Self-Empowering VLMs: Achieving Hierarchical Consistency via Self-Elicited Knowledge Distillation

Wei Yang[1,*], Yiran Zhu[1,*], Zilin Li[2], Xunjia Zhang[1], and Hongtao Wang[1,†]
[1]Department of Computer, North China Electric Power University, Baoding, China
[2]School of Information and Intelligent Science, Donghua University, Shanghai, China

Abstract

Vision-language models (VLMs) possess rich knowledge but often fail on hierarchical understanding tasks, where the goal is to predict a coarse-to-fine taxonomy path that remains consistent across all levels. We compare three inference paradigms for hierarchical VQA and find that stepwise reasoning, when conditioned on prior answers, significantly outperforms single-pass prompting. Further analysis indicates that the main limitation of current VLMs is their inability to maintain cross-level state, rather than a lack of taxonomic knowledge. Motivated by this diagnosis, we propose Self-Elicited Knowledge Distillation (SEKD), which requires no human labels or external tools: the same VLM is prompted to reason step by step and act as a teacher by exposing its hard labels, soft distributions, and decoder hidden states, while a single-pass student distills these signals. The student VLM remains efficient while approaching the accuracy of its multi-step teacher. It improves in-domain path consistency (HCA) by up to +29.50 percentage points, raises zero-shot HCA on an unseen taxonomy from 4.15% to 42.26%, and yields gains on challenging mathematical benchmarks. Because all supervision is self-elicited, SEKD scales to new taxonomies and datasets without annotation cost, providing a practical route to imbue compact VLMs with dependency-aware multi-step reasoning.

1. Introduction

Structured, multi-step reasoning is central to many AI problems, including perceiving scenes by identifying parts and relations, answering compositional queries via intermediate deductions, and following instructions by decomposing goals into ordered substeps[12]. In contrast to multi-step tasks like mathematical problem solving[9], hierarchical understanding offers an intuitive and comprehensive testbed: at each step of the reasoning path there is a uniquely defined target node, and under a fixed taxonomy there is only one correct root-to-leaf path, enabling precise measurement of both per-level accuracy and path-wise consistency[4, 6]. We focus on a hierarchical setting: given an image, the model answers a sequence of taxonomic questions whose answers form a coarse-to-fine path in an explicit tree (e.g., for a cat image in Fig. 1, L1 kingdom animalia, L2 phylum chordata, ..., L7 species felis catus). Each prediction must be consistent with its ancestors, since fine-grained decisions are valid only when higher-level predictions are correct. This requirement goes beyond taxonomies themselves, capturing multi-question consistency and the coherence of intermediate states in complex reasoning, thus shaping the model's multi-step reasoning ability.

Building on this hierarchical testbed, we uncover a severe failure mode: when confronted with multi-level queries, even strong VLMs rarely produce answers that are both accurate and consistent; accuracy degrades with depth, and path-level inconsistencies are common, undermining the reliability of full-path predictions. This leads to our central question: do VLMs lack taxonomic knowledge, or do they fail to organize it when tasks require structured, multi-step, dependency-aware reasoning?

To probe this question, we design and compare three paradigms for this task (see Sec. 2). When we decompose the task by hierarchy and require the model to strictly respect ancestor decisions across multiple calls, inconsistencies are substantially reduced and HCA improves. Further analysis indicates that current VLMs do not organize dependency-aware reasoning and struggle to maintain cross-level state under flat prompting, whereas stepwise conditioning provides the missing state-carrying scheme.

We trace this failure to an essentially monotone pre-training objective: training is dominated by single-turn next-token prediction, which biases VLMs toward myopic, single-pass solutions. Under flat prompting, related levels

*Co-first authors, contributed equally. They jointly conducted the literature review, designed and refined the method, and co-designed the experimental protocol. Wei Yang focused more on implementation and running experiments, as well as writing and polishing the experimental-details sections. Yiran Zhu focused more on refining the methodology, shaping the overall narrative and structure, and preparing the figures.

†Corresponding author. Contact email: wanght@ncepu.edu.cn

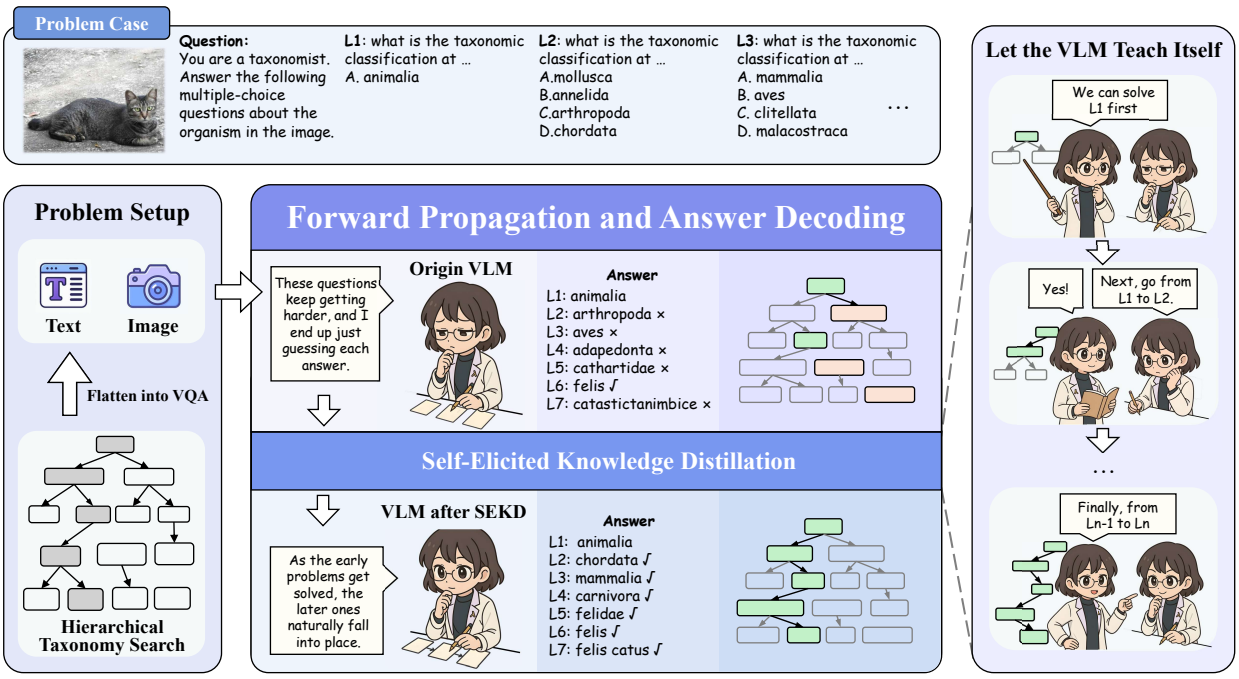

Figure 1. Overview of our hierarchical VQA formulation and Self-Elicited Knowledge Distillation (SEKD) framework.

are queried independently, so the model does not maintain the cross-level state that should flow from coarse to fine. The issue is not a lack of factual knowledge but the absence of an actively engaged, stepwise reasoning process. Rather than relying on supervised fine-tuning (which requires extra labels and risks catastrophic forgetting), pre-training redesigns [10, 21] (data- and compute-intensive), tool-augmented pipelines [38, 39] (brittle and high-latency), or long-chain test-time reasoning [35] (slow and high-variance), we aim to awaken the VLM's step-by-step reasoning by enforcing a prescribed process. In principle this requires no labels or external signals, but it raises a central question: how can we strengthen this ability without relying on external tools or supervision?

Our diagnostic experiments suggest a way forward. When we force the model to strictly respect the decision from the previous level and continue reasoning, repeatedly querying the model spends substantial compute but yields responses that are both accurate and highly consistent. In this paper, we treat this behavior as a training signal to strengthen the VLM itself, and operationalize it as Self-Elicited Knowledge Distillation (SEKD). In SEKD, the same VLM plays both teacher and student: the teacher is invoked multiple times so that its stepwise reasoning becomes explicit, and the student is trained to match the teacher's predictions and internal feature distributions. This yields a simple plug-in recipe that transfers structured, dependency-aware reasoning into a compact single-pass model.

SEKD yields large gains and strong generalization: it improves HCA on ImageNet-Animal [6] by 29.5 points and raises zero-shot HCA on the unseen Food101 taxonomy from 4.15% to 42.26% [2]. It also improves zero-shot performance on GQA [11] and its benefits transfer to challenging mathematical reasoning benchmarks. Guided by these results, we summarize our contributions as follows:

- Hierarchical consistency diagnostic protocol. We propose a diagnostic protocol that turns intuitive observations about flat-prompting failures into reproducible, quantitative assessments of hierarchical consistency, revealing that the bottleneck lies in maintaining cross-level state rather than in missing taxonomic knowledge.
- Self-Elicited Knowledge Distillation method. We propose a simple, data-efficient, label-free training scheme in which the same VLM acts as a stepwise teacher and a single-pass student, distilling logits, soft distributions, and decoder states to internalize dependency-aware reasoning without external supervision.
- Practical, generalizable, and stable reasoning. SEKD yields an efficient single-pass model that rivals larger baselines, transfers hierarchical understanding to unseen taxonomies and out-of-domain benchmarks, and maintains general VQA performance without forgetting.

2. Why VLMs Fail at Hierarchical VQA

2.1. Diagnostic protocol

To isolate the effect of the inference paradigm on hierarchical VQA, we conduct a controlled evaluation that changes

Table 1. Diagnostic comparison under matched compute (Dataset: iNat-Animal, Model: LLaVA-OV-7B). All metrics are percentages (%). Parentheses indicate absolute change vs. Joint Invocation. Additional diagnostic results are provided in Appendix D.

Protocol	HCA	LeafAcc	TOR	POR	S-POR
Joint Invocation	0.05	25.35	11.92	38.06	20.59
Independent Levels	4.60 (**+4.55**)	27.45 (**+2.10**)	51.93 (**+40.01**)	65.92 (**+27.86**)	55.43 (**+34.84**)
Conditioned Steps	8.85 (**+8.80**)	32.45 (**+7.10**)	55.66 (**+43.74**)	67.69 (**+29.63**)	57.71 (**+37.12**)

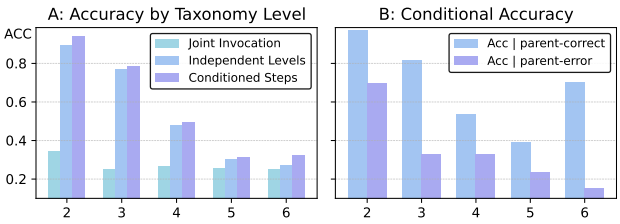

Figure 2. Depth-wise per-level accuracy and conditional accuracy.

only the paradigm while holding all other conditions fixed, and we require models to produce a complete taxonomy path for each image. On the same images and target paths, we compare three protocols: (i) Joint Invocation, where the model is invoked once and autoregressively emits all levels in a single turn; (ii) Independent Levels, where the image and the level-specific question are issued separately for each level with no cross-level conditioning; and (iii) Conditioned Steps, where each level conditions on the previous level's predicted answer as a known fact. We report five metrics: Hierarchical-Consistency Accuracy (HCA), Position Overlap Ratio (POR), Strict Position Overlap Ratio (S-POR), Transition-Order Ratio (TOR), and Leaf Accuracy (LeafAcc). Formal definitions are deferred to Sec. 4.1.

2.2. Core findings

We evaluate LLaVA-OV-7B [14] on iNat-Animal [31]. As shown in Tab. 1, Joint Invocation is weak (HCA 0.05%, LeafAcc 25.35%). Independent Levels improves but remains limited (HCA +4.55 pp; LeafAcc +2.10 pp). In contrast, Conditioned Steps yields the largest gains (HCA +8.80 pp; LeafAcc +7.10 pp), and TOR / POR / S-POR each increase by 30–44 pp. These gains cannot be explained by scaling alone; they arise from explicitly conditioning on the previous level's answer. Propagating the parent decision to the next step restricts the candidate classes to the corresponding subtree and imposes a path-level prior, which jointly improves cross-level coherence (HCA) and leaf accuracy, whereas independent per-level answering lacks such constraints and struggles to maintain global consistency.

2.3. Depth-wise and conditional analyses

To locate where and how the gains arise, we analyze performance by depth and conditioning. Because the first level in this dataset has no parent, depth-wise metrics are undefined at that level; we therefore report results for levels 2–6.

Fig. 2A shows that Conditioned Steps consistently outperforms Independent Levels and Joint Invocation across depths, and achieves a larger margin over Independent Levels at certain levels. Fig. 2B characterizes parent-child coupling via conditional accuracy: we define $\Delta_\ell = \mathrm{Acc}_{\ell+1|\mathrm{correct}_\ell} - \mathrm{Acc}_{\ell+1|\mathrm{error}_\ell}$ and a larger Δ_ℓ indicates that the correctness of the parent has a stronger influence on the child. Consistent with Fig. 2A, Conditioned Steps obtains noticeably larger gains over Independent Levels precisely at these high-coupling levels. Intuitively, conditioning propagates the parent decision as explicit state to the next level, restricting candidate classes to the corresponding subtree and inducing a path-level prior; at strongly coupled levels, this constraint more effectively shrinks the search space and improves per-level accuracy, while the inherited state also increases cross-level consistency, leading to higher TOR and, in turn, substantial gains in HCA.

2.4. Takeaway and design implication

Taken together, our diagnosis is that current VLMs are limited not by missing taxonomic knowledge but by an inability to organize interdependent sub-questions into a stepwise reasoning process, leading to depth-dependent degradation and cross-level inconsistency. In contrast, Conditioned Steps propagates parent decisions downward, supplying selected-branch context and jointly improving TOR and HCA. Guided by this insight, we first run explicit, parent-conditioned stepwise reasoning to extract cross-level signals, then use self-elicited distillation to distill these signals into a single-pass model. The resulting model internalizes the "choose the branch, then specialize" mechanism as a representation and, without additional inference latency, simultaneously improves TOR, HCA, and LeafAcc.

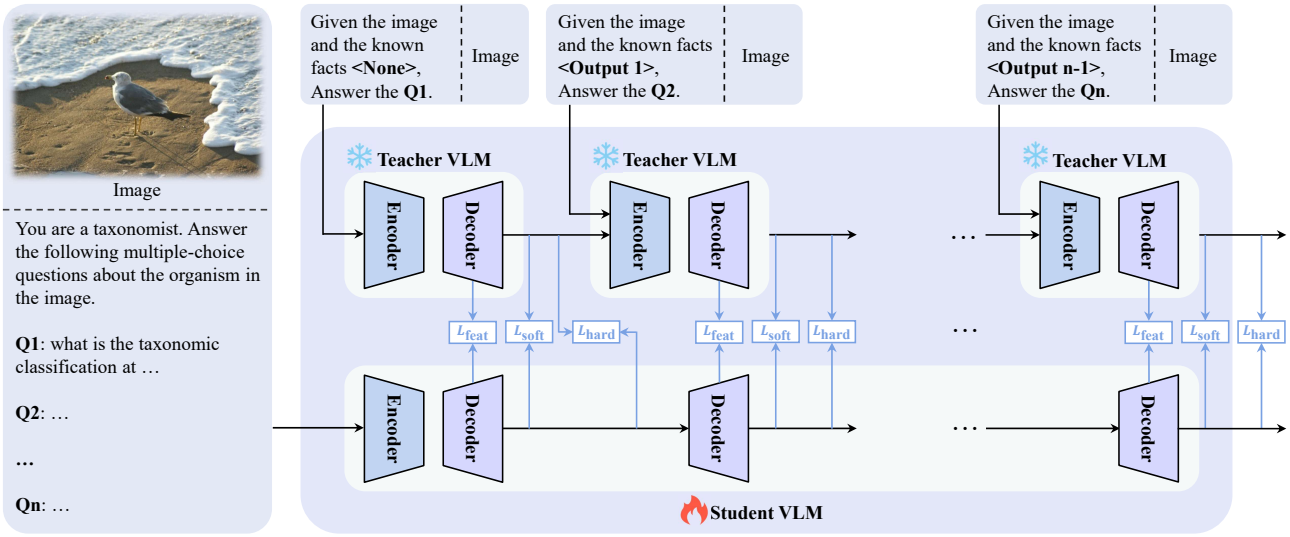

Figure 3. SEKD training architecture: a stepwise teacher VLM distills its intermediate hierarchical states into a single-pass student.

3. Self-Elicited Knowledge Distillation (SEKD)

3.1. Problem Setup

We study hierarchical visual recognition defined over a fixed taxonomy tree $\mathcal{T}$ of depth L. For each image x, the prediction target is a root-to-leaf category sequence $y_{1:L} = (y_1, \ldots, y_L)$, where y_l denotes the category at level l of the taxonomy. Each level l is associated with a multiple-choice query q_l. In our evaluation protocol, we adopt a *joint questioning* setting: the model receives the image together with all level-wise queries $(q_1, \ldots, q_L)$ in a single prompt, and is required to output a sequence of answers $a_{1:L} = (a_1, \ldots, a_L)$. As illustrated in Fig. 3, the model is instructed as a taxonomist and, given the image and the joint queries, must select one option per level. This interface makes the coarse-to-fine structure explicit and enables us to measure both per-level accuracy and path-level consistency of the predicted taxonomy path.

Within this setup, we use the same VLM in two distinct roles: a teacher model t_ϕ and a student model s_θ. The teacher model t_ϕ follows the Conditioned Steps paradigm: given the same set of hierarchical queries $(q_1, \ldots, q_L)$, it conceptually answers them level by level, producing an answer a_l for each level and injecting the previous prediction into the prompt of the current level as a known fact. The student model s_θ follows the joint questioning protocol described above: it receives the image and all queries in a single input, and directly generates the complete answer sequence $a_{1:L}$ in one forward pass.

3.2. Conditioned Steps Teacher

This section focuses on the behavior of the teacher model t_ϕ under the Conditioned Steps paradigm. The teacher is instantiated directly from an officially released pretrained vision–language model and remains fully frozen during our training; it only participates via forward passes to produce per-level labels and features, which are subsequently used to distill the student model.

Given an image x and the hierarchical queries $(q_1, \ldots, q_L)$ defined in Sec. 3.1, we query t_ϕ level by level. As shown in Fig. 3, at level l, the input to the teacher is a context that combines the image, the current query, and all previous answers,

$$c_l = \mathrm{Input}\big(x, q_l, a_{1:l-1}\big), \tag{1}$$

where $a_{1:l-1}$ denotes the textual answers from levels 1 to $l-1$ (empty when $l = 1$). In practice, $a_{1:l-1}$ is inserted into the prompt as "known facts", so the decision at level l is always conditioned on the image together with all earlier commitments along the taxonomy path, enabling the model to disambiguate progressively finer-grained categories.

With the level-l input c_l defined above, we feed it into the teacher model t_ϕ. Let $\mathcal{A}_l$ denote the option set at level l (e.g., letters A/B/C/…). A single forward pass of the teacher produces a probability distribution over options, which we denote by $p_l^{(T)}(\alpha)$ for $\alpha \in \mathcal{A}_l$, where the superscript (T) indicates that the distribution is produced by the teacher. The hard label $\tilde{y}_l$ is then defined as the most probable option, whose textual form a_l is inserted into the next-level prompt as a known fact.

For the feature signal, we focus on the internal representation associated with the token that actually expresses the

decision at level l. When the teacher generates the level-l answer in the decoder, it produces a sequence of final-layer hidden states $H_l^{(T)} = (h_1, \ldots, h_{T_l})$ where h_t denotes the decoder hidden state at time step t just before predicting the t-th output token. Among these output tokens, exactly one corresponds to the option selected from $\mathcal{A}_l$ (e.g., the token "A"). We denote the position of this option token in the output sequence by anchor$_l$. We define the level-l feature as the hidden state at this decision token, summarizing the model's state when committing to the option: $h_l^{(T)} = h_{\text{anchor}_l}$.

Running $l = 1 \to L$ produces $\{(\tilde{y}_l, p_l^{(T)}, h_l^{(T)})\}_{l=1}^{L}$, which will be used as process-level supervision to train the student s_θ under joint questioning, encouraging it to internalize cross-level state and maintain path consistency.

3.3. Joint Invocation Student

Unlike the teacher t_ϕ, the student model s_θ is the trainable component, initialized from the same pretrained VLM as the teacher. Under the Joint Invocation setting, s_θ receives the image together with all level-wise queries $(q_1, \ldots, q_L)$ in a single input and produces the entire answer sequence in one forward pass. As shown in Fig. 3, decoding is autoregressive and uses only the student's own previously generated tokens as context. Let $\hat{a}_l$ denote the option token predicted by the student at level l. We write

$$\hat{a}_l \sim P_\theta(\cdot \mid x, q_{1:l}, \hat{a}_{1:l-1}), \quad l = 1, \ldots, L, \tag{2}$$

where P_θ is the conditional distribution defined by s_θ. This yields the decision chain $a_1 \to a_2 \to \cdots \to a_L$.

Using the same option set $\mathcal{A}_l$ and anchor definition as in Sec. 3.2, the student outputs a triplet $(\hat{y}_l, p_l^{(S)}, h_l^{(S)})$ per level l, mirroring the teacher's signals: the predicted hard label $\hat{y}_l$, the soft probabilities $p_l^{(S)}$, and the feature at the decision token $h_l^{(S)}$ read from $H_l^{(S)}$ at anchor$_l$. Collecting across levels gives $\{(\hat{y}_l, p_l^{(S)}, h_l^{(S)})\}_{l=1}^{L}$, which provides a one-to-one alignment with the teacher-side sequence $\{(\tilde{y}_l, p_l^{(T)}, h_l^{(T)})\}_{l=1}^{L}$ for the losses defined in Sec. 3.4.

3.4. Distillation Objectives: Teaching Itself

Building on the teacher- and student-side signals defined in Secs. 3.2 and 3.3, this section couples them through training. At each level l, we align the student's $(\hat{y}_l,\ p_l^{(S)},\ h_l^{(S)})$ with the teacher's $(\tilde{y}_l,\ p_l^{(T)},\ h_l^{(T)})$ at anchor$_l$. To achieve SEKD's "Teaching Itself" objective for training s_θ under joint questioning, we employ three complementary losses:

(a) Hard-label loss.

$$\mathcal{L}_{\text{hard}} = \sum_{l=1}^{L} \text{CE}\big(p_l^{(S)}, \tilde{y}_l\big), \tag{3}$$

where CE is the cross-entropy loss. Hard labels make the student commit to a discrete choice at each level, aligning its predictions with the teacher's stepwise decisions.

(b) Soft-logit loss.

$$\mathcal{L}_{\text{soft}} = \sum_{l=1}^{L} \text{KL}\big(p_l^{(T)} \,\|\, p_l^{(S)}\big), \tag{4}$$

where KL denotes the Kullback–Leibler divergence. Soft logits convey the teacher's relative preferences over options, which is especially helpful at deeper levels where fine-grained confusions are common.

(c) Stepwise feature loss.

$$\mathcal{L}_{\text{feat}} = \sum_{l=1}^{L} \big\| W h_l^{(S)} - h_l^{(T)} \big\|_2^2, \tag{5}$$

where W is a linear projector that maps student features into the teacher space. This feature loss encourages the student to maintain a state similar to the teacher's at the decision locations, complementing the distributional imitation losses.

The total objective is:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{hard}} + \lambda_2 \mathcal{L}_{\text{soft}} + \lambda_3 \mathcal{L}_{\text{feat}}. \tag{6}$$

where λ_1, λ_2, and λ_3 are the weights for the corresponding loss. Losses are accumulated only at the constrained answer-token positions rather than over the entire sequence. This focuses learning on hierarchy-bearing tokens and reduces spurious gradients from instruction text, improving optimization stability in practice.

Table 2. Dataset statistics. # Levels: depth of hierarchy; # Items: total items in the benchmark paper.

Type	Dataset	# Levels	# Items
Internal	CUB-200-2011 [32]	4	5,794
	iNaturalist-Plant [31]	6	42.71K
	iNaturalist-Animal [31]	6	53.88K
	ImageNet-Animal [6]	11	19.85K
	ImageNet-Artifact [6]	8	24.55K
	Food-101 [2]	4	21.00K
External	GQA (test-dev) [11]	–	∼113K
	MathVista [22]	–	6,141
	MMBench [20]	–	2,974

4. Experiments

We evaluate SEKD along three axes. We measure its effect on in-domain HCA. We test generalization, asking whether it learns a transferable structured reasoning skill rather than memorizing data, using unseen taxonomies and external reasoning tasks. Finally, we assess utility and efficiency via data-efficiency ablations, stability tests (catastrophic forgetting), and a challenging "7B vs. 32B" comparison.

Table 3. Main performance of SEKD on the ImageNet-Animal test set (models trained on ImageNet-Animal). All metrics are percentages (%). Numbers in parentheses are absolute gains over the *Base* model.

Model	Compare	HCA	LeafAcc	TOR	POR	S-POR
LLaVA-OV-7B	*Base*	0.65	26.85	17.10	40.95	25.16
	Teacher	33.40 (**+32.75**)	64.40 (**+37.55**)	74.76 (**+57.66**)	84.37 (**+43.42**)	73.55 (**+48.39**)
	Student	30.15 (**+29.50**)	62.30 (**+35.45**)	74.33 (**+57.23**)	84.16 (**+43.21**)	73.65 (**+48.49**)
InternVL2.5-8B	*Base*	10.20	43.75	39.71	59.57	40.96
	Teacher	35.85 (**+25.65**)	64.50 (**+20.75**)	77.13 (**+37.42**)	85.35 (**+25.78**)	78.16 (**+37.20**)
	Student	37.35 (**+27.15**)	65.20 (**+21.45**)	78.53 (**+38.82**)	86.41 (**+26.84**)	79.42 (**+38.46**)
InternVL3-8B	*Base*	6.50	32.60	24.60	45.37	32.61
	Teacher	36.45 (**+29.95**)	68.60 (**+36.00**)	76.72 (**+52.12**)	85.58 (**+40.21**)	76.84 (**+44.23**)
	Student	23.95 (**+17.45**)	54.25 (**+21.65**)	59.21 (**+34.61**)	72.11 (**+26.74**)	62.43 (**+29.82**)
Qwen2.5-VL-7B	*Base*	33.70	69.85	67.43	80.04	66.06
	Teacher	48.65 (**+14.95**)	77.65 (**+7.80**)	82.57 (**+15.14**)	89.48 (**+9.44**)	82.38 (**+16.32**)
	Student	49.80 (**+16.10**)	78.70 (**+8.85**)	83.38 (**+15.95**)	90.08 (**+10.04**)	83.30 (**+17.24**)

4.1. Experimental Setup

Datasets. Our hierarchical evaluation relies on the benchmarks and taxonomic structures introduced by Tan et al. [30]. We adopt their defined hierarchical levels. Our internal sets are iNaturalist (iNat-Animal, iNat-Plant) [31], ImageNet (Animal, Artifact) [6], CUB-200-2011 [32], and Food-101 [2]. For all internal datasets, we split the total items 6:2:2 into train/val/test. Unlike the per-level evaluation in Tan et al., we adopt a stricter single-invocation, multi-question protocol that tests whether a model maintains state across all levels in one pass. We also use Food-101 for internal zero-shot, GQA (test-dev) and MathVista [22] for external generalization, and MMBench [20] for stability. Dataset statistics are in Tab. 2.

Evaluation Protocol. All evaluations use our single-invocation, multi-question setting. For external datasets, we report standard accuracy, absolute drop Δpp, and relative rate r. For hierarchical datasets, we adopt the five path-level metrics of Tan et al. [30]: HCA (strict full-path accuracy), LeafAcc (leaf-node accuracy), POR (average per-level accuracy), TOR (accuracy of parent–child transitions), and S-POR (normalized length of the longest correct prefix from the root). Formal definitions are given in Appendix A.

Implementation Details. We implement SEKD using PyTorch on NVIDIA L20 and H20 GPUs. The base models evaluated include LLaVA-OV-7B [14], InternVL2.5-8B [5], InternVL3-8B [44], and Qwen2.5-VL-7B [1]; "Base" refers to the zero-shot performance of these official checkpoints without modification. For all models, the vision encoder is frozen and we fine-tune only the LLM side with LoRA (rank $r = 64$, $\alpha = 16$, dropout 0.05), applied to all query, key, value, and output projection layers. Images are resized so that the long side is 448 px and then center-cropped or padded. Training uses the AdamW optimizer ($\beta = (0.9, 0.999)$, weight decay $= 0.01$) with a learning rate of 2×10^{-5}, a cosine decay schedule with 3% linear warmup, and FP16 precision. The total distillation loss uses fixed weights $\lambda_1 = 2.0$, $\lambda_2 = 1.0$, and $\lambda_3 = 0.5$. We use an effective batch size of 1 via gradient accumulation, gradient clipping at 1.0, and train for at most 3 epochs.

4.2. SEKD Hierarchical Consistency Evaluation

We evaluate four VLLMs on the ImageNet-Animal benchmark; Tab. 3 reports results for the Base, Teacher, and SEKD Student. Under the single-invocation setting, the Base models nearly fail at path-level reasoning: for example, HCA is 0.65% for LLaVA-OV-7B and 6.50% for InternVL3-8B, despite reasonable leaf accuracy. The multi-step Teacher, obtained by running Conditioned Steps, largely repairs this, yielding large absolute gains in HCA and consistency metrics (e.g., LLaVA's HCA rises to 33.40% and TOR to 74.76%). The SEKD Student recovers most of this improvement while remaining single-pass: for LLaVA, it reaches 30.15% HCA and 74.33% TOR, with similar patterns for InternVL2.5, InternVL3, and Qwen2.5-VL. These gains are not merely leaf-level but reflect a systematic repair of path coherence, as seen in boosts to TOR, POR, and S-POR. The Base–Teacher–Student gap supports our diagnosis: off-the-shelf VLLMs lack cross-level state in a single invocation, the multi-step Teacher exposes the missing dependency-aware reasoning, and SEKD distills this behavior into an efficient forward pass. Further hierarchical consistency results are reported in Appendix E.1.

4.3. Generalization Analysis

To verify that SEKD imparts a transferable "structured reasoning ability" rather than simply overfitting to the training

Table 4. SEKD generalization on an internal dataset and external benchmarks. Food-101 is an internal zero-shot taxonomy; we report HCA and LeafAcc as mean ± std over students trained on five internal taxonomies. GQA and MathVista are external benchmarks; for these, numbers in parentheses for *Student* rows denote absolute gains (percentage points) over the corresponding *Base* model, whereas for Food-101 they denote standard deviations.All metrics are percentages (%). Full results for additional backbones are in the Appendix E.2.

Model	Compare	Food-101 (zero-shot)		GQA	MathVista		
		HCA	LeafAcc	Acc	Average	Arithmetic	Geometry
InternVL3-8B	*Base*	7.85	27.00	46.66	64.90	62.04	70.29
InternVL3-8B	*Student*	24.43 (±**7.14**)	56.18 (±**12.11**)	53.05 (**+6.39**)	65.10 (**+0.20**)	62.61 (**+0.57**)	71.55 (**+1.26**)
Qwen2.5-VL-7B	*Base*	16.70	59.85	58.83	64.80	60.06	63.18
Qwen2.5-VL-7B	*Student*	43.39 (±**1.59**)	90.09 (±**0.35**)	66.09 (**+7.26**)	65.20 (**+0.40**)	60.62 (**+0.56**)	66.11 (**+2.93**)

Table 5. Ablation on loss components. All values are percentages (%). Deltas in parentheses are vs. *Full SEKD*. Complete ablation results are provided in Appendix F.

Variant	HCA	LeafAcc	TOR	POR	S-POR
Base	2.95	30.15	27.64	48.94	33.56
Full SEKD	17.15	41.75	64.55	74.35	66.74
w/o $\mathcal{L}_{hard}$	14.60 (**-2.55**)	41.30 (**-0.45**)	56.36 (**-8.19**)	68.76 (**-5.59**)	59.01 (**-7.73**)
w/o $\mathcal{L}_{feat}$	15.35 (**-1.80**)	42.65 (**+0.90**)	62.54 (**-2.01**)	73.45 (**-0.90**)	64.36 (**-2.38**)
w/o $\mathcal{L}_{soft}$	15.55 (**-1.60**)	43.85 (**+2.10**)	61.47 (**-3.08**)	72.70 (**-1.65**)	63.68 (**-3.06**)

data, we conduct a series of rigorous generalization tests.

Internal Zero-Shot Generalization. We first test zero-shot transfer to an unseen taxonomy, Food-101. Students trained on five internal taxonomies are directly evaluated on Food-101 without any fine-tuning. As shown in Tab. 4, we report the Base performance together with the mean ± std of the corresponding Students. Both InternVL3-8B and Qwen2.5-VL-7B show low base HCA (7.85% and 16.70%), while SEKD Students achieve much stronger consistency (24.43% and 43.39%), along with large gains in leaf accuracy. This indicates that SEKD learns a general structured reasoning strategy that transfers to a new taxonomy.

External Task Generalization. We examine whether this internalized state-maintenance skill carries over to non-hierarchical reasoning benchmarks. On GQA, SEKD improves accuracy over the Base models (InternVL3-8B: 46.66% → 53.05%, +6.39pp; Qwen2.5-VL-7B: +7.26pp), suggesting that cross-question state learned from hierarchical paths benefits compositional VQA. On MathVista, the Students yield modest but consistent gains in overall accuracy (InternVL3-8B: +0.20pp; Qwen2.5-VL-7B: +0.40pp), with larger improvements on specific step-wise skills: both models improve on Arithmetic and Geometry (e.g., Qwen2.5-VL-7B: +0.56pp on Arithmetic, +2.93pp on Geometry). Together, these results suggest that the step-by-step process induced by taxonomies partially transfers to broader logical and numerical reasoning tasks.

Table 6. Few-shot efficiency on iNat-Animal (InternVL3-8B). All values are percentages (%). Metrics are reported as mean ± std over five random seeds.

# Samples	HCA	LeafAcc	TOR	POR	S-POR
10,000	17.39 (±**0.55**)	43.37 (±**1.13**)	63.64 (±**1.86**)	73.83 (±**1.30**)	65.72 (±**1.54**)
5,000	13.46 (±**0.78**)	45.15 (±**1.57**)	57.36 (±**2.32**)	70.44 (±**1.78**)	59.79 (±**1.95**)
2,000	10.59 (±**1.62**)	41.21 (±**2.93**)	49.54 (±**6.92**)	64.62 (±**5.44**)	52.98 (±**5.68**)
1,000	9.09 (±**1.23**)	37.88 (±**1.54**)	43.64 (±**4.42**)	59.88 (±**3.35**)	48.17 (±**3.67**)
500	8.29 (±**1.64**)	36.98 (±**2.53**)	39.52 (±**7.19**)	56.85 (±**5.49**)	44.71 (±**5.98**)
200	8.03 (±**0.76**)	36.36 (±**0.96**)	40.33 (±**2.73**)	57.35 (±**2.08**)	45.34 (±**2.25**)

4.4. Ablation Studies

To verify the contribution of the components and its data efficiency in few-shot scenarios, we conduct ablation studies on the iNat-Animal dataset using InternVL3-8B.

Loss Function Ablation. Tab. 5 ablates the three SEKD losses: $\mathcal{L}_{hard}$, $\mathcal{L}_{soft}$, and $\mathcal{L}_{feat}$. All ablated variants (HCA $\geq$ 14.60%) still dramatically outperform the Base model (2.95% HCA), confirming the strength of the distillation scheme itself. Among the components, dropping $\mathcal{L}_{hard}$ hurts most, reducing HCA from 17.15% to 14.60% (-2.55pp), showing that matching the teacher's final decisions is crucial for learning the hierarchy. Removing $\mathcal{L}_{feat}$ gives the next largest drop (-1.80pp), indicating that aligning intermediate states helps internalize the stepwise process. Removing $\mathcal{L}_{soft}$ has the smallest but still nontrivial effect (-1.60pp), suggesting that all three terms contribute and jointly yield the best performance.

Few-Shot Efficiency. Tab. 6 illustrates the data efficiency of SEKD, plotting HCA performance against the number of training samples. We report the mean and standard deviation across five random seeds [42, 21, 87, 13,

Table 7. Efficiency comparison between a 7B SEKD student and a 32B base model (Qwen 2.5-VL). All values are percentages (%). Parentheses show the difference of 7B relative to 32B (in pp).

Dataset	Model	HCA	LeafAcc
iNat-Animal	32B base	22.25	55.55
	7B student	20.50 **(-1.75)**	48.75 **(-6.80)**
iNat-Plant	32B base	19.20	52.00
	7B student	20.30 **(+1.10)**	48.95 **(-3.05)**
ImgNet-Animal	32B base	49.45	78.20
	7B student	49.80 **(+0.35)**	78.70 **(+0.50)**
ImgNet-Artifact	32B base	18.45	83.00
	7B student	13.05 **(-5.40)**	81.45 **(-1.55)**

Table 8. Practicality & efficiency comparison at batch size=1.

Model	Parameters	Peak VRAM (GB)	Throughput (samples/sec)
32B Base	32B	62.72	1.16
7B Student	7B	15.65	3.26
Improvement	–	4.0× Reduction	2.81× Speedup

100] for each sample size. The results are highly encouraging: even with only 200 training samples, the SEKD Student (HCA: 8.03 ± 0.76%) already achieves a +5.08pp gain over the full-data Base Model (HCA: 2.95%). This demonstrates that SEKD is not simply overfitting but is capable of imparting the structural reasoning process with remarkable data efficiency. Performance scales smoothly with data, reaching 17.39 ± 0.55% HCA at 10,000 samples.

4.5. Practicality and Efficiency Analysis

To evaluate efficiency, we compare our distilled Qwen2.5-VL-7B *Student* against the larger Qwen2.5-VL-32B *Base*. As shown in Tab. 7 and Tab. 8, the 7B Student is more practical, requiring $4.0\times$ less VRAM (15.65 GB), enabling deployment on 16–24 GB GPUs instead of 80 GB-class accelerators and delivers $2.81\times$ higher throughput, without catastrophic performance loss. The 7B Student remains competitive, matching or even surpassing the 32B Base on ImgNet-Animal (HCA: $+0.35$ pp) and iNat-Plant (HCA: $+1.10$,pp), while the 32B Base retains a clear advantage on ImgNet-Artifact (HCA: -5.40,pp), likely due to its stronger raw visual knowledge for fine-grained artifact recognition. Overall, SEKD is an effective capability-transfer method whose realized gains are modulated by the student's base visual capacity; moreover, results on MMBench indicate that SEKD maintains overall VQA performance without catastrophic forgetting (see Appendix E.2 for details).

5. Related Work

Hierarchical Understanding in Vision and VQA. In hierarchical understanding for vision and VQA, most prior work hard-wires the taxonomy into the model via specialized losses, representations, or architectures. Typical strategies include hierarchy-aware training objectives that enforce tree-path consistency [25], hyperbolic or graph-based label embeddings that preserve hierarchical margins [24, 36, 37], and modules that jointly refine coarse and fine predictions to couple decisions across levels [19, 33]. While these approaches improve path consistency, they require per-level supervision and taxonomy-specific components, whereas we leave the base VLM architecture unchanged and instead rely on stepwise reasoning so that hierarchical state is formed and maintained internally.

Multi-step reasoning and process supervision. From this perspective, many works study how to guide models through intermediate reasoning. One line makes the process explicit: models are prompted to "think step by step" and complex questions are decomposed into sub-tasks, while tool- or agent-based pipelines insert external modules such as retrieval and calculators into the reasoning chain [10, 38]. These procedures improve interpretability but increase test-time latency. Another line supervises the process implicitly: Lu et al. show that training on annotated step-by-step lectures (chains of thought) improves multimodal reasoning [21], but such intermediate annotations are expensive to collect at scale. We follow this implicit route without external process supervision: a stepwise self-teacher exposes its own intermediate signals, and a single-pass student internalizes this procedure, achieving complex, hierarchy-aware reasoning without extra labels or test-time latency.

Knowledge Distillation for Reasoning. In vision–language models, distillation has increasingly been used to transfer multi-step reasoning. Visual Program Distillation (VPD) distills tool-augmented program traces from an LLM–tool system into a single VLM that answers complex visual queries in one forward pass [10]. EKDA distills knowledge from a LLaMA teacher with graph-based alignment to improve knowledge-based VQA [26], and Agent-of-Thought Distillation constructs verified multi-step chains for VideoQA and distills them into a video–language model [28]. While these methods target reasoning, they rely on external teachers, tools, or natural-language chains. Classical self-distillation instead aligns logits or hidden features within a single network [7, 41, 43], but typically ignores dependency-aware cross-level state. In contrast, we distill the hierarchical reasoning process itself: a stepwise self-teacher exposes taxonomy-aligned intermediate signals during training, and a single-pass student internalizes them, yielding ancestor-to-leaf consistency without additional labels, tools, or test-time latency.

6. Conclusion

In this work, we identify a core deficiency of VLMs on hierarchical tasks: errors stem less from missing knowledge than from an inability to maintain coarse-to-fine cross-level state under flat prompting, leading to poor path consistency. We address this with SEKD, a label-free, external-teacher-free procedure in which a VLM first runs in a multi-step conditioned mode to expose intermediate answers and internal states, then distills them into a compact single-pass student. SEKD substantially improves hierarchical understanding while preserving efficiency. Future work could extend this self-internalization principle to more complex or implicit structures and explore iterative distillation or integration with tool-augmented frameworks.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 6, 13, 19

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 — mining discriminative components with random forests. In *European Conference on Computer Vision (ECCV)*, 2014. 2, 5, 6, 12

[3] Walid Bousselham, Hilde Kuehne, and Cordelia Schmid. Vold: Reasoning transfer from LLMs to vision-language models via on-policy distillation. *arXiv preprint arXiv:2510.23497*, 2025. 20

[4] Yihan Cao, Lei Feng, and Bo An. Consistent hierarchical classification with a generalized metric. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*. PMLR, 2024. 1

[5] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024. 6, 13

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 1, 2, 5, 6, 12, 21

[7] Tommaso Furlanello, Zachary C. Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born-again neural networks. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018. 8

[8] Ziyu Guo, Xinyan Chen, Renrui Zhang, Ruichuan An, Yu Qi, Dongzhi Jiang, Xiangtai Li, Manyuan Zhang, Hongsheng Li, and Pheng-Ann Heng. Are video models ready as zero-shot reasoners? an empirical study with the mme-cof benchmark. *arXiv preprint arXiv:2510.26802*, 2025. 19

[9] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2021. 1

[10] Yuke Hu, Saining Xie, Amanpreet Singh, Xiaolong Wang, Aravind Mahendran, et al. Visual program distillation: Distilling tools and programmatic reasoning into vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 8

[11] Drew A. Hudson and Christopher D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 5, 12

[12] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[13] Hyolim Kang, Yunsu Park, Youngbeom Yoo, Yeeun Choi, and Seon Joo Kim. Open-ended hierarchical streaming video understanding with vision language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. 19

[14] 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. *Transactions on Machine Learning Research*, 2025. 3, 6, 13, 19

[15] Liunian Harold Li et al. Symbolic chain-of-thought distillation: Small models can also 'think' step-by-step. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023. 20

[16] Tong Liang and Jim Davis. Making better mistakes in clip-based zero-shot classification with hierarchy-aware language prompts. *arXiv preprint arXiv:2503.02248*, 2025. 12

[17] Weifeng Lin, Xinyu Wei, Ruichuan An, Tianhe Ren, Tingwei Chen, Renrui Zhang, Ziyu Guo, Wentao Zhang, Lei Zhang, and Hongsheng Li. Perceive anything: Recognize, explain, caption, and segment anything in images and videos, 2025. 19

[18] Wei Liu, Junlong Li, Xiwen Zhang, Fan Zhou, Yu Cheng, and Junxian He. Diving into self-evolving training for multimodal reasoning. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*. PMLR, 2025. 20

[19] Yi Liu et al. Where to focus: Investigating hierarchical attention relationship for fine-grained visual classification. In *European Conference on Computer Vision (ECCV)*, 2022. 8

[20] Yiyang Liu et al. Mmbench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision (ECCV)*, 2024. 5, 6, 12

[21] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Aishwarya Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2, 8

[22] Pan Lu et al. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, 2024. 5, 6, 12

[23] Yulin Luo, Shengbang Tong, Bocheng Zou, Ruichuan An, Yang Yang, Chongxuan Li, and Bo An. Llm as dataset analyst: Subpopulation structure discovery with large language model. In *European Conference on Computer Vision (ECCV)*. Springer, 2024. 20

[24] K. T. Noor et al. Taxonomy-guided routing in capsule network for hierarchical image classification. *Knowledge-Based Systems*, 2025. 8, 21

[25] Seulki Park, Yan Zhang, Stella X. Yu, Sara Beery, and Jingyi Huang. Visually consistent hierarchical image classification (h-cast). In *International Conference on Learning Representations (ICLR)*, 2025. 8, 21

[26] Xinsong Qin et al. Efficient knowledge distillation and alignment for improved kb-vqa (ekda). *Scientific Reports*, 2025. 8

[27] Leonardo Ranaldi, Federico Ranaldi, and Giulia Pucci. R2-multiomnia: Leading multilingual multimodal reasoning via self-training. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 8220–8234, 2025. 20

[28] Yifan Shi et al. Enhancing video-llm reasoning via agent-of-thoughts distillation (aotd). In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 8

[29] Wentao Tan, Qiong Cao, Yibing Zhan, Chao Xue, and Changxing Ding. Beyond human data: Aligning multimodal large language models by iterative self-evolution. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2025. 20

[30] Yilun Tan, Yujun Qing, and Bo Gong. Vision llms are bad at hierarchical visual understanding, and llms are the bottleneck. *arXiv preprint arXiv:2505.24840*, 2025. 6, 11, 12, 21

[31] Grant Van Horn, Elijah Cole, Sara Beery, Kimberly Wilber, Serge Belongie, and Oisin Mac Aodha. Benchmarking representation learning for natural world image collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3, 5, 6, 12

[32] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Technical report, California Institute of Technology, 2011. 5, 6, 12

[33] Rui Wang et al. Consistency-aware feature learning for hierarchical fine-grained visual classification. In *Proceedings of the 31st ACM International Conference on Multimedia (ACM MM)*, 2023. 8

[34] Yimu Wang, Mozhgan Nasr Azadani, Sean Sedwards, and Krzysztof Czarnecki. Hawaii: Hierarchical visual knowledge transfer for efficient vision-language models, 2025. NeurIPS 2025. 21

[35] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2, 20

[36] Shi-Long Xu et al. Hyperbolic space with hierarchical margin boosts fine-grained learning from coarse labels. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 8, 21

[37] Kun Yi et al. Exploring hierarchical graph representation for large-scale zero-shot image classification (hgr-net). In *European Conference on Computer Vision (ECCV)*, 2022. 8, 21

[38] Shen Yin, Tao Lei, and Yukun Liu. Toolvqa: A dataset for multi-step reasoning vqa with external tools. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. 2, 8

[39] Andy Zeng, Matthew Attarian, Brian Ichter, Krzysztof Choromanski, Alex Wong, Stefan Welker, Fabio Tombari, Ashwin Purohit, Michael S. Ryoo, Vikas Sindhwani, Jonathan Lee, Vincent Vanhoucke, and Peter Florence. Socratic models: Composing zero-shot multimodal reasoning with language. In *International Conference on Learning Representations (ICLR)*, 2023. 2

[40] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11975–11986, 2023. 12

[41] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 8, 20

[42] Ruohong Zhang, Bowen Zhang, Yanghao Li, Haotian Zhang, Zhiqing Sun, Zhe Gan, Yinfei Yang, Ruoming Pang, and Yiming Yang. Improve vision language model chain-of-thought reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1631–1662, 2025. 20

[43] Zhilu Zhang and Mert R. Sabuncu. Self-distillation as instance-specific label smoothing. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 8

[44] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 6, 13

Self-Empowering VLMs: Achieving Hierarchical Consistency via Self-Elicited Knowledge Distillation

Supplementary Material

In the appendices, we collect all technical details and full experimental results to support reproducibility and deepen understanding of our method. Appendix A formalizes all hierarchical metrics and evaluation protocols; Appendix B summarizes dataset statistics, taxonomy construction, and VQA instance generation; Appendix C details model configuration, training setup, and efficiency; Appendix D presents diagnostic analyses of hierarchical behaviour; Appendix E reports complete in-domain and generalization results across datasets and backbones, together with catastrophic forgetting analyses; Appendix F provides ablations and sensitivity studies of the SEKD loss design; and Appendix G shows qualitative examples and visualisations.

A. Metric Definitions

In this section, we define the path-level metrics, conditional accuracy, and catastrophic forgetting measures used in our experiments.

For each evaluation sample $i \in \{1, \dots, N\}$, let $x^{(i)}$ denote the input image and $y^{(i)}_{1:L_i} = (y^{(i)}_1, \dots, y^{(i)}_{L_i})$ its target taxonomy path, where L_i is the path depth and $y^{(i)}_\ell$ is the ground-truth label at level ℓ. We denote the prediction at level ℓ by $\hat{y}^{(i)}_\ell$; for datasets with fixed depth L we have $L_i = L$ for all i. We use five standard path-level metrics: HCA, LeafAcc, POR, S-POR, and TOR to capture different aspects of hierarchical behavior. Our formulas follow the definitions of Tan et al.[30]

Hierarchical-Consistency Accuracy (HCA). HCA counts how often the entire predicted path from root to leaf matches the ground-truth path exactly; it is the strictest notion of hierarchical correctness we use.

$$\mathrm{HCA} = \frac{1}{N} \sum_{i=1}^{N} \prod_{\ell=1}^{L_i} \mathbb{I}\big(\hat{y}^{(i)}_\ell = y^{(i)}_\ell\big), \tag{7}$$

where $\mathbb{I}(\cdot)$ is the indicator function.

Leaf Accuracy (LeafAcc). LeafAcc focuses on the finest-grained decision at the last level.

$$\mathrm{LeafAcc} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\big(\hat{y}^{(i)}_{L_i} = y^{(i)}_{L_i}\big). \tag{8}$$

Point-Overlap Ratio (POR). POR averages the per-level accuracy along each taxonomy path, so it captures how often individual nodes are correct without requiring the whole path to be right.

$$\mathrm{POR} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{L_i} \sum_{\ell=1}^{L_i} \mathbb{I}\big(\hat{y}^{(i)}_\ell = y^{(i)}_\ell\big). \tag{9}$$

Strict Point-Overlap Ratio (S-POR). S-POR keeps only the longest contiguous block of correct predictions within a path and normalises its length, rewarding models that stay correct over long stretches rather than scattering isolated hits. For the i-th sample we define

$$\text{S-POR} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{L_i} \max_{1 \le a \le b \le L_i} \Big[(b-a+1) \prod_{\ell=a}^{b} \mathbb{I}\big(\hat{y}^{(i)}_\ell = y^{(i)}_\ell\big) \Big]. \tag{10}$$

Top Overlap Ratio (TOR). TOR evaluates local consistency by looking at each parent–child pair and asking whether both endpoints of the edge are predicted correctly.

$$\mathrm{TOR} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{L_i - 1} \sum_{\ell=1}^{L_i - 1} \mathbb{I}\big(\hat{y}^{(i)}_\ell = y^{(i)}_\ell\big) \mathbb{I}\big(\hat{y}^{(i)}_{\ell+1} = y^{(i)}_{\ell+1}\big). \tag{11}$$

High TOR indicates that neighbouring levels tend to agree with the true hierarchy, even if the full path is not always perfect.

Conditional Accuracy. For the depth-wise analysis in Fig. 2 of the main paper, we study how errors propagate along the hierarchy by measuring *conditional accuracy*. For each neighbouring pair $\ell \to \ell+1$, we compute the accuracy at level $\ell+1$ separately for samples where level ℓ is correct and where it is wrong. Restricting to samples with $L_i \ge \ell+1$, we define

$$\mathcal{I}_{\mathrm{corr}(\ell)} = \big\{ i \mid L_i \ge \ell+1,\ \hat{y}^{(i)}_\ell = y^{(i)}_\ell \big\}, \tag{12}$$

$$\mathcal{I}_{\mathrm{err}(\ell)} = \big\{ i \mid L_i \ge \ell+1,\ \hat{y}^{(i)}_\ell \ne y^{(i)}_\ell \big\}. \tag{13}$$

The conditional accuracies are

$$\mathrm{Acc}_{\ell+1|\mathrm{correct}_\ell} = \frac{1}{|\mathcal{I}_{\mathrm{corr}(\ell)}|} \sum_{i \in \mathcal{I}_{\mathrm{corr}(\ell)}} \mathbb{I}\big(\hat{y}^{(i)}_{\ell+1} = y^{(i)}_{\ell+1}\big), \tag{14}$$

$$\mathrm{Acc}_{\ell+1|\mathrm{error}_\ell} = \frac{1}{|\mathcal{I}_{\mathrm{err}(\ell)}|} \sum_{i \in \mathcal{I}_{\mathrm{err}(\ell)}} \mathbb{I}\big(\hat{y}^{(i)}_{\ell+1} = y^{(i)}_{\ell+1}\big), \tag{15}$$

and we summarise the parent–child dependence at depth ℓ by

$$\Delta_\ell = \mathrm{Acc}_{\ell+1|\mathrm{correct}_\ell} - \mathrm{Acc}_{\ell+1|\mathrm{error}_\ell}, \tag{16}$$

so larger Δ_ℓ means that making the parent correct helps the child considerably more.

Table 9. Internal hierarchical datasets used in our experiments. L denotes the number of hierarchical levels.

Dataset	L	#Train	#Val	#Test
iNat-Animal[31]	6	10k	2k	2k
iNat-Plant[31]	6	10k	2k	2k
ImageNet-Animal[6]	11	10k	2k	2k
ImageNet-Artifact[6]	8	10k	2k	2k
CUB-200-2011[32]	4	3476	1159	1159
Food-101[2]	4	-	-	2k

Catastrophic forgetting on external benchmarks. For non-hierarchical external datasets, we compare the accuracy of the Base model before SEKD and the Student model after SEKD. For a dataset d, let $\mathrm{Acc}_{\mathrm{base}}$ and $\mathrm{Acc}_{\mathrm{SEKD}}$ denote the two accuracies, and report

$$\Delta\mathrm{pp} = \mathrm{Acc}_{\mathrm{SEKD}} - \mathrm{Acc}_{\mathrm{base}}, \quad r = \frac{\mathrm{Acc}_{\mathrm{SEKD}}}{\mathrm{Acc}_{\mathrm{base}}}. \tag{17}$$

where $\Delta\mathrm{pp}$ is the absolute change in accuracy(in percentage points) and r is the relative accuracy ratio.

B. Datasets and Taxonomy Construction

B.1. Internal hierarchical datasets

We evaluate SEKD on six internal hierarchical datasets from a recent hierarchical VQA benchmark [30]: iNat-Animal, iNat-Plant, ImageNet-Animal, ImageNet-Artifact, CUB-200-2011, and Food-101. Each dataset is given as a rooted taxonomy, and every image is labeled with a single root-to-leaf path. iNat-Animal and iNat-Plant use a six-level biological hierarchy (kingdom–phylum–class–order–family–genus/species), where the singleton *kingdom* level is ignored in depth-wise metrics. ImageNet-Animal and ImageNet-Artifact use 11- and 8-level WordNet-based trees. CUB-200-2011 adopts a four-level bird hierarchy (order–family–genus–species), and Food-101 uses a four-level tree derived from the semantic groupings of Liang and Davis [16], with leaves matching the 101 original classes and upper levels grouping dishes by cuisine and main ingredients. We use all taxonomies as released and do not modify.

All dataset splits are constructed at the image level, and no image ID appears in more than one split. For each dataset, we randomly partition the images into train/val/test splits with a 6:2:2 ratio, and draw some training, validation, and evaluation samples exclusively from the corresponding split. Table 9 reports, for each dataset, the hierarchical depth L and the number of images in each split.

B.2. Hierarchical question templates and choices

For each internal hierarchical dataset, we convert every image–level pair into a multiple-choice VQA instance. We use a single English template Sec. B.2.1 across datasets and only substitute the dataset-specific label strings. Each question is associated with an image, a level index ℓ, and four answer choices (A–D) Sec. B.2.2 from that taxonomy level, with exactly one correct label.The model is instructed to answer using only option letters.

B.2.1. Prompt templates.

Fig. 4 illustrates both prompt formats. The student and teacher use the same question at each level but differ in how levels are presented and conditioned on. In the joint setting (student), the image and all level-wise questions are concatenated into a single prompt: we state that there are no known facts, ask the model to act as a taxonomist, and list questions Q1–QL, each with four options A–D; the model is instructed to return L capital letters separated by single spaces (e.g., B D A C) with no extra text. In the conditioned-steps setting (teacher), we instead query one level at a time: at level ℓ, the prompt begins with a "Known facts" line summarizing the model's own label choices at previous levels (or "Known facts: None." at the first level), followed by the level-ℓ question and its four options A–D, and the teacher replies with a single letter in A,B,C,D, which we map back to its label name and append to the known-facts block before querying the next level.

B.2.2. Choice ordering and reuse.

For each image and taxonomy level, the four answer choices (A–D) are taken from the same level and consist of the ground-truth label plus three "confusing" labels, following the hierarchical benchmark. Concretely, SigLIP [40] computes cosine similarities between the image and all non-ground-truth labels at that level, and the top three are used as distractors; these four labels (one correct, three distractors) are then randomly permuted and assigned to A–D, so the correct answer has no fixed position. We directly reuse the released four-choice sets, including their randomized ordering, without any re-sampling or reshuffling. For a given image–level pair, the same four choices and their A/B/C/D assignments are used in all teacher and student prompts, so the mapping from option letters to semantic labels is consistent within that instance.

B.3. External benchmarks

For GQA[11], we follow the official evaluation split and annotations, but group all questions associated with the same image into a single VQA call so that the model answers multiple questions in one invocation. For MathVista[22], we use the official testmini split and the released evaluation script without any modification. For MMBench[20], we evaluate on the MMBench-Dev-EN split in the standard single-image, single-question multiple-choice setting, which we mainly use to monitor whether SEKD introduces any catastrophic forgetting on generic multimodal abilities.

(a) Conditioned-step prompt (teacher)

```
You are a taxonomist.
Known facts:  Level 1 = prediction lable1; Level 2 = prediction lable2; ...
Based on taxonomy and the known facts above, where does the
organism in the image fall in terms of <level-ℓ name>?
Question (current level):  Based on taxonomy and the known facts above,
where does the organism in the image fall in terms of <level-name>?
A. option-1 B. option-2 C. option-3 D. option-4
Answer with the option's letter from the given choices directly.
```

(b) Joint prompt (student)

```
Known facts:  None.
You are a taxonomist.  Answer the following multiple-choice
questions about the organism in the image.
Return EXACTLY L capital letters separated by single spaces,
one per question, in order (e.g., "B D A C ...").
For each question, choose ONLY from the letters shown with its options.
Do not output any words or explanations.
Q1 (level 1):  <question text>
A. option-1 B. option-2 C. option-3 D. option-4
Q2 (level 2):  <question text>
A. option-1 B. option-2 C. option-3 D. option-4
...
```

Figure 4. Illustration of the conditioned-step (teacher) and joint (student) prompts used in our hierarchical VQA tasks.

Table 10. Inference latency comparison on InternVL3-8B. Latency is measured per sample on iNat-Animal with average depth $L = 7$ levels.

Model	Invocation mode	Avg. latency (ms/sample)	Speedup
Self-Teacher	$L \approx 7$ passes (multi-step)	1162.35	1.00×
SEKD student	1 pass (single-step)	347.62	3.34×

C. Implementation Details and Training Setup

This section summarizes the implementation details of SEKD: Section C.1 gives the model configuration and training setup (teacher–student initialization, frozen vision encoders, LoRA placement, and optimization), Section C.2 reports training overhead and inference latency, and Section C.3 details the design of the stepwise feature loss.

C.1. Model configuration and training setup

We instantiate SEKD on four open-source VLM backbones: LLaVA-OV-7B[14], InternVL2.5-8B[5], InternVL3-8B[44], and Qwen2.5-VL-7B[1]. For each backbone, the teacher and student are initialized from the same official checkpoint. The teacher is kept fully frozen and only used to generate logits and hidden features under the conditioned-step prompting protocol, while the student is fine-tuned with LoRA. In all experiments the vision encoder is completely frozen; on the language side we freeze all base weights and train only LoRA adapters together with the multimodal projector. For LLaVA-OV-7B, LoRA is applied to the attention projections q_proj, k_proj, v_proj, o_proj and the multimodal projector. For InternVL2.5-8B and InternVL3-8B, LoRA is attached to mlp1.1, mlp1.3, wqkv, and wo. For Qwen2.5-VL-7B, LoRA is applied to model.visual.merger.mlp.0, model.visual.merger.mlp.2, and the attention projections q_proj, k_proj, v_proj, o_proj.

All students share the same optimization setup. We use AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$) with weight decay 0.01, base learning rate 2×10^{-5}, and a cosine schedule with 3% linear warmup, training in FP16 with gradient clipping at 1.0. Unless otherwise noted, all experiments run on a single NVIDIA L20 GPU with batch size 1, gradient accumulation 8, and at most 3 epochs per internal hierarchical dataset; the Qwen2.5-VL-7B vs. Qwen2.5-VL-32B comparison uses an NVIDIA H20 due to model size. We fix the random seed to 42 for all student-training runs. After each epoch we select the checkpoint with the best HCA on the validation split and report results for this model. The distillation loss is a weighted sum of hard-label cross-entropy, soft-label KL, and feature matching with $\lambda_1 = 2.0$, $\lambda_2 = 1.0$, $\lambda_3 = 0.5$, KD temperature 1.0 or 2.0, and L2 feature dis-

Table 11. Diagnostic comparison under matched compute (Dataset: iNat-Animal). All metrics are percentages (%). Parentheses indicate absolute change vs. Joint Invocation.

Model	Protocol	HCA	LeafAcc	TOR	POR	S-POR
LLaVA-OV-7B	Joint Invocation	0.05	25.35	11.92	38.06	20.59
	Independent Levels	4.60 (**+4.55**)	27.45 (**+2.10**)	51.93 (**+40.01**)	65.92 (**+27.86**)	55.43 (**+34.84**)
	Conditioned Steps	8.85 (**+8.80**)	32.45 (**+7.10**)	55.66 (**+43.74**)	67.69 (**+29.63**)	57.71 (**+37.12**)
InternVL2.5-8B	Joint Invocation	1.85	26.35	30.53	51.35	36.85
	Independent Levels	8.70 (**+6.85**)	28.50 (**+2.15**)	57.88 (**+27.34**)	69.82 (**+18.47**)	60.65 (**+23.80**)
	Conditioned Steps	11.15 (**+9.30**)	34.60 (**+8.25**)	57.68 (**+27.14**)	68.94 (**+17.59**)	60.19 (**+23.34**)
InternVL3-8B	Joint Invocation	2.95	30.15	27.64	48.94	33.56
	Independent Levels	11.75 (**+8.80**)	35.80 (**+5.65**)	60.68 (**+33.04**)	72.52 (**+23.58**)	63.20 (**+29.64**)
	Conditioned Steps	16.65 (**+13.70**)	44.05 (**+13.90**)	62.63 (**+34.99**)	73.31 (**+24.37**)	64.26 (**+30.70**)
Qwen2.5-VL-7B	Joint Invocation	9.90	44.15	56.41	71.68	57.61
	Independent Levels	18.90 (**+9.00**)	41.40 (**-2.75**)	68.42 (**+12.01**)	77.98 (**+6.30**)	70.86 (**+13.25**)
	Conditioned Steps	25.80 (**+15.90**)	47.80 (**+3.65**)	70.91 (**+14.50**)	79.09 (**+7.41**)	72.65 (**+15.04**)

tance. LoRA hyperparameters are shared across backbones (rank 64, scaling $\alpha = 16$, dropout 0.05). On the teacher side we use greedy decoding with max_new_tokens=1 and no sampling (temperature=1.0, top_p=1.0), so the teacher signal is deterministic for a given prompt.

C.2. Training and inference cost

Training cost. The additional training overhead of SEKD is modest. On the 10K-sample iNat-Animal subset and a single NVIDIA L20 GPU, the one-time offline generation of teacher pseudo-labels (Sec. 4.2) requires 5.58 GPU-hours. Subsequent student distillation (1-epoch LoRA fine-tuning of InternVL3-8B) takes 1.24 GPU-hours on the same hardware. In total, the full SEKD pipeline on this subset finishes in under 7 GPU-hours, which is small compared to the pre-training cost of the underlying VLMs.

Inference latency. We benchmark wall-clock latency for the 8B multi-step Self-Teacher (L separate forward passes) against our 8B SEKD student (a single pass). All measurements are taken on a single NVIDIA L20 GPU with batch size 1 and the same decoding configuration as in our main experiments. On iNat-Animal, where the average path depth is $L \approx 7$ levels, the Self-Teacher requires 1,162.35 ms per sample, while the SEKD student reduces this to 347.62 ms, yielding a 3.34× speedup (Table 10). This matches the expected L-to-1 ratio between multi-step and single-step inference and shows that amortizing hierarchical reasoning into one joint pass substantially improves latency.

C.3. Anchor tokens and stepwise feature alignment

To complement the stepwise feature distillation loss in Sec. 3.2, we detail here how answer anchors are selected and how student and teacher features are aligned under matched contexts.

Anchor tokens and parsing. For each taxonomy level we treat the option letter token (A/B/C/D) as the decision anchor. On the teacher side, the model is explicitly instructed to answer with a single letter; we take that final letter token as the anchor and use its hidden state as the level-ℓ teacher feature. On the student side, the model is asked to output exactly L capital letters separated by single spaces (for example, "B D A C"). During evaluation we scan the decoded string from left to right, collect the first L capital letters in $\{A, B, C, D\}$, and assign them in order to levels $1, \ldots, L$, ignoring whitespace, punctuation, and any extra text. The hidden state at each of these letter positions is used as the corresponding student anchor for feature matching, and the same letters are also used to compute the hard-label and soft-label losses.

Causal masking and feature alignment. At the moment of generating the level-ℓ answer token, the causal attention mask ensures that the student can only attend to the image features, the prompt tokens up to and including the level-ℓ question, and its own predicted answers for previous levels $a_{1:\ell-1}$; future queries $q_{\ell+1:L}$ and their options remain masked. In the conditioned-steps protocol, the teacher at level ℓ sees the same information set: the image, the current question q_ℓ, and its previously predicted labels summarized as known facts. Both the student feature $h_\ell^{(S)}$ and teacher feature $h_\ell^{(T)}$ are taken at the single answer-token position under matched contexts (image, q_ℓ, and $a_{1:\ell-1}$). This makes the stepwise feature loss well-defined and complementary to distributional imitation: it aligns internal representations at the decision points where each hierarchical level is re-

solved, without leaking information from future levels.

D. Complete Diagnostic Results and Hierarchical Behaviour Analyses

This section presents the complete diagnostic results for all four models on the iNat-Animal dataset, complementing Sec. 2 of the main paper. We report their diagnostic behaviour and show representative failure–repair examples under Joint Invocation and Conditioned Steps.

D.1. Backbone-Level Diagnostic Comparison

As shown in Tab. 11, across all four backbones the three paradigms follow a consistent ordering on strict path-level accuracy: HCA is lowest under Joint Invocation, higher under Independent Levels, and highest under Conditioned Steps. On LLaVA-OV-7B, for example, HCA rises from 0.05% (Joint) to 4.60% (Independent) and 8.85% (Conditioned), and on InternVL3-8B from 2.95% to 11.75% and 16.65%, indicating that per-level separation helps but still falls short of fully conditioned reasoning. Local consistency metrics TOR, POR, and S-POR improve even more: for LLaVA-OV-7B, TOR increases from 11.92% to 51.93% and 55.66%, with both InternVL variants showing similar 30–35 pp gains, suggesting that the main effect is repairing cross-level transitions rather than isolated decisions. Qwen2.5-VL-7B starts from a stronger Joint baseline (HCA 9.90%, TOR 56.41%), so its absolute gains are smaller but follow the same pattern, with Conditioned Steps again yielding the best trade-off (HCA 25.80%, TOR 70.91%, S-POR 72.65%). Overall, these results confirm that the hierarchical failure mode identified in Sec. 2 is robust across architectures, and that explicitly propagating parent decisions effectively restores path-wise coherence in the single-image, multi-question setting.

D.2. Failure–repair case study

Figure 5 visualizes four representative iNat-Animal examples under our three diagnostic paradigms. For each image, we list the ground-truth taxonomy together with the paths produced by Joint Invocation, Independent Levels, and Conditioned Steps, and mark each node with a green check or red cross depending on correctness. Across examples, Joint Invocation often makes an early branching error and then follows an incompatible subtree, Independent Levels repairs many higher-level nodes but can still leave local inconsistencies, and Conditioned Steps yields a coherent path that closely tracks the ground-truth taxonomy. For instance, in the top row (*Ruditapes philippinarum*), Joint Invocation predicts the wrong phylum (*Cnidaria*) and then continues along an incorrect branch, Independent Levels recovers the correct phylum and most intermediate nodes but ends with a wrong species (*Mya arenaria*), whereas Conditioned Steps exactly matches the ground-truth path from *Mollusca* down to *Ruditapes philippinarum*.

E. Complete Main and Generalization Results

This section presents the complete main results underlying the in-domain experiments in Sec. 4.2 and the generalization and forgetting studies in Secs. 4.3 and 4.5 of the main paper. Sec. E.1 reports full in-domain hierarchical results for all internal taxonomies and backbones, Sec. E.2 provides detailed cross-dataset generalization and catastrophic forgetting results on Food-101, MathVista, and MMBench, and Sec. E.3 summarizes an additional efficiency comparison between the 7B SEKD student and the 32B base model.

E.1. Complete in-domain hierarchical results

Table 12 reports the full in-domain hierarchical results for all four VLM backbones on the five datasets. Across all backbones and taxonomies, the SEKD Student consistently improves over the Base Model in HCA, LeafAcc, and all path metrics. For example, on ImageNet-Animal, HCA increases from 0.65% to 30.15% for LLaVA-OV-7B, from 10.20% to 37.35% for InternVL2.5-8B, from 6.50% to 23.95% for InternVL3-8B, and from 33.70% to 49.80% for Qwen2.5-VL-7B. The local consistency metrics TOR and S-POR typically gain tens of percentage points, indicating that SEKD not only raises leaf accuracy but also substantially strengthens path-wise coherence, in line with the trends reported in the main-text Tab. 3.

E.2. Complete Generalization Results and Catastrophic Forgetting

Tables 13, 14, and 15 provide the full numbers for our cross-dataset generalization and catastrophic forgetting analyses.

Cross-dataset generalization. Tab. 13 reports zero-shot transfer to Food-101, averaging students trained on each of the five internal taxonomies. Across all backbones, the SEKD Student strongly outperforms the Base model on every path-level metric; for example, HCA improves from 4.15% to 42.26% for LLaVA-OV-7B, from 8.25% to 35.06% for InternVL2.5-8B, from 7.85% to 24.43% for InternVL3-8B, and from 16.70% to 43.39% for Qwen2.5-VL-7B. LeafAcc, TOR, POR, and S-POR follow the same pattern, with relatively small standard deviations across training taxonomies, indicating that the distilled hierarchical signal transfers robustly to an unseen food taxonomy rather than overfitting to a specific label tree. On MathVista (Tab. 14), student models match or slightly improve overall accuracy and consistently boost arithmetic and geometric reasoning (e.g., InternVL2.5-8B gains $+2.30$ pp on the overall score and around $+2$ pp on both Arithmetic and Geometry), suggesting that SEKD can enhance multi-step visual reasoning beyond hierarchical classification.

Ground Truth	Joint Invocation	Independent Levels	Conditioned Steps
Mollusca	Cnidaria ✗	Mollusca ✓	Mollusca ✓
Bivalvia	Clitellata ✗	Bivalvia ✓	Bivalvia ✓
Venerida	Littorinimorpha ✗	Mytilida ✗	Venerida ✓
Veneridae	Ostreidae ✗	Veneridae ✓	Veneridae ✓
Ruditapes	Littorina ✗	Ruditapes ✓	Ruditapes ✓
Ruditapes	Mya arenaria ✗	Mya Arenaria ✗	Ruditapes
Philippinarum			Philippinarum ✓

Ground Truth	Joint Invocation	Independent Levels	Conditioned Steps
Arthropoda	Mollusca ✗	Arthropoda ✓	Arthropoda ✓
Insecta	Malacostraca ✗	Insecta ✓	Insecta ✓
Lepidoptera	Eulipotyphla ✗	Architaenioglossa ✗	Lepidoptera ✓
Geometridae	Strigopidae ✗	Geometridae ✓	Geometridae ✓
Ematurga	Euclidia ✗	Ematurga ✓	Ematurga ✓
Ematurga Atomari	Pyrausta	Pyrausta	Ematurga
	Despicata ✗	Despicata ✗	Atomari ✓

Ground Truth	Joint Invocation	Independent Levels	Conditioned Steps
Chordata	Mollusca ✗	Chordata ✓	Chordata ✓
Mammalia	Insecta ✗	Mammalia ✓	Mammalia ✓
Lagomorpha	Cuculiformes ✗	Lagomorpha ✓	Lagomorpha ✓
Leporidae	Mephitidae ✗	Leporidae ✓	Leporidae ✓
Lepus	Lepus ✓	Sylvilagus ✗	Lepus ✓
Lepus Californicus	Lepus	Lepus	Lepus
	Americanus ✗	Californicus ✓	Californicus ✓

Ground Truth	Joint Invocation	Independent Levels	Conditioned Steps
Mollusca	Mollusca ✓	Mollusca ✓	Mollusca ✓
Gastropoda	Clitellata ✗	Gastropoda ✓	Gastropoda ✓
Littorinimorpha	Cardiida ✗	Littorinimorpha ✓	Littorinimorpha ✓
Strombidae	Ostreidae ✗	Strombidae ✓	Strombidae ✓
Strombus	Tritia ✗	Strombus ✓	Strombus ✓
Strombus Alatus	Buccinum	Littorina Littorea ✗	Strombus Alatus ✓
	Undatum ✗		

Figure 5. Failure–repair case study across three paradigms on iNat-Animal. Each row shows the ground-truth taxonomy and the paths from Joint Invocation, Independent Levels, and Conditioned Steps; green checks and red crosses mark correct and incorrect nodes.

Catastrophic forgetting. Table 15 focuses on catastrophic forgetting on MMBench. For all four backbones, the accuracy difference between Base and Student is below 0.5 pp, and the relative forgetting ratio r remains under 1% (e.g., $r = 0.51\%$ for LLaVA-OV-7B and $r = 0.10\%$ for Qwen2.5-VL-7B). These small changes indicate that SEKD preserves broad multimodal understanding and does not introduce noticeable catastrophic forgetting on a standard VLM benchmark.

E.3. Complete Results: 7B Student vs. 32B Base

To complement the practicality and efficiency analysis in Sec. 4.5, Tab. 16 reports full path-level metrics for the Qwen2.5-VL-7B SEKD Student compared to the Qwen2.5-VL-32B Base model on internal taxonomies. Across most datasets, the 7B Student matches or closely tracks the 32B Base in HCA and LeafAcc while remaining competitive on TOR, POR, and S-POR, reinforcing that SEKD can transfer hierarchical reasoning ability from a larger model to a much smaller one with limited loss in path-level consistency.

F. Full Ablations and Sensitivity Analyses

This section reports ablation studies that complement the main paper. Section F.1 completes the analysis of the SEKD loss by isolating and recombining the hard-label, soft-logit, and feature-matching terms under a shared evaluation protocol. Section F.2 varies the weights $(\lambda_1, \lambda_2, \lambda_3)$ in the total objective and compares several representative configurations, including the setting used in the main experiments.

F.1. Ablations on loss components

Table 17 shows that Full SEKD brings large gains over the base InternVL3-8B model on iNat-Animal (HCA 2.95% → 17.15%, LeafAcc 30.15% → 41.75%, TOR 27.64% → 64.55%). Using only hard labels or only soft logits already recovers most of this improvement and clearly outperforms

Table 12. **Full Main Results.** Performance comparison across four VLM backbones on five hierarchical datasets. All metrics are percentages (%). The **SEKD Student** (Ours) significantly outperforms the Base Model (Joint Invocation) in single-pass inference across all metrics.

Backbone	Dataset	Method	HCA	Leaf Acc	TOR	POR	S-POR
LLaVA-OV-7B	iNat-Animal	Base Model	0.05	25.35	11.92	38.06	20.59
		Student	**5.40** (+5.35)	**33.80** (+8.45)	**52.83** (+40.91)	**67.39** (+29.33)	**55.01** (+34.42)
	iNat-Plant	Base Model	0.05	26.10	11.80	37.58	20.82
		Student	**4.55** (+4.50)	**34.35** (+8.25)	**45.38** (+33.58)	**61.86** (+24.28)	**49.35** (+28.53)
	CUB-200	Base Model	1.38	24.76	10.76	34.15	19.84
		Student	**10.01** (+8.63)	**37.53** (+12.77)	**30.72** (+19.96)	**55.31** (+21.16)	**41.80** (+21.96)
	ImgNet-Anim.	Base Model	0.65	26.85	17.10	40.95	25.16
		Student	**30.15** (+29.50)	**62.30** (+35.45)	**74.33** (+57.23)	**84.16** (+43.21)	**73.65** (+48.49)
	ImgNet-Arti.	Base Model	0.60	27.15	11.73	39.83	25.86
		Student	**15.35** (+14.75)	**79.20** (+52.05)	**42.40** (+30.67)	**69.45** (+29.62)	**40.35** (+14.49)
InternVL2.5-8B	iNat-Animal	Base Model	1.85	26.35	30.53	51.35	36.85
		Student	**10.30** (+8.45)	**37.15** (+10.80)	**59.09** (+28.56)	**70.63** (+19.28)	**61.83** (+24.98)
	iNat-Plant	Base Model	1.75	27.45	31.63	52.99	35.61
		Student	**7.85** (+6.10)	**34.20** (+6.75)	**49.34** (+17.71)	**63.37** (+10.38)	**52.61** (+16.00)
	CUB-200	Base Model	14.93	40.90	37.27	59.34	47.22
		Student	**17.86** (+2.93)	**44.00** (+3.10)	**42.08** (+4.81)	**63.59** (+4.25)	**50.11** (+2.89)
	ImgNet-Anim.	Base Model	10.20	43.75	39.71	59.57	40.96
		Student	**37.35** (+27.15)	**65.20** (+21.45)	**78.53** (+38.82)	**86.41** (+26.84)	**79.42** (+38.46)
	ImgNet-Arti.	Base Model	2.95	48.60	21.38	50.46	29.47
		Student	**16.60** (+13.65)	**78.40** (+29.80)	**43.85** (+22.47)	**69.86** (+19.40)	**42.68** (+13.21)
InternVL3-8B	iNat-Animal	Base Model	2.95	30.15	27.64	48.94	33.56
		Student	**17.15** (+14.20)	**41.75** (+11.60)	**64.55** (+36.91)	**74.35** (+25.41)	**66.74** (+33.18)
	iNat-Plant	Base Model	2.35	29.00	26.85	48.63	32.36
		Student	**9.10** (+6.75)	**38.60** (+9.60)	**44.76** (+17.91)	**61.39** (+12.76)	**47.84** (+15.48)
	CUB-200	Base Model	12.34	45.73	36.15	58.18	35.38
		Student	**19.24** (+6.90)	**47.97** (+2.24)	**44.55** (+8.40)	**65.06** (+6.88)	**45.71** (+10.33)
	ImgNet-Anim.	Base Model	6.50	32.60	24.60	45.37	32.61
		Student	**23.95** (+17.45)	**54.25** (+21.65)	**59.21** (+34.61)	**72.11** (+26.74)	**62.43** (+29.82)
	ImgNet-Arti.	Base Model	4.35	41.80	20.35	48.38	28.76
		Student	**8.30** (+3.95)	**64.85** (+23.05)	**30.19** (+9.84)	**58.89** (+10.51)	**31.32** (+2.56)
Qwen2.5-VL-7B	iNat-Animal	Base Model	9.90	44.15	56.41	71.68	57.61
		Student	**20.50** (+10.60)	**48.75** (+4.60)	**68.71** (+12.30)	**78.83** (+7.15)	**70.06** (+12.45)
	iNat-Plant	Base Model	4.85	35.30	38.61	58.21	41.61
		Student	**20.30** (+15.45)	**48.95** (+13.65)	**64.83** (+26.22)	**76.39** (+18.18)	**65.04** (+23.43)
	CUB-200	Base Model	15.53	50.39	33.82	61.30	44.91
		Student	**34.34** (+18.81)	**63.42** (+13.03)	**59.59** (+25.77)	**77.48** (+16.18)	**63.44** (+18.53)
	ImgNet-Anim.	Base Model	33.70	69.85	67.43	80.04	66.06
		Student	**49.80** (+16.10)	**78.70** (+8.85)	**83.38** (+15.95)	**90.08** (+10.04)	**83.30** (+17.24)
	ImgNet-Arti.	Base Model	6.45	74.85	30.74	61.09	30.44
		Student	**13.05** (+6.60)	**81.45** (+6.60)	**39.47** (+8.73)	**67.65** (+6.56)	**35.22** (+4.78)

the base model on all hierarchical metrics, whereas using only feature matching remains far below Full SEKD and can even underperform the base model on HCA and path consistency. This indicates that label-level supervision from the teacher (hard or soft) is necessary to anchor meaningful hierarchical decisions; feature alignment alone is not sufficient to learn the taxonomy.

When we combine two losses at a time (i.e., remove one component), performance remains substantially above the base model and much closer to Full SEKD. With either hard or soft label supervision in place, adding feature matching (w/o $\mathcal{L}_{soft}$ and w/o $\mathcal{L}_{feat}$) provides additional gains over the corresponding label-only variants: for example, compared to *only* $\mathcal{L}_{hard}$, adding features in w/o $\mathcal{L}_{soft}$ improves HCA from 14.45% to 15.55%, LeafAcc from 40.90% to 43.85%, and S-POR from 63.24% to 63.68%; relative to *only* $\mathcal{L}_{soft}$, w/o $\mathcal{L}_{feat}$ increases TOR from 61.30% to 62.54% and S-POR from 63.10% to 64.36%. At the same time, removing the hard-label loss still causes the largest degradation among the three. Overall, hard-label distillation provides the primary signal, soft logits refine the teacher's preference distribution, and feature matching acts as an auxiliary signal that is ineffective on its own but reliably improves path-related metrics once strong label super-

Table 13. Internal zero-shot generalization on Food-101 (averaged). "Student" denotes the mean ± std on Food-101 from models trained separately on the 5 internal datasets. We report full path-level metrics; all metrics are percentages (%).

Model	Compare	HCA (%)	Leaf Acc (%)	TOR (%)	POR (%)	S-POR (%)
LLaVA-OV-7B	Base	4.15	24.30	15.53	45.00	35.82
	Student	42.26 (±**3.80**)	84.95 (±**1.92**)	59.75 (±**3.92**)	80.58 (±**2.21**)	65.18 (±**2.91**)
InternVL2.5-8B	Base	8.25	49.40	20.78	54.22	38.22
	Student	35.06 (±**5.29**)	82.60 (±**0.61**)	52.98 (±**5.91**)	77.12 (±**2.49**)	60.99 (±**4.69**)
InternVL3-8B	Base	7.85	27.00	18.50	46.63	38.14
	Student	24.43 (±**7.14**)	56.18 (±**12.11**)	37.39 (±**10.27**)	62.72 (±**7.88**)	51.99 (±**6.94**)
Qwen2.5-7B	Base	16.70	59.85	32.25	62.04	45.82
	Student	43.39 (±**1.59**)	90.09 (±**0.35**)	59.18 (±**1.22**)	81.36 (±**0.61**)	65.29 (±**1.21**)

Table 14. External generalization on MathVista (Accuracy, %).

Model	Compare	Average (%)	Arithmetic Reas. (%)	Geometry Reas. (%)
LLaVA-OV-7B	Base	54.90	50.71	69.87
	Student	55.20 (**+0.30**)	52.97 (**+2.26**)	72.11 (**+2.24**)
InternVL2.5-8B	Base	59.50	58.64	70.71
	Student	61.80 (**+2.30**)	60.62 (**+1.98**)	72.38 (**+1.67**)
InternVL3-8B	Base	64.90	62.04	70.29
	Student	65.10 (**+0.20**)	62.61 (**+0.57**)	71.55 (**+1.26**)
Qwen2.5-7B	Base	64.80	60.06	63.18
	Student	65.20 (**+0.40**)	60.62 (**+0.56**)	66.11 (**+2.93**)

Table 15. Catastrophic forgetting evaluation on MMBench. Δ (pp) and relative forgetting r (%) are reported as decreases (↓).

Model	Acc (Base) %	Acc (Student) %	Δ (pp)	Forgetting r (%)
LLaVA-OV-7B	84.54	84.11	−0.43	0.51
InternVL2.5-8B	86.86	86.43	−0.43	0.49
InternVL3-8B	88.57	88.32	−0.26	0.29
Qwen2.5-7B	86.34	86.25	−0.09	0.10

vision is present.

F.2. Sensitivity to loss weights

Table 18 examines the sensitivity of SEKD to the loss weights $(\lambda_1, \lambda_2, \lambda_3)$ on iNat-Animal with InternVL3-8B. All three settings substantially outperform the base model without SEKD (Table 17), but the hard-dominated configuration $(2.0, 1.0, 0.5)$ used in the main paper is clearly the best: compared to the balanced setting $(1.0, 1.0, 1.0)$, it improves HCA from 13.70% to 17.15%, LeafAcc from 39.20% to 41.75%, TOR from 52.49% to 64.55%, and S-POR from 56.08% to 66.74%. Shifting more weight from hard labels to features with $(0.5, 1.0, 2.0)$ further reduces all hierarchical metrics, for example HCA drops to 11.60% and S-POR to 51.44%.

These trends are consistent with the ablations in Appendix F.1 and explain our choice of $(\lambda_1, \lambda_2, \lambda_3) = (2.0, 1.0, 0.5)$. Appendix F.1 shows that hard-label distillation contributes the most to HCA and path metrics, soft-logit distillation is helpful on its own, and feature matching is the weakest when used alone. The default weighting explicitly encodes this importance order $\mathcal{L}_{hard} \succ \mathcal{L}_{soft} \succ \mathcal{L}_{feat}$. When we equalize the three losses or emphasize the feature term, the hierarchical metrics consistently degrade relative to the default setting, although they still remain above the base model. This confirms that our coefficient choice is not arbitrary but a simple and effective instantiation of the relative importance of the three components.

G. Qualitative examples and visualizations

Figure 6 provides qualitative comparisons between the base model and the SEKD student on four representative cases. Each panel shows one image together with its ground-truth taxonomy or label, followed by the predictions from the

Table 16. Full path-level efficiency comparison between a 7B SEKD student and a 32B base model (Qwen 2.5-VL) on internal taxonomies. All values are percentages (%). Parentheses show the difference of 7B relative to 32B (7B−32B, in pp).

Dataset	Model	HCA	Leaf Acc	TOR	POR	S-POR
iNat-Animal	32B base	22.25	55.55	68.04	79.48	68.00
	7B student	20.50 **(-1.75)**	48.75 **(-6.80)**	68.71 **(+0.67)**	78.83 **(-0.65)**	70.06 **(+2.06)**
iNat-Plant	32B base	19.20	52.00	62.66	76.71	59.00
	7B student	20.30 **(+1.10)**	48.95 **(-3.05)**	64.83 **(+2.17)**	76.39 **(-0.32)**	65.04 **(+6.04)**
ImgNet-Animal	32B base	49.45	78.20	81.08	88.97	76.97
	7B student	49.80 **(+0.35)**	78.70 **(+0.50)**	83.38 **(+2.30)**	90.08 **(+1.11)**	83.30 **(+6.33)**
ImgNet-Artifact	32B base	18.45	83.00	46.49	72.20	43.24
	7B student	13.05 **(-5.40)**	81.45 **(-1.55)**	39.47 **(-7.02)**	67.65 **(-4.55)**	35.22 **(-8.02)**

Table 17. Ablation on loss components. All values are percentages (%). Deltas in parentheses are vs. *Full SEKD*.

Variant	HCA	LeafAcc	TOR	POR	S-POR
Base	2.95	30.15	27.64	48.94	33.56
Full SEKD	17.15	41.75	64.55	74.35	66.74
only $\mathcal{L}_{hard}$	14.45 **(-2.70)**	40.90 **(-0.85)**	60.84 **(-3.71)**	72.04 **(-2.31)**	63.24 **(-3.50)**
only $\mathcal{L}_{soft}$	15.25 **(-1.90)**	43.10 **(+1.35)**	61.30 **(-3.25)**	72.66 **(-1.69)**	63.10 **(-3.64)**
only $\mathcal{L}_{feat}$	0.10 **(-17.05)**	26.50 **(-15.25)**	13.73 **(-50.82)**	39.33 **(-35.02)**	22.66 **(-44.08)**
w/o $\mathcal{L}_{hard}$	14.60 **(-2.55)**	41.30 **(-0.45)**	56.36 **(-8.19)**	68.76 **(-5.59)**	59.01 **(-7.73)**
w/o $\mathcal{L}_{feat}$	15.35 **(-1.80)**	42.65 **(+0.90)**	62.54 **(-2.01)**	73.45 **(-0.90)**	64.36 **(-2.38)**
w/o $\mathcal{L}_{soft}$	15.55 **(-1.60)**	43.85 **(+2.10)**	61.47 **(-3.08)**	72.70 **(-1.65)**	63.68 **(-3.06)**

base model and the student, with correct entries marked by green checks and incorrect ones by red crosses. On Food-101 (*shrimp_and_grits*), the base model produces a sequence of unrelated dishes, whereas the student recovers the correct coarse-to-fine food type and final dish label. On ImageNet-Artifact (*school bus*), the base model drifts along a consumer-goods branch, while the student tracks the correct vehicle hierarchy down to the specific object class. On CUB (*Savannah Sparrow*) and iNat-Plant (*Geranium richardsonii*), the base model makes severe branching errors in the bird and plant taxonomies, respectively, whereas the student restores the full root-to-leaf paths.

H. More Related Work

In this section, we discuss recent advances in large multimodal models and position our work within the broader context of reasoning distillation, self-supervised evolution, and structured representation learning.

Table 18. Sensitivity to loss weights on iNat-Animal with InternVL3-8B. All values are percentages (%). The total loss is $\mathcal{L} = \lambda_1\mathcal{L}_{hard} + \lambda_2\mathcal{L}_{soft} + \lambda_3\mathcal{L}_{feat}$.

$(\lambda_1, \lambda_2, \lambda_3)$	HCA	LeafAcc	TOR	POR	S-POR
(2.0, 1.0, 0.5)	17.15	41.75	64.55	74.35	66.74
(1.0, 1.0, 1.0)	13.70	39.20	52.49	65.64	56.08
(0.5, 1.0, 2.0)	11.60	38.35	47.16	61.89	51.44

H.1. Recent Advances in Large Multimodal Models

The field of multimodal learning has witnessed a paradigm shift with the emergence of Large Multimodal Models (LMMs). Architecture designs such as LLaVA-OneVision [14] and Qwen2.5-VL [1] have established a standard recipe which connects a pre-trained visual encoder with a large language model via a specialized projector. Recent generalist models, such as Perceive Anything [17], have further pushed the boundaries by unifying diverse capabilities—including recognition, explanation, and segmentation—into a single, powerful foundation model. Despite these strides in perceptual recognition and single-turn interactions, current state-of-the-art models often falter when faced with tasks requiring structured multi-step reasoning. This limitation is not confined to static images; recent benchmarks like MME-CoF [8] and frameworks like OpenHOUSE [13] reveal that video-based VLMs also struggle to maintain consistency across hierarchical levels or temporal frames. Our work investigates this specific gap: the inability to organize rich knowledge coherently without explicit structural guidance.

H.2. Distilling Intermediate Reasoning for Efficient Inference

Recent advancements in Large Language Models have demonstrated that the reasoning capabilities of large mod-

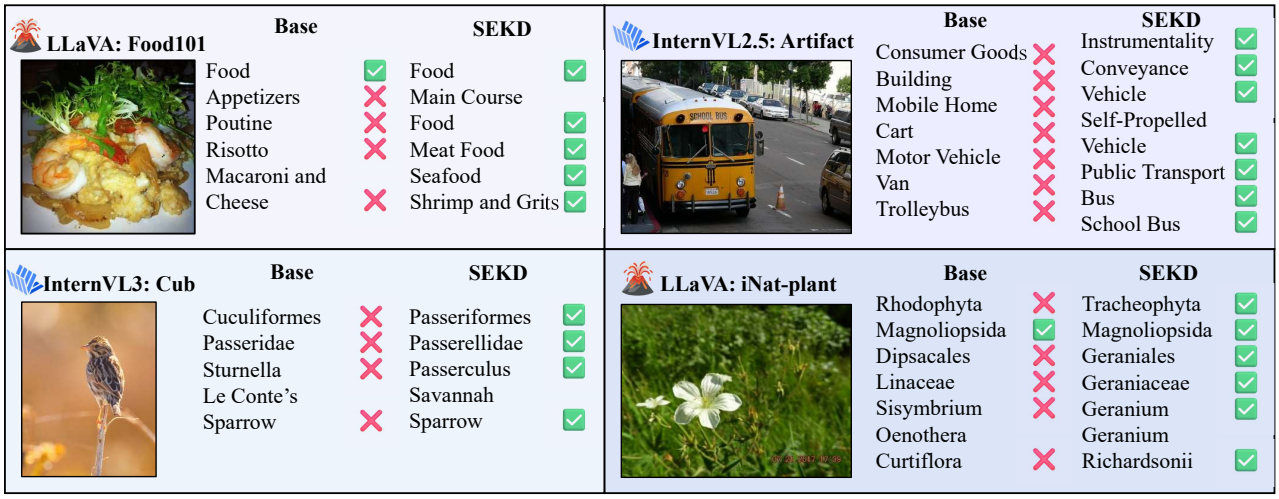

Figure 6. Qualitative comparison between the base model and the SEKD student on Food-101, ImageNet-Artifact, CUB, and iNat-Plant. Each panel shows the image, ground-truth taxonomy or label, and the predictions from the base and student models, with correct entries marked by green checks and incorrect ones by red crosses.

els or multi-step prompting strategies can be transferred to smaller models. This line of research typically employs Chain-of-Thought (CoT) distillation [15, 35] where a student model is trained to generate the rationale steps produced by a teacher. To enhance this, recent works employ various alignment strategies: VOLD [3] transfers reasoning from strong LLMs using reinforcement learning, while Zhang et al. [42] utilize outcome-based rewards to calibrate CoT reasoning. While these approaches successfully align model outputs with human preferences in general domains, traditional knowledge distillation often avoids aligning intermediate features because the teacher and student models typically possess *heterogeneous architectures*. This structural disparity results in mismatched feature dimensions and divergent semantic representations within hidden layers, making direct feature alignment computationally expensive or ineffective. Consequently, prior works on reasoning transfer have largely restricted themselves to output-level imitation. Our approach differs significantly by exploiting the *homogeneous nature* of our self-elicited framework. Since the teacher and student share an identical model backbone and initialization, the barrier of semantic incompatibility is removed. This allows us to directly align the decoder hidden states at critical decision points with high precision. By doing so, SEKD ensures the student internalizes the actual dependency-aware cognitive state of the teacher rather than simply overfitting to the superficial textual output.

H.3. Self-Supervised Evolution and Refinement

There is a growing body of literature focusing on enabling models to improve themselves without human annotation [41]. One prevalent direction involves inference-time strategies such as self-consistency or self-correction. These methods require the model to generate multiple sampling paths or iteratively critique its own previous outputs to arrive at a final answer. While effective, they impose a significant computational burden during deployment as they necessitate repeated forward passes for a single query. Another emerging paradigm is *self-evolving training*, where models iteratively generate synthetic reasoning data to fine-tune themselves. Examples include M-STaR [18] which utilizes reward models, Beyond Human Data [29] which applies iterative DPO, and R2-MultiOmnia [27] which leverages self-training with structured output rewards. Additionally, Luo et al. [23] utilize LLMs to actively discover implicit subpopulation structures to guide learning. These paradigms have demonstrated remarkable success in improving open-ended reasoning by scaling up test-time computation or training data. However, these approaches typically rely on complex data curation pipelines involving uncertainty estimation or external reward models to filter out low-quality generations. Our SEKD framework offers a more direct and efficient alternative to these paradigms. Unlike general self-evolving frameworks that require computationally expensive reward modeling to filter noisy generations, SEKD leverages the strict hierarchical taxonomy as a *natural and high-quality verifier*. This allows us to achieve robust self-improvement in a single distillation round without the need for iterative data curation. We condense the reasoning capability into the model weights for efficient single-pass inference by enforcing a conditioned stepwise process that instantly elicits a stronger teacher state from the base model.

H.4. Structured Representations in Deep Learning

The problem of mapping visual data to hierarchical taxonomies has a long history in computer vision [6]. Traditional approaches often engineered specialized architectures or loss functions to enforce parent-child constraints. Notable examples include the use of hyperbolic embeddings to capture exponential tree structures [36] or the design of dedicated graph neural networks (GNNs) to propagate label dependencies [24, 37]. Recent works like HAWAII [34] continue this trend by modifying the vision backbone with hierarchical experts. These methods explicitly encode the taxonomy into the model structure or the geometric space of the representations. Although effective for capturing specialized hierarchical features, in the era of foundation models, such explicit architectural modifications are less practical as they disrupt the general-purpose nature of pre-trained transformers. Recent studies on Vision-Language Models mostly rely on the implicit knowledge within the pre-trained weights, which we show is insufficient for strict path consistency [25, 30]. SEKD bridges this gap by learning structured representations implicitly. We do not alter the architecture or use specialized embedding layers. We instead shape the general-purpose representations of the VLM to respect taxonomic dependencies through the distillation of a structured reasoning process. This allows the model to retain its open-vocabulary capabilities while adhering to rigid hierarchical constraints when necessary.