adapter to yield a cutoff summary $\tilde{\mathbf{s}}_i$. The two summaries are then projected by a lightweight head g_ψ into ℓ_2-normalized embeddings:

$$\mathbf{z}_i = g_\psi(\mathbf{s}_i), \tag{3}$$

$$\tilde{\mathbf{z}}_i = g_\psi(\tilde{\mathbf{s}}_i), \tag{4}$$

which serve as two views for the contrastive objective. Beyond contrasting instances without semantic awareness, the objective should reflect how fine-grained categories relate to each other through their attributes. Such attribute structure also comes from the MLLM itself, serving as a self-annotator, prompted to annotate the visible attributes of each training image, with category names strictly excluded (detailed in App. B.1). The responses are parsed into a fixed-length attribute vector $\mathbf{a}_i \in \mathbb{R}^D$, and vectors within each category are averaged into a class-level profile:

$$\mathbf{q}_c \;=\; \frac{1}{|\mathcal{I}_c|} \sum_{i \in \mathcal{I}_c} \mathbf{a}_i, \tag{5}$$

where $\mathcal{I}_c$ denotes the set of training images belonging to category c, followed by dimension-wise standardization across classes. As the annotator is the backbone itself, no distillation from stronger models or external priors is introduced. The class profiles then define how strongly two samples should attract each other. For a batch of samples, we compute soft weights from the attribute distance between their categories:

$$w_{ij} \;=\; \frac{\exp\big(-\|\mathbf{q}_{c_i} - \mathbf{q}_{c_j}\|_2/\beta\big)}{\sum_k \exp\big(-\|\mathbf{q}_{c_i} - \mathbf{q}_{c_k}\|_2/\beta\big)}, \tag{6}$$

where β is a temperature controlling the sharpness. Categories sharing similar attributes thus serve as soft positives, injecting the fine-grained inter-class geometry into the objective. Based on these, we propose the Self-elicited Attribute-guided Contrastive Loss (SACL), formulated as

$$\mathcal{L}_{\text{SACL}} \;=\; -\sum_i \sum_j w_{ij} \,\log \frac{\exp\big(\text{sim}(\mathbf{z}_i, \tilde{\mathbf{z}}_j)/\tau\big)}{\sum_k \exp\big(\text{sim}(\mathbf{z}_i, \tilde{\mathbf{z}}_k)/\tau\big)}, \tag{7}$$

where $\text{sim}(\cdot,\cdot)$ denotes cosine similarity and τ is the temperature. Rather than merely separating instances, SACL organizes the summaries so that attribute-similar categories stay close while remaining distinguishable, instilling the desired fine-grained discriminative structure. Finally, the organized evidence is injected back into the frozen pathway through a zero-initialized projection P_ω as

$$\mathbf{H}_i^\ell \;\leftarrow\; \mathbf{H}_i^\ell + P_\omega(\mathbf{S}_i). \tag{8}$$

The injection is performed only once at the middle layer $\ell = \lfloor N/2 \rfloor$ by default, while the joint objective in Eq. 1 trains the lightweight parameters end-to-end.

4 Experiments

In this section, we first introduce the datasets and empirical settings in our experiments. Then, we present the main results of our approach, as well as comparisons with state-of-the-arts. Comprehensive ablations further verify the effectiveness of OASIS, and confirm that it indeed mitigates the dispersion.

4.1 Datasets and Empirical Settings

We conduct experiments on six widely-used fine-grained benchmarks. *CUB* (Wah et al., 2011) consists of 200 bird species with 11,788 images, providing part-level annotations and a 312-dimensional class-attribute profile. *Stanford Dogs* (Khosla et al., 2011) contains 20,580 images of 120 breeds

Table 1: Accuracy (%) on six fine-grained recognition benchmarks across four backbones. Params denotes the number of trainable parameters. Results are reported as mean±std over three random seeds. Zero-shot models rarely produce the make-model-year format required by Stanford Cars, yielding zero strict accuracy.

Backbone	Method	Params	iNat-Fungi	iNat-Plantae	CUB	Dogs	Cars	Aircraft
Qwen2.5-VL-3B	Zero-shot	–	20.32	27.09	36.83	52.28	0.00	53.83
	SFT	∼3.0B	69.54±0.26	65.63±0.30	79.19±0.11	79.32±0.50	87.65±0.27	79.36±0.39
	OASIS (Ours)	∼0.1B	**79.91**±0.11	**78.75**±0.18	**86.19**±0.17	**82.32**±0.37	**92.02**±0.31	**82.03**±0.26
Qwen3-VL-4B	Zero-shot	–	8.12	11.97	37.10	54.23	0.00	57.74
	SFT	∼3.0B	70.76±0.20	62.19±0.45	80.01±0.15	81.92±0.38	89.21±0.20	85.27±0.26
	OASIS (Ours)	∼0.1B	**82.99**±0.27	**78.41**±0.26	**87.59**±0.18	**84.76**±0.35	**90.75**±0.23	**91.87**±0.25
InternVL2.5-4B	Zero-shot	–	2.46	4.21	11.41	27.63	0.00	23.46
	SFT	∼3.0B	62.17±0.31	35.10±0.29	54.90±0.29	62.21±0.31	74.19±0.36	52.57±0.28
	OASIS (Ours)	∼0.1B	**78.65**±0.26	**71.26**±0.34	**79.60**±0.24	**78.36**±0.21	**84.61**±0.45	**70.05**±0.33
LLaVA-OV-7B	Zero-shot	–	0.29	0.15	8.00	25.03	0.00	56.71
	SFT	∼7.0B	73.40±0.27	69.58±0.39	84.66±0.34	85.31±0.26	92.05±0.23	92.61±0.47
	OASIS (Ours)	∼0.3B	**84.25**±0.15	**79.33**±0.20	**90.03**±0.29	**87.70**±0.33	**95.56**±0.20	**95.70**±0.40

Table 2: Accuracy (%) compared with related adaptation methods on Qwen2.5-VL-3B. Prompt tuning reports the best result over prompt lengths $K \in \{20, 100, 200\}$. Results are mean±std over three random seeds.

Method	iNat-Fungi	iNat-Plantae	CUB	Dogs	Cars	Aircraft
SFT	69.54±0.26	65.63±0.30	79.19±0.11	79.32±0.50	87.65±0.27	79.36±0.39
Prompt tuning	67.89±0.28	66.70±0.25	80.69±0.21	78.77±0.15	85.73±0.19	76.93±0.37
LoRA	73.21±0.26	70.04±0.23	82.47±0.35	81.29±0.10	89.75±0.27	80.79±0.31
OASIS (Ours)	**79.91**±0.11	**78.75**±0.18	**86.19**±0.17	**82.32**±0.37	**92.02**±0.31	**82.03**±0.26

sourced from ImageNet, exhibiting large intra-class variation. *Stanford Cars* (Krause et al., 2013) comprises 16,185 images of 196 models defined at the granularity of make, model, and year. *FGVC-Aircraft* (Maji et al., 2013) covers 100 aircraft variants with 10,000 images in a four-level hierarchy. We further evaluate on the Fungi and Plantae subsets of *iNaturalist 2021* (Van Horn et al., 2021), containing 341 and 4,271 species with 90,048 and 1,148,702 training images, respectively, collected as real-world observations with long-tailed appearance variation.

OASIS is trained in a single stage on four widely-used MLLMs, *i.e.*, Qwen2.5-VL-3B (Bai et al., 2025b), Qwen3-VL-4B (Bai et al., 2025a), InternVL2.5-4B (Chen et al., 2024), and LLaVA-OneVision-7B (Li et al., 2024), updating only $\sim 3\%$ parameters while keeping the backbone frozen, with the evidence aggregated, organized, and injected back at the middle layer $\ell = \lfloor N/2 \rfloor$ by default. For fair comparisons, OASIS and all trained baselines are optimized for 3 epochs with AdamW under identical schedules on 8 NVIDIA H100 GPUs, and we set $\lambda = 1$, $\tau = 0.2$, and $\beta = 1$ by default. At inference, all methods share the same simple instructions (cf. App. C.1).

4.2 Main Results

As shown in Tab. 1, OASIS consistently surpasses full fine-tuning across all settings, while updating only $\sim 3\%$ of the parameters, *e.g.*, lifting Qwen2.5-VL-3B from 79.19% to 86.19% on CUB, exceeding full SFT by 7.0%. The advantage becomes more pronounced on the more challenging iNaturalist subsets, where the gains reach 13.1% on iNat-Plantae with 4,271 species. The gains hold across model families and scales, and grow substantially larger on InternVL2.5-4B, reaching 24.7 on CUB and 36.2 on iNat-Plantae. Notably, the gains scale with the severity of the diagnosed dispersion, as InternVL exhibits a much deeper drop in answer-position decodability on CUB (cf. Sec. 2) and benefits the most. This corroborates our central claim that the bottleneck lies not in the evidence itself, but in organizing it into a readout-usable form, which a lightweight module suffices to close.

As reported in Tab. 2, tuning-based alternatives bring limited gains under matched budgets. Prompt tuning performs on par with full SFT, as task-level vectors at the input side can hardly carry the discriminative structure of hundreds of categories. LoRA surpasses full SFT by constraining the update to preserve the pretrained representation (Zhu et al., 2026; Hu et al., 2022), yet still trails OASIS on all six benchmarks, with the gap up to 8.71% on iNat-Plantae. Rather than tuning the model toward the task, OASIS organizes the dispersed evidence the frozen MLLM already encodes, offering a simple yet effective mechanism for fine-grained recognition.

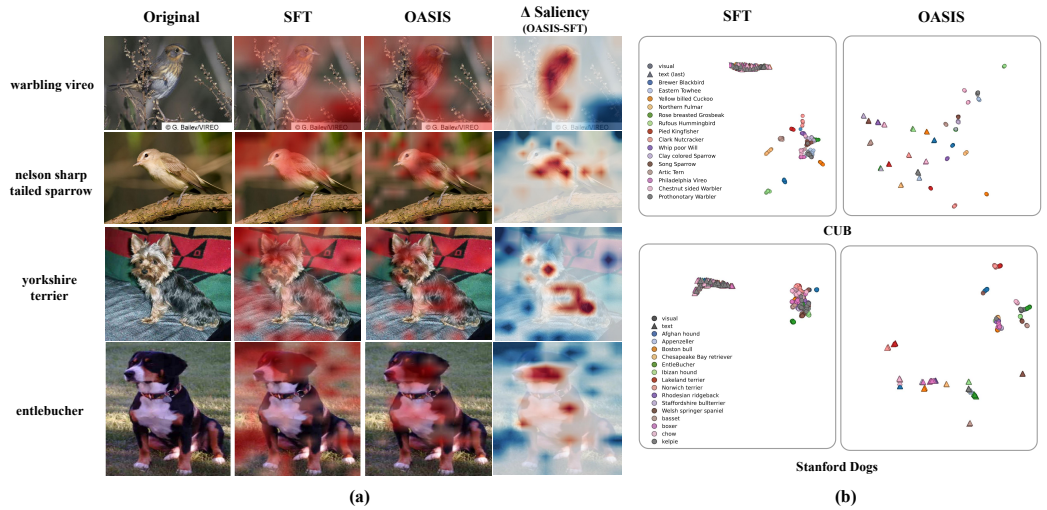


Figure 3: Qualitative analyses of OASIS on Qwen2.5-VL-3B. (a) Saliency of the generated answer over visual tokens on CUB and Stanford Dogs. The last column shows the saliency difference (OASIS − SFT), where red marks regions attended more by OASIS and blue by SFT. OASIS concentrates saliency on the discriminative object regions, whereas SFT disperses it over the background. (b) UMAP (McInnes et al., 2018) of visual tokens (circles) and the answer-position text token (triangles) at the injection layer, over 15 categories with 25 samples each. Under SFT, categories remain entangled and collapse into undifferentiated clumps; with OASIS, the text token of each image is drawn to the visual evidence of its own category, and categories become well separated.

4.3 Ablation Studies and Discussions

We hereby analyze and discuss our OASIS method from the following aspects based on ablation studies with two VLM backbones across two datasets, *i.e.*, *CUB* and *Stanford Dogs*.

OASIS organizes the representation toward fine-grained structure. To further verify the effectiveness of OASIS in guiding the representation, we first qualitatively examine how the fine-grained visual evidence is exploited and how the latent space is organized. Specifically, we visualize the visual-token saliency over the image. As shown in Fig. 3(a), OASIS concentrates saliency on the discriminative object regions, whereas SFT fails to guide the generation toward the fine-grained cues. Meanwhile, we visualize the latent representations at the same layer via UMAP after PCA reduction. In contrast to SFT, whose representations remain entangled among fine-grained categories with no discriminative structure, OASIS reorganizes the space by category, drawing the answer-position token to the visual evidence of its own class and separating the categories apart (cf. Fig. 3(b)).

Beyond the qualitative evidence, we further conduct the layer-wise probing on the OASIS-injected model. The answer-position decodability jumps sharply at the injection layer, lifting by 30.6% on Qwen2.5-VL-3B and 47.2% on InternVL2.5-4B, and stays stable through all subsequent layers, while layers before the injection remain identical to the frozen model (cf. Fig. 4). Notably, the deeper the diagnosed dispersion, the larger the restoration, and this quantitatively verifies that simply aggregating and organizing the dispersed evidence suffices to make it directly decodable at the readout.

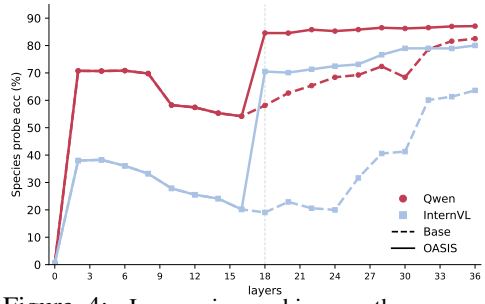


Figure 4: Layer-wise probing on the answer-position text token with and without OASIS.

Aggregating with different mechanisms To aggregate the evidence effectively, we simply introduce a learnable summary token. We compare it against three alternatives, mean pooling, attention pooling with a learnable readout query, and a token-level variant that applies the same organization objective directly on sampled sequence positions without aggregation. As reported in Tab. 3, organizing without aggregation performs the worst, trailing OASIS by 2.46% on CUB and 2.17% on Dogs, as the attribute guidance becomes inefficient when imposed on evidence still dispersed across positions. Attention pooling holds the same parameters yet its query only reads out at the end, falling

Table 3: Accuracy (%) of aggregating with different mechanisms with Qwen2.5-VL-3B.

Aggregation mechanism	CUB	Dogs
w/o aggregation	83.73	80.15
Attention pooling	84.10	81.96
Mean pooling	85.39	81.47
Summary token (OASIS)	**86.19**	**82.32**

Table 4: Accuracy (%) of aggregating from different evidence sources with Qwen2.5-VL-3B. Text-only restricts the adapter to non-visual tokens.

Aggregation source	CUB	Dogs
Full SFT (reference)	79.19	79.32
Text-only tokens	80.37	77.09
Visual + text (OASIS)	**86.19**	**82.32**

behind the summary token that joins the self-attention and adaptively retrieves the evidence across the whole sequence. The summary token in OASIS is thus a simple yet effective way to condense the dispersed fine-grained evidence.

Aggregating from different evidence sources As fine-grained evidence distributes across visual and textual tokens, we further explore whether the evidence dispersed in textual tokens alone suffices, by restricting the aggregation input to text tokens, in contrast to OASIS that reads both. As shown in Table 4, this restriction degrades the accuracy by 5.82% on CUB and 5.23% on Stanford Dogs, revealing that critical visual evidence for fine-grained recognition resides only in the visual tokens. Aggregating from the full sequence recovers this evidence directly from the visual source, and consistently surpasses full SFT that updates $30\times$ more parameters.

Effectiveness of the organization objective We dissect the training objective by incrementally adding each term, as reported in Table 5. Training the aggregation module with cross-entropy alone already surpasses full SFT (81.97% vs. 79.19% on CUB), confirming that aggregating and injecting the dispersed evidence alone delivers a solid gain with $\sim$3% parameters. On top of it, vanilla InfoNCE adds 1.85%, and SACL further contributes the largest share of 2.37%, together lifting the accuracy by 7.0% over full SFT. The key thus lies not in contrasting instances but in the fine-grained attribute structure instilled into the representation, and the improvement stems from how the evidence is organized rather than from the training signal itself.

Table 5: Accuracy (%) of different training objectives with Qwen2.5-VL-3B.

Objective	CUB	Dogs
Full SFT (reference)	79.19	79.32
$\mathcal{L}_{\mathrm{CE}}$ only (λ=0)	81.97	79.27
+ vanilla InfoNCE	83.82	81.03
+ SACL (OASIS)	**86.19**	**82.32**

Organizing with self-elicited attributes To validate the effectiveness of self-elicited attribute annotation, we conduct experiments on CUB, which provides official attribute vectors, comparing soft targets built from three sources, *i.e.*, the official expert annotations (Gold), attributes self-elicited by the backbone itself, and those annotated by a larger model of the same family. As reported in Tab. 6, expert attributes yield the best accuracy, confirming that the quality of attribute structure directly translates into that of organization. Self-elicited attributes retain most of this benefit without additional annotation cost, still surpassing the vanilla contrastive objective by 2.37%, while scaling up the annotator brings no further gain. OASIS thus adopts self-elicitation by default, readily extending to datasets without expert annotations.

Table 6: Accuracy (%) of organizing with attributes from different annotators on CUB. Gold denotes the official 312-d expert attributes, and 8B denotes annotation by a larger model of the same family.

Attribute annotator	Qwen2.5-VL	InternVL2.5
Gold (official)	**88.15**	**80.20**
Self-elicited (OASIS)	86.19	79.60
Larger annotator (8B)	86.10	79.70

Aggregating at different layers As previously explored, the dispersion occurs at intermediate layers, and OASIS accordingly intervenes at $\ell = \lfloor N/2 \rfloor$ by default. Varying the injection layer around this choice, injecting at any middle layer brings consistent improvements without cherry-picking (cf. Tab. 7), indicating that OASIS is insensitive to the exact position as long as the intervention lands where the evidence disperses. The default $\lfloor N/2 \rfloor$ thus serves as a universal choice across backbones.

Table 7: Accuracy (%) of injecting at different layers around the default $\ell = \lfloor N/2 \rfloor$. OASIS is insensitive to the exact injection layer within the middle region.

Backbone	Dataset	$\ell-2$	$\ell-1$	ℓ (default)	$\ell+1$	$\ell+2$
Qwen2.5-VL-3B	CUB	86.07	**86.60**	86.19	86.29	85.87
	Dogs	80.89	80.71	**82.32**	81.30	81.20
InternVL2.5-4B	CUB	79.09	79.21	79.60	**80.00**	79.83
	Dogs	77.87	78.11	78.36	79.08	**79.69**

5 Related Work

MLLMs for Fine-Grained Recognition Recent work (Yu et al., 2026) reveals that MLLMs perceive visual appearances well yet struggle to recognize subordinate-level categories. Existing methods fall into two lines. Fine-tuning methods update the model with abundant classification data (Zhang et al., 2024b), attribute-augmented alignment between objects and category names (He et al., 2025), or CoT-supervised fine-tuning (He et al., 2026). Training-free approaches instead exploit sparse attention activations as discriminative features (Mitra et al., 2025), reason over the extracted visual attributes with LLMs (Liu et al., 2024b), or prompt optimization with unlabeled data (Hong et al., 2025). Our approach also involves training, yet stems from an empirical finding and introduces only a lightweight module that reorganizes the fine-grained evidence already inside the MLLM, without external priors or backbone retraining.

Exploring Internal Representations of MLLMs The internal representations of MLLMs have long been an intriguing question. Recent works trace the internal information flow, revealing that visual information is broadcast, selectively injected, and finally aggregated across layers (Jiang et al., 2025; Zhang et al., 2025), while final-layer representations shift toward next-token computation (Hernandez et al., 2024). Meanwhile, the generative performance of MLLMs is observed to fall below the linear separability of their own representations (Zhang et al., 2024b; Mitra et al., 2025), and eliciting such internally encoded yet unexpressed information has proven effective in LLMs (Li et al., 2023). We further find that such internal discrepancy is substantially amplified in fine-grained recognition, as it demands exploiting subtle fine-grained cues rather than salient ones. Accordingly, we organize the dispersed evidence at the middle layer for direct fine-grained readout.

Lightweight Representation Adaptation Parameter-efficient methods adapt frozen models by constraining updates to low-rank subspaces (Hu et al., 2022) or injecting learnable prompts (Zhou et al., 2022; Zhang et al., 2024a), which have also been adopted for adapting models to fine-grained tasks (He et al., 2025). Representation steering shows that adding extracted directions to hidden states elicits behaviors the model internally encodes, such as truthfulness (Li et al., 2023) and visual grounding (Li et al., 2025). Differently, OASIS learns to aggregate and organize per-image evidence, injecting an input-dependent representation that carries fine-grained structure into the frozen pathway.

6 Conclusion

In this work, we studied FGVR with MLLMs from the perspective of deploying the intrinsic prior. Different from previous studies that improve MLLMs with external supervision, we started with asking whether the fine-grained knowledge already resides within. Through layer-wise probing, we revealed that the fine-grained evidence is present in the hidden states yet disperses away from the answer position at intermediate layers, hurting fine-grained recognition far more than the coarse-grained counterpart. Accordingly, we proposed OASIS, which aggregates the dispersed evidence with a learnable summary token and organizes it by self-elicited attributes, keeping the backbone entirely frozen. With only $\sim$3% trainable parameters, OASIS consistently surpassed the compared methods across six benchmarks and four backbones. In the future, we plan to extend the organization of intrinsic priors beyond fine-grained recognition to fine-grained manipulation, where vision-language-action models rely even more heavily on exploiting what the MLLMs already encode.

AI USE STATEMENT

In this work, we used generative AI tools for the following tasks with required disclosure:

- *Design or provide feedback on research methodology or experiments.* We used them to discuss the design of the ablation studies and the counterfactual analyses that validate the proposed method.
- *Implement methods.* The framework and the key components were written by the authors; the tools were used to debug the code and improve its readability.
- *Interpret results.* The tools helped analyze the layer-wise probing curves and the ablation results.

We have not used generative AI tools to generate synthetic data sets, to help develop theoretical models or conceptual frameworks, to propose or refine hypotheses, to assist with translation, or to clean and reformat datasets. Note that the attribute annotation in OASIS is produced by the backbone MLLM itself as an integral component of the proposed method (cf. Sec. 3.3 and App. B.1), rather than as an assistive tool in our research workflow. The remaining required-disclosure tasks, namely formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, and supporting qualitative and thematic data analysis, are not applicable.

Additionally, we used generative AI tools for the following tasks with recommended disclosure:

- *Edit a research paper to improve readability.* The initial drafts were written by the authors, and the tools were used to polish the wording and to shorten passages under the page limit.
- *Identify relevant literature.* We used them to search for related work, and every returned paper was checked by the authors before being cited.
- *Suggest a structure for a research paper.* We discussed the organization of the method and experiment sections with the tools.
- *Propose a title or keywords for a research paper.* We discussed candidate titles with the tools.

We have reviewed all AI-assisted work, including verifying every AI-suggested reference against the original publication, and take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The source code is available at `https://anonymous.4open.science/r/OASIS_MLLM`. The prompts for recognition and for the self-elicited attribute annotation are provided in App. B. The probing protocol behind our empirical findings is detailed in App. A, and the implementation details of OASIS and all compared methods are provided in App. C.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan,

Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Jingzhou Chen, Peng Wang, Jian Liu, and Yuntao Qian. Label relation graphs enhanced hierarchical residual network for hierarchical multi-granularity classification. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 4858–4867, 2022.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, Lixin Gu, Xuehui Wang, Qingyun Li, Yiming Ren, Zixuan Chen, Jiapeng Luo, Jiahao Wang, Tan Jiang, Bo Wang, Conghui He, Botian Shi, Xingcheng Zhang, Han Lv, Yi Wang, Wenqi Shao, Pei Chu, Zhongying Tu, Tong He, Zhiyong Wu, Huipeng Deng, Jiaye Ge, Kai Chen, Kaipeng Zhang, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. Int. Conf. Learn. Representations*, 2021.

Hulingxiao He, Geng Li, Zijun Geng, Jinglin Xu, and Yuxin Peng. Analyzing and boosting the power of fine-grained visual recognition for multi-modal large language models. In *Proc. Int. Conf. Learn. Representations*, pp. 89927–89944, 2025.

Hulingxiao He, Zijun Geng, and Yuxin Peng. Fine-R1: Make multi-modal LLMs excel in fine-grained visual recognition by chain-of-thought reasoning. In *Proc. Int. Conf. Learn. Representations*, pp. 68140–68158, 2026.

Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. Linearity of relation decoding in transformer language models. In *Proc. Int. Conf. Learn. Representations*, pp. 10504–10526, 2024.

Yunqi Hong, Sohyun An, Andrew Bai, Neil Lin, and Cho-Jui Hsieh. Unlabeled data improves fine-grained image zero-shot classification with multimodal LLMs. In *Proc. Advances in Neural Inf. Process. Syst.*, pp. 74304–74326, 2025.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *Proc. Int. Conf. Learn. Representations*, 2022.

Zhangqi Jiang, Junkai Chen, Beier Zhu, Tingjin Luo, Yankun Shen, and Xu Yang. Devils in middle layers of large vision-language models: Interpreting, detecting and mitigating object hallucinations via attention lens. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 25004–25014, 2025.

Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Fei-Fei Li. Novel dataset for fine-grained image categorization: Stanford dogs. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn. Workshops*, 2011.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3D object representations for fine-grained categorization. In *Proc. IEEE Int. Conf. Comp. Vis. Workshops*, pp. 554–561, 2013.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proc. Conf. Empirical Methods in Natural Language Processing*, pp. 3045–3059, 2021.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Proc. Advances in Neural Inf. Process. Syst.*, pp. 41451–41530, 2023.

Zhuowei Li, Haizhou Shi, Yunhe Gao, Di Liu, Zhenting Wang, Yuxiao Chen, Ting Liu, Long Zhao, Hao Wang, and Dimitris N. Metaxas. The hidden life of tokens: Reducing hallucination of large vision-language models via visual information steering. In *Proc. Int. Conf. Mach. Learn.*, pp. 35799–35819, 2025.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 26296–26306, 2024a.

Mingxuan Liu, Subhankar Roy, Wenjing Li, Zhun Zhong, Nicu Sebe, and Elisa Ricci. Democratizing fine-grained visual recognition with large language models. In *Proc. Int. Conf. Learn. Representations*, pp. 16942–16973, 2024b.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Chancharik Mitra, Brandon Huang, Tianning Chai, Zhiqiu Lin, Assaf Arbelle, Rogerio Feris, Leonid Karlinsky, Trevor Darrell, Deva Ramanan, and Roei Herzig. Enhancing few-shot vision-language classification with large multimodal model features. In *Proc. IEEE Int. Conf. Comp. Vis.*, pp. 2760–2772, 2025.

Grant Van Horn, Elijah Cole, Sara Beery, Kimberly Wilber, Serge Belongie, and Oisin Mac Aodha. Benchmarking representation learning for natural world image collections. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 12884–12893, 2021.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, Pasadena, CA, 2011.

Xiu-Shen Wei, Yi-Zhe Song, Oisin Mac Aodha, Jianxin Wu, Yuxin Peng, Jinhui Tang, Jian Yang, and Serge Belongie. Fine-grained image analysis with deep learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(12):8927–8948, 2022.

Hong-Tao Yu, Yuxin Peng, Serge Belongie, and Xiu-Shen Wei. Benchmarking large vision-language models on fine-grained image tasks: A comprehensive evaluation. In *Proc. Int. Conf. Learn. Representations*, pp. 44677–44700, 2026.

Renrui Zhang, Jiaming Han, Chris Liu, Aojun Zhou, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. LLaMA-Adapter: Efficient fine-tuning of large language models with zero-initialized attention. In *Proc. Int. Conf. Learn. Representations*, pp. 44389–44418, 2024a.

Yuhui Zhang, Alyssa Unell, Xiaohan Wang, Dhruba Ghosh, Yuchang Su, Ludwig Schmidt, and Serena Yeung-Levy. Why are visually-grounded language models bad at image classification? In *Proc. Advances in Neural Inf. Process. Syst.*, pp. 51727–51753, 2024b.

Zhi Zhang, Srishti Yadav, Fengze Han, and Ekaterina Shutova. Cross-modal information flow in multimodal large language models. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pp. 19781–19791, 2025.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *Int. J. Comput. Vision*, 130(9):2337–2348, 2022.

Jie Zhu, Yiyang Su, and Xiaoming Liu. Can textual reasoning improve the performance of mllms on fine-grained visual classification? *arXiv preprint arXiv:2601.06993*, 2026.

APPENDIX OVERVIEW

This appendix provides additional analyses, annotation details, and implementation details omitted from the main body due to space limitations:

- Section A examines how fine-grained information is available across layers and backbones, detailing the probing protocol and the granularity-wise retention ratio.
- Section B describes the self-elicited attribute annotation, including the annotation protocol, the instruction templates, and annotation examples.
- Section C provides implementation details, covering the prompt templates, the architecture and training of OASIS, and the compared methods, namely supervised fine-tuning, parameter-matched LoRA, and prompt tuning.

A Fine-Grained Availability Across Layers and Backbones

A.1 Implementation Details

Probing protocol. We probe on CUB under the official split, with 5,994 training and 5,794 test images, and formulate a 200-way species classification task. The backbone is entirely frozen and runs in inference mode without gradient tracking. For each image, we perform a prompt-only prefill, where neither the ground-truth category name nor any generated answer enters the sequence, so that the extracted states reflect what the model encodes before producing the answer. At each decoder depth ℓ, we extract two representations,

$$\mathbf{p}_{\text{vis}}^{\ell} = \frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} \mathbf{h}_i^{\ell}, \tag{9}$$

$$\mathbf{p}_{\text{last}}^{\ell} = \mathbf{h}_{T_i}^{\ell}, \tag{10}$$

where $\mathcal{V}$ denotes the set of visual token positions, $\mathbf{h}_i^{\ell}$ the hidden state at position i, and T_i the last token of the full prompt after the chat template, which is not necessarily the last natural-language word of the question. Note that $\mathbf{p}_{\text{vis}}^{\ell}$ averages over all visual tokens rather than classifying each of them separately, and no ground-truth bounding box or part crop is used at any stage. A linear classifier is then trained on each representation with the cross-entropy loss, and its test accuracy measures how much species information is linearly decodable at that depth and position.

Granularity-wise retention ratio. To examine how the dispersion varies with category granularity, we exploit the taxonomic hierarchy of CUB released by Chen et al. (2022), which organizes the 200 species into a four-level structure of order, family, genus, and species. Each training and test image is relabeled at the four granularities, and we repeat the probing protocol above at every depth and position, yielding the accuracy $\mathrm{Acc}_{\ell,g}^{\text{vis}}$ and $\mathrm{Acc}_{\ell,g}^{\text{last}}$ for granularity g at depth ℓ. The retention ratio is then defined as

$$r_{\ell,g} \;=\; \frac{\mathrm{Acc}_{\ell,g}^{\text{last}}}{\mathrm{Acc}_{\ell,g}^{\text{vis}}}, \tag{11}$$

which measures how much of the evidence available at the visual tokens is retained at the answer position. A ratio close to one indicates that the evidence transfers intact, whereas a lower ratio reveals that it has dispersed away before reaching the position that conditions the generation. As the probes at different granularities differ only in the label space, the ratio isolates the effect of granularity from the absolute difficulty of each task.

B Self-Elicited Attribute Annotation

Fine-grained information is already present within the vision-language model; we elicit it using the same model, without introducing external priors. This section describes the annotation protocol and shows the instruction templates and examples for the six fine-grained benchmarks.

B.1 Annotation Protocol

We use the backbone itself as the annotator, so that no knowledge from an external model or human annotator enters the pipeline. For each benchmark, we first define a fixed set of attribute dimensions covering the visual properties that distinguish categories in that domain (e.g., beak shape and wing pattern for birds; headlight and grille shape for cars). The full dimension lists and instruction templates are given in Section B.2.

For each training image, the annotator is prompted to describe the object along several visual dimensions. The instruction explicitly forbids naming or guessing the category, so that the model reports observable appearance rather than recalling a class label. Each response is parsed into a fixed-length vector $a \in \mathbb{R}^D$, where D is the number of attribute dimensions.

We then average the per-image vectors within each training category to obtain a class-level profile q_c, following the construction of the official CUB-200-2011 (Wah et al., 2011) attributes. Each

dimension is standardized across classes by z-scoring. The resulting profiles are used only to compute the soft targets w_{ij}; they are never provided to the model at inference time.

This procedure requires a single forward pass per training image and no external supervision, and it can extend to any fine-grained benchmark by specifying the attribute dimensions for that domain.

B.2 Instruction Templates and Annotation Examples

We use a common instruction template across benchmarks, instantiated with the object type and attribute dimensions of each domain. Below we give the instantiated instruction for each dataset, together with a representative annotation produced under it.

CUB-200-2011 (Wah et al., 2011) contains 11,788 images of 200 bird species, with 5,994 training and 5,794 test images. We annotate $D = 18$ visual attribute dimensions covering plumage, beak, wing, and body shape. The instantiated instruction is shown in Figure 5a, and a representative annotation in Figure 5b.

Instruction Template for CUB-200-2011

System

You are a careful bird morphology annotator. Describe only what is VISIBLE in the image. Do NOT guess the species name. Do NOT invent unseen parts. If a part is occluded, cropped, blurry, or unclear, use value "not_visible". Respond with a single JSON object only — no markdown, no commentary.

User

<image>
Annotate the bird's visible morphological attributes.
Output exactly this JSON schema (all keys required):

```
{
  "body_size": {"value": one of ['tiny', 'small', 'medium', 'large', 'not_visible'], "confidence": 0.0-1.0},
  "body_shape": {"value": one of ['compact', 'elongated', 'stocky', 'slender', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "bill_length": {"value": one of ['short', 'medium', 'long', 'not_visible'], "confidence": 0.0-1.0},
  "bill_shape": {"value": one of ['hooked', 'pointed', 'conical', 'flat', 'curved', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "bill_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "crown_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "head_pattern": {"value": one of ['plain', 'striped', 'spotted', 'capped', 'masked', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "eye_ring": {"value": one of ['present', 'absent', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "throat_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "breast_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "belly_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "back_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "wing_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "wing_pattern": {"value": one of ['plain', 'barred', 'spotted', 'striped', 'multicolored', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "tail_length": {"value": one of ['short', 'medium', 'long', 'not_visible'], "confidence": 0.0-1.0},
  "tail_shape": {"value": one of ['pointed', 'rounded', 'forked', 'square', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "tail_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "leg_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0}
}
```

Rules:
- value must be one of the listed options for that field (or not_visible).
- confidence is your certainty in [0,1]; use low confidence when unsure.
- Prefer coarse colors from the list; use other if none fit.
- Never write species / genus / common names.

(a) Self-elicited attribute instruction for CUB-200-2011. The annotator is constrained to a closed set of visual options per field, may report `not_visible` under occlusion, and is explicitly forbidden from producing species, genus, or common names.

Self-Elicited Fine-Grained Attribute Annotation Examples

CUB-200-2011



Brandt Cormorant


```
{
  "body_size": {"value": "medium", "confidence": 0.8}, "body_shape": {"value": "compact", "confidence": 0.9},
  "bill_length": {"value": "short", "confidence": 0.9}, "bill_shape": {"value": "hooked", "confidence": 0.7},
  "bill_color": {"value": "black", "confidence": 0.9}, "crown_color": {"value": "black", "confidence": 0.9},
  "head_pattern": {"value": "plain", "confidence": 0.9}, "eye_ring": {"value": "present", "confidence": 0.9},
  "throat_color": {"value": "black", "confidence": 0.9}, "breast_color": {"value": "black", "confidence": 0.9},
  "belly_color": {"value": "black", "confidence": 0.9}, "back_color": {"value": "black", "confidence": 0.9},
  "wing_color": {"value": "black", "confidence": 0.9}, "wing_pattern": {"value": "plain", "confidence": 0.9},
  "tail_length": {"value": "short", "confidence": 0.9}, "tail_shape": {"value": "pointed", "confidence": 0.9},
  "tail_color": {"value": "black", "confidence": 0.9}, "leg_color": {"value": "black", "confidence": 0.9}
}
```




Yellow Warbler


```
{
  "body_size": {"value": "medium", "confidence": 0.9}, "body_shape": {"value": "compact", "confidence":
0.9},
  "bill_length": {"value": "short", "confidence": 0.9}, "bill_shape": {"value": "pointed", "confidence": 0.9},
  "bill_color": {"value": "yellow", "confidence": 0.9}, "crown_color": {"value": "yellow", "confidence": 0.7},
  "head_pattern": {"value": "plain", "confidence": 0.9}, "eye_ring": {"value": "present", "confidence": 0.9},
  "throat_color": {"value": "yellow", "confidence": 0.9}, "breast_color": {"value": "yellow", "confidence": 0.9},
  "belly_color": {"value": "yellow", "confidence": 0.9}, "back_color": {"value": "yellow", "confidence": 0.9},
  "wing_color": {"value": "yellow", "confidence": 0.9}, "wing_pattern": {"value": "plain", "confidence": 0.9},
  "tail_length": {"value": "short", "confidence": 0.8}, "tail_shape": {"value": "pointed", "confidence": 0.7},
  "tail_color": {"value": "yellow", "confidence": 0.9}, "leg_color": {"value": "not_visible", "confidence": 0.0}
}
```


(b) Representative annotation produced by the Qwen2.5-VL-3B (Bai et al., 2025b) for images in CUB under the instruction in Figure 5a.

Figure 5: Instruction template and annotation example for CUB-200-2011.

Stanford Dogs (Khosla et al., 2011) contains 20,580 images of 120 dog breeds, with 12,000 training and 8,580 test images. We annotate 11 visual attributes covering body, head, muzzle, ear, and coat, each drawn from a closed set of coarse visual options. The instantiated instruction is shown in Figure 6a, and a representative annotation in Figure 6b.

Instruction Template for Stanford Dogs

System You are a careful dog morphology annotator. Describe only what is VISIBLE. Do NOT guess the breed name. If unclear use "not_visible". JSON only — no markdown.

User

```
<image>
Annotate the dog's visible morphological attributes.
Output exactly this JSON schema (all keys required):
{
  "ear_shape": {"value": one of ['erect', 'drop', 'semi-erect', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "muzzle_length": {"value": one of ['short', 'medium', 'long', 'not_visible'], "confidence": 0.0-1.0},
  "head_shape": {"value": one of ['wedge', 'blocky', 'rounded', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "leg_length": {"value": one of ['short', 'medium', 'long', 'not_visible'], "confidence": 0.0-1.0},
  "eye_shape": {"value": one of ['round', 'almond', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "coat_length": {"value": one of ['short', 'medium', 'long', 'not_visible'], "confidence": 0.0-1.0},
  "coat_texture": {"value": one of ['smooth', 'wiry', 'curly', 'fluffy', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "coat_pattern": {"value": one of ['solid', 'spotted', 'brindle', 'merle', 'bicolor', 'tricolor', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "coat_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "body_size": {"value": one of ['toy', 'small', 'medium', 'large', 'giant', 'not_visible'], "confidence": 0.0-1.0},
  "body_build": {"value": one of ['lean', 'stocky', 'muscular', 'other', 'not_visible'], "confidence": 0.0-1.0}
}
Rules:
- value must be one of the listed options (or not_visible).
- confidence in [0,1]; low when unsure.
- Never write class / species / breed / model names.
```

(a) Self-elicited attribute instruction for Stanford Dogs. As with CUB, the annotator selects from a closed set of coarse visual options, may report `not_visible` under occlusion, and is forbidden from producing breed names.

Self-Elicited Fine-Grained Attribute Annotation Examples

Stanford Dogs



Yorkshire_terrier

```
{
  "ear_shape": {"value": "erect", "confidence": 0.8}, "muzzle_length": {"value": "short", "confidence": 0.9},
  "head_shape": {"value": "rounded", "confidence": 0.9}, "leg_length": {"value": "short", "confidence": 0.8},
  "eye_shape": {"value": "round", "confidence": 0.9}, "coat_length": {"value": "long", "confidence": 0.9},
  "coat_texture": {"value": "wiry", "confidence": 0.9}, "coat_pattern": {"value": "bicolor", "confidence": 0.9},
  "coat_color": {"value": "brown", "confidence": 0.9}, "body_size": {"value": "toy", "confidence": 0.9},
  "body_build": {"value": "lean", "confidence": 0.9}
}
```



Samoyed

```
{
  "ear_shape": {"value": "erect", "confidence": 0.8}, "muzzle_length": {"value": "short", "confidence": 0.9},
  "head_shape": {"value": "rounded", "confidence": 0.9}, "leg_length": {"value": "medium", "confidence": 0.9},
  "eye_shape": {"value": "round", "confidence": 0.9}, "coat_length": {"value": "long", "confidence": 0.9},
  "coat_texture": {"value": "fluffy", "confidence": 0.9}, "coat_pattern": {"value": "solid", "confidence": 0.9},
  "coat_color": {"value": "white", "confidence": 0.9}, "body_size": {"value": "toy", "confidence": 0.9},
  "body_build": {"value": "lean", "confidence": 0.9}
}
```

(b) Representative annotation produced by the Qwen2.5-VL-3B (Bai et al., 2025b) for images in Stanford Dogs under the instruction in Figure 6a.

Figure 6: Instruction template and annotation example for Stanford Dogs.

Stanford Cars (Krause et al., 2013) contains 16,185 images of 196 car models, with 8,144 training and 8,041 test images. We annotate 8 visual attributes covering body type, roofline, headlight and grille shape, wheel design, and color, each drawn from a closed set of coarse visual options. The instantiated instruction is shown in Figure 7a, and a representative annotation in Figure 7b.

Instruction Template for Stanford Cars

System

You are a careful car morphology annotator. Describe only what is VISIBLE. Do NOT guess the make/model/year. If unclear use "not_visible".
JSON only — no markdown.

User

<image>
Annotate the car's visible morphological attributes.
Output exactly this JSON schema (all keys required):

```
{
  "body_type": {"value": one of ['sedan', 'suv', 'coupe', 'hatchback', 'convertible', 'van', 'truck', 'wagon', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "body_size": {"value": one of ['compact', 'midsize', 'fullsize', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "door_count": {"value": one of ['2-door', '4-door', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "grille_shape": {"value": one of ['horizontal', 'mesh', 'vertical', 'none', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "headlight_shape": {"value": one of ['round', 'rectangular', 'angular', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "wheel_style": {"value": one of ['5-spoke', 'multi-spoke', 'solid', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "body_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "roof_type": {"value": one of ['hardtop', 'sunroof', 'convertible', 'other', 'not_visible'], "confidence": 0.0-1.0}
}
```

Rules:
- value must be one of the listed options (or not_visible).
- confidence in [0,1]; low when unsure.
- Never write class / species / breed / model names.

(a) Self-elicited attribute instruction for Stanford Cars. As with the other benchmarks, the annotator selects from a closed set of coarse visual options, may report `not_visible` under occlusion, and is forbidden from producing make or model names.

Self-Elicited Fine-Grained Attribute Annotation Examples

Stanford Cars



Bentley Continental GT Coupe 2007

```
{
  "body_type": {"value": "coupe", "confidence": 0.9}, "body_size": {"value": "fullsize", "confidence": 0.9},
  "door_count": {"value": "other", "confidence": 0.9}, "grille_shape": {"value": "mesh", "confidence": 0.9},
  "headlight_shape": {"value": "round", "confidence": 0.9}, "wheel_style": {"value": "other", "confidence": 0.9},
  "body_color": {"value": "red", "confidence": 0.9}, "roof_type": {"value": "hardtop", "confidence": 0.9}
}
```



Hyundai Sonata Sedan 2012

```
{
  "body_type": {"value": "sedan", "confidence": 0.5}, "body_size": {"value": "midsize", "confidence": 0.5},
  "door_count": {"value": "other", "confidence": 0.5}, "grille_shape": {"value": "horizontal", "confidence": 0.5},
  "headlight_shape": {"value": "round", "confidence": 0.5}, "wheel_style": {"value": "other", "confidence": 0.5},
  "body_color": {"value": "black", "confidence": 0.5},  "roof_type": {"value": "hardtop", "confidence": 0.5}
}
```

(b) Representative annotation produced by the Qwen2.5-VL-3B (Bai et al., 2025b) for images in Stanford Cars under the instruction in Figure 7a.

Figure 7: Instruction template and annotation example for Stanford Cars.

Aircraft (Maji et al., 2013) contains 10,000 images of 100 aircraft variants, with 6,667 training and 3,333 test images. We annotate 10 visual attributes covering fuselage, wing, engine, tail, and landing gear, each drawn from a closed set of coarse visual options. The instantiated instruction is shown in Figure 8a, and a representative annotation in Figure 8b.

Instruction Template for Aircraft

System

You are a careful aircraft morphology annotator. Describe only what is VISIBLE. Do NOT guess the model name. If unclear use "not_visible". JSON only — no markdown.

User

<image>
Annotate the aircraft's visible morphological attributes.
Output exactly this JSON schema (all keys required):

```
{
  "engine_count": {"value": one of ['1', '2', '3', '4', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "engine_placement": {"value": one of ['nose', 'wing', 'rear', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "wing_position": {"value": one of ['low', 'mid', 'high', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "wing_shape": {"value": one of ['straight', 'swept', 'delta', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "tail_type": {"value": one of ['conventional', 't-tail', 'v-tail', 'twin', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "fuselage_length": {"value": one of ['short', 'medium', 'long', 'not_visible'], "confidence": 0.0-1.0},
  "landing_gear": {"value": one of ['retractable', 'fixed', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "prop_or_jet": {"value": one of ['propeller', 'jet', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "body_color": {"value": one of ['black', 'brown', 'white', 'gray', 'yellow', 'red', 'blue', 'green', 'orange', 'buff', 'other', 'not_visible'], "confidence": 0.0-1.0},
  "size_class": {"value": one of ['light', 'regional', 'narrowbody', 'widebody', 'military', 'other', 'not_visible'], "confidence": 0.0-1.0}
}
```

Rules:
- value must be one of the listed options (or not_visible).
- confidence in [0,1]; low when unsure.
- Never write class / species / breed / model names.

(a) Self-elicited attribute instruction for FGVC-Aircraft. As with the other benchmarks, the annotator selects from a closed set of coarse visual options, may report `not_visible` under occlusion, and is forbidden from producing manufacturer, family, or variant names.

Self-Elicited Fine-Grained Attribute Annotation Examples

Aircraft



```
{
  "engine_count": {"value": "4", "confidence": 0.9}, "engine_placement": {"value": "wing", "confidence": 0.9},
  "wing_position": {"value": "mid", "confidence": 0.9}, "wing_shape": {"value": "delta", "confidence": 0.9},
  "tail_type": {"value": "conventional", "confidence": 0.9}, "fuselage_length": {"value": "long", "confidence": 0.9},
  "landing_gear": {"value": "retractable", "confidence": 0.9}, "prop_or_jet": {"value": "jet", "confidence": 0.9},
  "body_color": {"value": "white", "confidence": 0.9}, "size_class": {"value": "widebody", "confidence": 0.9}
}
```

Il-76



```
{
  "engine_count": {"value": "1", "confidence": 0.9}, "engine_placement": {"value": "nose", "confidence": 0.9},
  "wing_position": {"value": "mid", "confidence": 0.9}, "wing_shape": {"value": "delta", "confidence": 0.9},
  "tail_type": {"value": "conventional", "confidence": 0.9}, "fuselage_length": {"value": "long", "confidence": 0.9},
  "landing_gear": {"value": "retractable", "confidence": 0.9}, "prop_or_jet": {"value": "jet", "confidence": 0.9},
  "body_color": {"value": "black", "confidence": 0.9}, "size_class": {"value": "military", "confidence": 0.9}
}
```

Tornado

(b) Representative annotation produced by the Qwen2.5-VL-3B (Bai et al., 2025b) for images in Aircraft under the instruction in Figure 8a.

Figure 8: Instruction template and annotation example for FGVC-Aircraft.

C Implementation Details

C.1 Prompt Templates

All methods share the same instruction template for each dataset, as listed in Tab. 8. The placeholder `<image>` denotes the image input, which is serialized differently across backbones following their official chat templates rather than being passed as a literal string. Since the backbones differ in how strictly they follow the output format, a few evaluations additionally append an explicit output constraint, *e.g.*, *Answer with the scientific species name only* for the iNaturalist subsets, where the

Table 8: Canonical task templates for direct fine-grained recognition. The placeholder `<image>` denotes the image input.

Dataset	System instruction	Question
CUB (Wah et al., 2011)	You are an expert ornithologist specializing in fine-grained bird species recognition. Name the bird's species.	`<image>` What species is this bird?
Stanford Dogs (Khosla et al., 2011)	You are an expert canine specialist specializing in fine-grained dog breed recognition. Name the dog's breed.	`<image>` What breed is this dog?
Stanford Cars (Krause et al., 2013)	You are an expert automotive specialist specializing in fine-grained car model recognition. Name the car's model.	`<image>` What model is this car?
FGVC-Aircraft (Maji et al., 2013)	You are an expert aviation specialist specializing in fine-grained aircraft model recognition. Name the aircraft's model.	`<image>` What model is this aircraft?
iNat21 Fungi & Plantae (Van Horn et al., 2021)	You are an expert naturalist specializing in fine-grained species recognition. Name the organism's scientific species name.	`<image>` What species is this?

models tend to reply with common names. These constraints only affect the output serialization and are applied identically to all compared methods on the same backbone and dataset.

C.2 OASIS

Architecture. The adapter f_ϕ is a two-layer Transformer encoder with the same width d as the backbone hidden space, a feed-forward width of $4d$, pre-layer normalization, GELU activation, and dropout of 0.1. It applies bidirectional self-attention over all valid prompt positions, including both visual and textual tokens. The learnable summary token $\mathbf{u} \in \mathbb{R}^d$ is initialized from $\mathcal{N}(0, 0.02^2)$ and lives only inside the adapter, leaving the backbone sequence length unchanged. The projector g_ψ is a two-layer MLP with GELU, producing ℓ_2-normalized embeddings. The residual projection P_ω is the affine map whose weight and bias are both initialized to zero, so that the backbone output is exactly preserved at initialization. Residuals are added position-wise at prompt positions only, while answer and padding positions remain unchanged.

Training. We use AdamW with a learning rate of 1×10^{-4}, zero weight decay, a linear warmup over the first 3% of steps followed by cosine decay, and train for 3 epochs. The second view is constructed by zeroing a random contiguous span covering $\rho = 15\%$ of the ordered visual-token features before the adapter normalization, which preserves token positions and requires no additional backbone forward pass.

C.3 Compared Methods

Supervised fine-tuning (SFT) We fine-tune all language-model parameters together with the multimodal connector, while keeping the visual encoder frozen. Training uses the direct recognition prompt and the autoregressive cross-entropy on the target answer, with AdamW, a cosine schedule with 3% warmup, zero weight decay, and gradient clipping at 1.0. The effective batch size is 64 for Qwen2.5-VL-3B and Qwen3-VL-4B, and 8 for InternVL2.5-4B and LLaVA-OneVision-7B.

Parameter-matched LoRA. We choose the LoRA rank separately for each backbone to approximately match the trainable parameter budget of OASIS, rather than using a fixed rank across models. LoRA is applied only to the language decoder, targeting the attention projections `q_proj`, `k_proj`, `v_proj`, and `o_proj`, as well as the feed-forward projections `gate_proj`, `up_proj`, and `down_proj`. All pretrained parameters, including the visual encoder and the multimodal connector, remain frozen. We set the scaling parameter to $\alpha = 2r$, use a LoRA dropout of 0.05.

For Qwen2.5-VL-3B and InternVL2.5-4B, we use $r = 58$, yielding 108,509,184 trainable parameters, compared with 109,377,664 for OASIS. For LLaVA-OneVision-7B, we use $r = 133$, yielding 335,577,088 trainable parameters, compared with 334,541,440 for OASIS. The parameter budgets thus differ by less than 1%. Matched-LoRA training follows the same optimizer, schedule, and epochs as above, with effective batch sizes of 64 for Qwen2.5-VL-3B and Qwen3-VL-4B, and 8 for InternVL2.5-4B and LLaVA-OneVision-7B.

Prompt tuning. For Qwen2.5-VL-3B, we prepend K learnable continuous prompt embeddings to the input sequence and freeze all pretrained parameters, following the standard practice of prompt tuning (Lester et al., 2021). The prompt vectors are initialized from sampled vocabulary embeddings. We evaluate $K \in \{20, 100, 200\}$, corresponding to 40,960, 204,800, and 409,600 trainable parameters, respectively, and report for each dataset the highest test accuracy among the three prompt lengths. This best-over-K reporting provides a favorable comparison for prompt tuning rather than selecting K on a held-out validation set.