TaxaAdapter: Scaling Fine-grained Species Image Generation to the Tree of Life

Mridul Khurana[1†] **Amin Karimi Monsefi**[2] **Justin Lee**[2] **Medha Sawhney**[1]
David Carlyn[2] **Julia Chae**[3] **Jianyang Gu**[2] **Rajiv Ramnath**[2]
Sara Beery[3] **Wei-Lun Chao**[4] **Anuj Karpatne**[5†] **Cheng Zhang**[6†]

[1]Virginia Tech [2]The Ohio State University [3]MIT
[4]Boston University [5]University of Florida [6]Texas A&M University

imageomics.github.io/TaxaAdapter

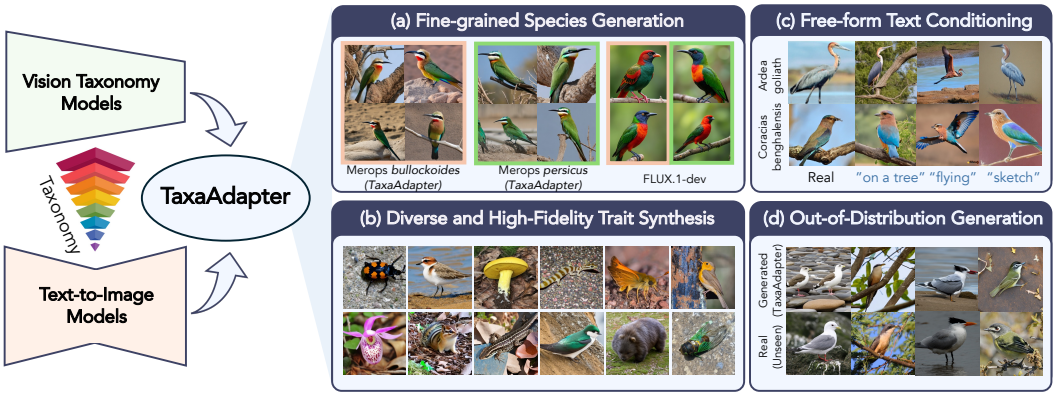


Figure 1: Existing text-to-image models struggle with fine-grained species details. We propose **TaxaAdapter** to inject VTM-derived embeddings into a frozen diffusion model to make fine-grained synthesis taxonomy-aware, enabling: (a) fine-grained species synthesis with accurate morphological details, (b) diverse and high-fidelity trait synthesis, (c) free-form text control, and (d) strong out-of-distribution generalization.

Abstract

Accurately generating images across the Tree of Life is difficult: there are over 10M distinct species on Earth, many of which differ only by subtle visual traits. Despite the remarkable progress in text-to-image synthesis, existing models often fail to capture the fine-grained visual cues that define species identity, even when their outputs appear photorealistic. To this end, we propose TaxaAdapter, a simple and lightweight adapter that incorporates the taxonomic embeddings of Vision Taxonomy Models (VTMs) such as BioCLIP to perform scalable fine-grained species generation over the entire Tree of Life. TaxaAdapter injects VTM embeddings using taxonomy-text dual conditioning into a frozen text-to-image diffusion model, improving species-level fidelity while preserving flexible text control over attributes such as pose, style, and background. Extensive experiments demonstrate that TaxaAdapter consistently improves morphology fidelity and species-identity accuracy over strong baselines. To better evaluate these improvements, we also introduce a multimodal Large Language Model-based metric that summarizes trait-level descriptions from generated and real images, providing a more interpretable measure of morphological consistency. Beyond showing improvements over standard benchmarking experiments, we observe that TaxaAdapter exhibits

[†]{mridul@vt.edu, akarptne@ufl.edu, chzhang@tamu.edu}.

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

strong generalization capabilities in open-world settings, enabling species synthesis in challenging regimes such as few-shot species with only a handful of training images and even species unseen during training. Overall, our results highlight that VTMs are a key ingredient for scalable, fine-grained species generation.

1 Introduction

Generating images of fine-grained species across the Tree of Life is a problem of growing importance in biodiversity science that presents a unique challenge for text-to-image synthesis. Unlike broad object categories, species identity is often defined by subtle morphological cues [41] such as slight differences in beak curvature or plumage patterns. For example, closely related bird species may differ only in faint head or throat markings (Figure 1-(a)). In addition to this challenge, the underlying distribution of biodiversity images is extremely long-tailed: among the more than 10M species on Earth, many have only a handful of curated photos. This motivates new methods for fine-grained generation that enable biologists to generate faithful visualizations of target species even in an open-world setting with little to no data, under varying contexts (e.g., pose) and rendering styles (e.g., schematic sketches) to discover diagnostic traits of closely related taxa.

Recent advances in text-to-image (T2I) diffusion models [35, 31, 3, 18, 2, 51] trained on large-scale web data have reshaped image synthesis by strengthening the alignment between textual descriptions and visual details. However, these models predominantly learn representations organized around coarse semantic categories (e.g., "bird" or "fish") rather than precise taxonomic distinctions (e.g., *Merops bullockoides*). Hence, they often struggle to capture the subtle morphological cues of individual species, producing images that appear realistic but belong to the wrong species (see example images from FLUX.1-dev [18] in Figure 1-(a)) . There is a growing body of work on learning taxonomic embeddings in diffusion models to improve fine-grained generation [25, 15, 30]. Taxonomy hierarchically organizes species into nested groups based on shared traits, relationships, and evolutionary origins, providing a structured prior for species image generation. While promising, these approaches require training these embeddings from scratch, making them computationally expensive to scale and limiting their ability to operate in unseen-species settings, hindering *out-of-distribution* generation performance.

In parallel, there have been several recent efforts for learning taxonomy embeddings over the entire Tree of Life that are aligned with image embeddings for open-world recognition of species. This includes foundational works in building Vision Taxonomy Models (VTMs) such as BioCLIP [40, 6], TaxaBind [38], and BioTrove-CLIP [47]. A unique feature of VTMs is that their taxonomy embeddings are pretrained to be *aligned* with image-based features, however for a different goal of classification rather than image generation. We ask the question: *How can we leverage the image-aligned taxonomic embeddings of VTMs for open-world fine-grained image generation?*

We present **TaxaAdapter**, a lightweight adapter plug-in that injects taxonomy embeddings from VTMs into a frozen text-to-image diffusion model. We employ a simple dual conditioning design using text and taxonomic embeddings, where both these embeddings are sourced from frozen text encoders and taxonomy encoders of VTMs, respectively. This simplicity of our design—without specialized architectural changes or custom loss inventions—is the key strength of our work. By aligning the conditioning space of generative models with image–aligned taxonomy embeddings, TaxaAdapter enables scalable and accurate fine-grained species generation (Figure 1-(b)) over the entire Tree of Life while preserving flexible attribute control through text prompts (Figure 1-(c)).

Extensive experiments show that TaxaAdapter significantly improves species-level fidelity and morphological consistency compared with strong baselines. To better evaluate these improvements, we introduce a novel Multimodal Large Language Model (MLLM)-based trait evaluation framework that summarizes trait-level descriptions from generated and real images, providing an interpretable measure for morphological fidelity. Beyond these improvements on standard benchmark experiments, we show that TaxaAdapter exhibits strong generalization capabilities in out-of-distribution settings, enabling synthesis for rare species with limited training images as well as species never seen during training. To the best of our knowledge, TaxaAdapter is the first framework that bridges diffusion models and pre-trained VTMs, closing the gap between taxonomic knowledge and generative modeling, and offering a practical route for scaling fine-grained generation over the Tree of Life.

2 Related Work

Fine-grained Generation aims to model categories with subtle visual differences, such as closely related fish species, fine facial attributes, or detailed object variations. Prior work explores several directions. Sketch-based control methods guide synthesis through user-provided structure, enabling local detail refinement [27, 56, 20, 51]. Attribute-manipulation approaches condition on latent or semantic attributes to adjust specific features [12, 16, 49, 7, 24, 50]. Personalized generation aims to preserve identity or instance-specific features by tuning part of the model [54, 36, 21, 44, 37]. Recently, TaxaDiffusion [25] and PhyloDiffusion [15] explore conditioning on hierarchical knowledge (e.g., taxonomy) for species image synthesis. However, training taxonomy embeddings in these models from scratch is expensive and limits their scope only to species seen during training. Moreover, retraining the embedding space of diffusion models risks overwriting its native text-conditioning pathway, limiting their flexibility to be guided with text prompts.

Conditional Image Generation extends diffusion models with additional signals such as segmentation masks, sketches, depth maps, or learned adapters [30, 20, 51]. These methods control structure or appearance using auxiliary inputs or small parameter updates, often through cross-attention or feature modulation. However, existing conditioning methods assume that the conditioning input already carries meaningful visual information. Taxonomic text does not meet this requirement: it encodes relationships, not appearance. As a result, zero-shot conditioning on taxonomic names provides limited benefit in fine-grained biological settings. Another approach for fine-grained generation is in-context learning by prompting example images of species [19]. However, this violates the open-world setting where we want to generate images even for novel or unseen species.

Vision Taxonomy Models [40, 6, 38, 47] align images and taxonomy in a shared embedding space. CLIP [32] and its variants learn general-purpose aligned embeddings, while domain-specific models such as BioCLIP [40, 6], TaxaBind [38] & BioTrove-CLIP [47] extend this idea by pairing species images with taxonomic information. These models organize taxa according to visual similarity and have shown promise for classification and retrieval tasks.

3 TaxaAdapter

TaxaAdapter solves the twin goals of fine-grained generation (preserving morphological identity of species) and open-world generation (generalizing over the Tree of Life), making it different from all previous works in this direction. We first provide a motivation for why species names are not sufficient for fine-grained generation and why injecting taxonomic embeddings is non-trivial for the entire Tree of Life. We then introduce our key innovation of taxonomy-text dual conditioning.

3.1 Why Species Names are Not Sufficient for Fine-grained Generation?

Taxonomic names of species alone provide weak visual conditions for text encoders such as CLIP [32] or T5 [33], as they are trained largely for broad semantic alignment rather than fine-grained biological morphology. For example, a prompt such as "a photo of *Merops persicus*" identifies a species taxonomically, but does not explicitly describe the visual traits such as wing patterns, beak shape, or body proportions that uniquely differentiate a species, creating a mismatch between taxonomic names and image features. As a result, even strong T2I models can fail to preserve the morphological identity of species when conditioned solely on species names. Note that this is different from the case where explicit visual attributes are supplied in the text prompt, such as "'a photo-realistic painting of a bird with *green* wings sitting on a tree", which T2I models are trained to handle well. We thus need to move beyond species names to taxonomic embeddings that are aligned with visual information.

3.2 Why is it Non-trivial to Inject Taxonomic Embeddings in Diffusion Models?

A natural way to incorporate taxonomy in T2I models is to fine-tune taxonomic embeddings over a known set of species. However, fine-tuning the backbone (or LoRA) to map thousands of taxonomic names to morphology from scratch, even with progressive training [25], does not scale well in long-tailed datasets where many species have only a few images, and can degrade the model's original text control. The second natural way to leverage taxonomic embeddings from VTMs for open-world generation is to directly swap the text encoder in T2I models with VTM's taxonomic

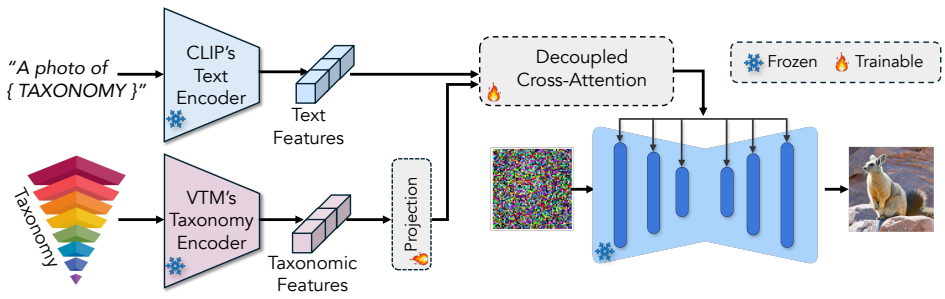


Figure 2: TaxaAdapter pipeline overview. Given a taxonomic name (e.g., `Kingdom` → `Species`), we extract taxonomy-image aligned embeddings using a frozen vision taxonomy model (e.g., BioCLIP [6]) and obtain complementary text features from the frozen CLIP text encoder. The dual conditioning streams are fused through a decoupled cross-attention mechanism, while the diffusion backbone remains frozen.

encoder. However, since diffusion backbones are pretrained for text inputs and not VTM embeddings, there is an interface mismatch that leads to inferior generation quality, as shown empirically in Section 5.4.

3.3 Proposed Idea: Taxonomy-Text Dual Conditioning

TaxaAdapter adopts a dual conditioning mechanism to preserve the ability of T2I models to be controlled using free-form text prompts while being grounded in image-aligned taxonomic embeddings provided by pre-trained VTMs. Specifically, TaxaAdapter employs an adapter-style [48, 26, 20], decoupled conditioning design inspired by prior works on conditional diffusion models [51, 48, 22, 56]. We leave the original text-conditioning pathway untouched and attach a separate, lightweight branch that carries the VTM taxonomy embedding into the U-Net through an additional cross-attention module, as illustrated in Figure 2 (see Section 5.1 and Section A.1 for implementation and training details). The VTM branch is responsible for injecting taxonomy–image aligned information learned from biological pretraining, while the text branch preserves the pretrained model's ability to follow natural-language prompts across arbitrary contexts. A key realization of our work is that instead of introducing a new architectural primitive, we can *repurpose adapter-style training for scaling diffusion models to the entire Tree-of-Life while being morphologically accurate*.

Why is TaxaAdapter Novel? Replacing the image conditioning pathway of adapter-style methods with taxonomic embeddings is not an obvious choice, since it is more natural to treat taxonomy as text. Indeed, previous works in fine-grained generation such as TaxaDiffusion [25] use taxonomic names as inputs in the text encoder of CLIP. However, taxonomic names alone do not provide visual information necessary for fine-grained generation. This is where we leverage the unique feature of VTM taxonomic embeddings that are pretrained to be *aligned with image-based features* as a complementary source of visual information in the dual conditioning mechanism of TaxaAdapter. Conditioning on VTM taxonomic embeddings also enables high-level control of morphological identity across all images of a species rather than low-level spatial control over a particular image instance using sketches or segmentation maps, as explored in prior adapter-style works [48, 26, 20].

What Does TaxaAdapter Enable? Our work, for the first time, enables morphologically accurate fine-grained species generation over the entire Tree of Life, even for species never seen during training. Because VTMs organize taxa according to visual and taxonomic structure, a species with limited training images can still benefit from representations learned from related taxa at the `Genus`, `Family`, or higher taxonomic levels, enabling open-world generation. The first two rows of Fig. 3 show *extremely similar* species that belong to the same genus (*Pipilo*). Their subtle trait variations on wing color patterns (see yellow-highlighted boxes) are accurately captured in images generated by TaxaAdapter, demonstrating strong semantic alignment in the taxonomy–image space. On the other hand, images from T2I models appear photorealistic but are not faithful to species identity. TaxaAdapter also preserves free-form text prompting while maintaining taxonomic identity.

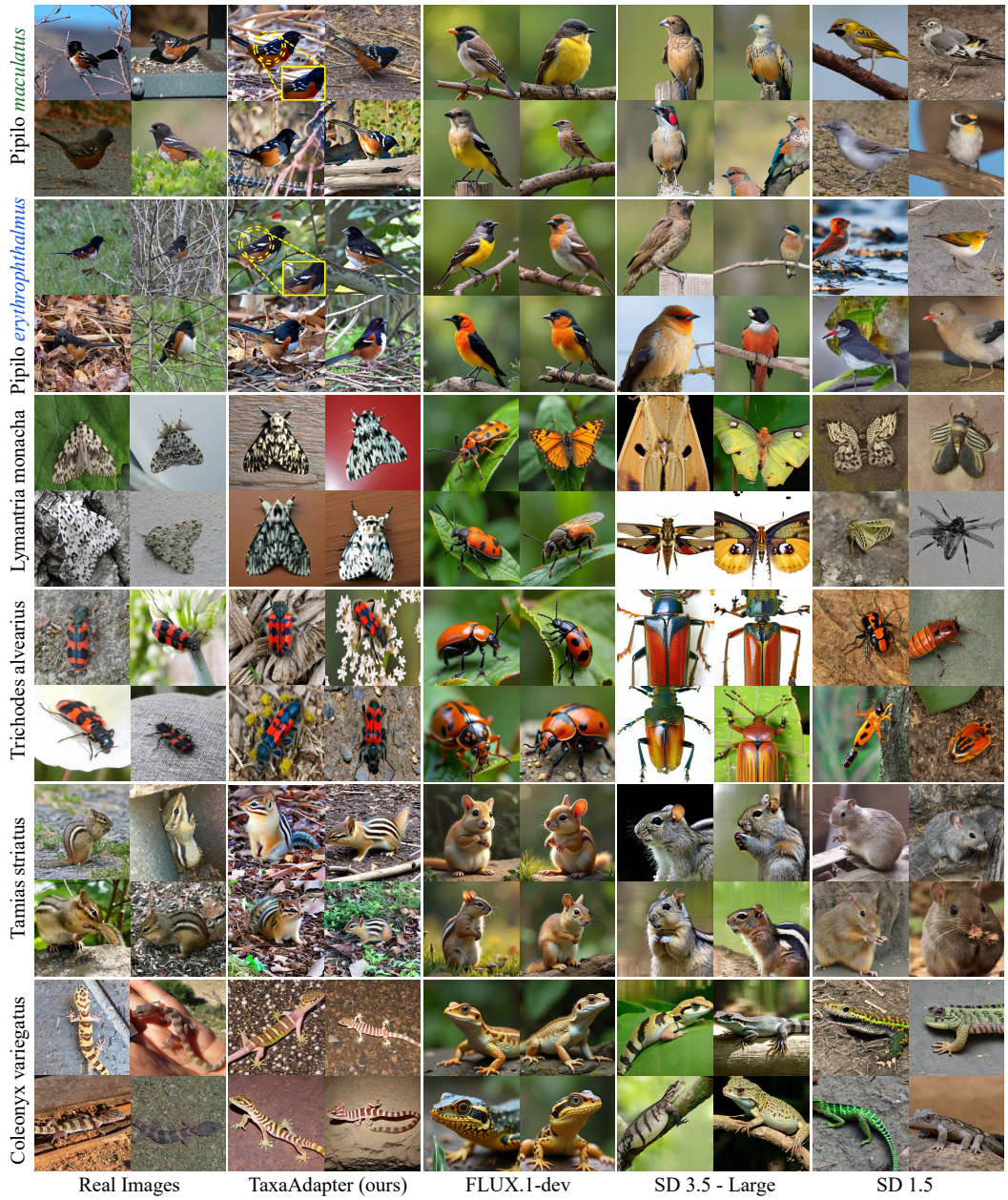


Figure 3: Qualitative comparison on TREEOFLIFE-1M. Each row shows a different species spanning birds, mammals, insects, and reptiles. TaxaAdapter generates morphology-faithful images that align with taxonomy-defining traits (e.g., texture patterns, body shape, coloration) while maintaining realistic textures and backgrounds. Notably, the first two rows illustrate two **extremely similar species** under the same `Genus`, where subtle differences in white spotting patterns on bird wings (highlighted in yellow circles) are correctly captured. Please see Appendix C.5 for additional qualitative results.

4 Proposed Framework for Trait Fidelity Evaluation

Evaluating fine-grained species generation is challenging since standard metrics such as FID [39] and LPIPS [52] emphasize low-level structure (e.g., color and texture) but correlate weakly with species-defining morphological traits. Expert-annotated, trait-level descriptions for species are difficult to obtain at scale. Therefore, we propose a scalable trait-fidelity metric based on trait captions generated by MLLMs. We generated captions of both real and generated images and use the real image captions as a reference. Figure 4 illustrates the pipeline, which consists of three steps:

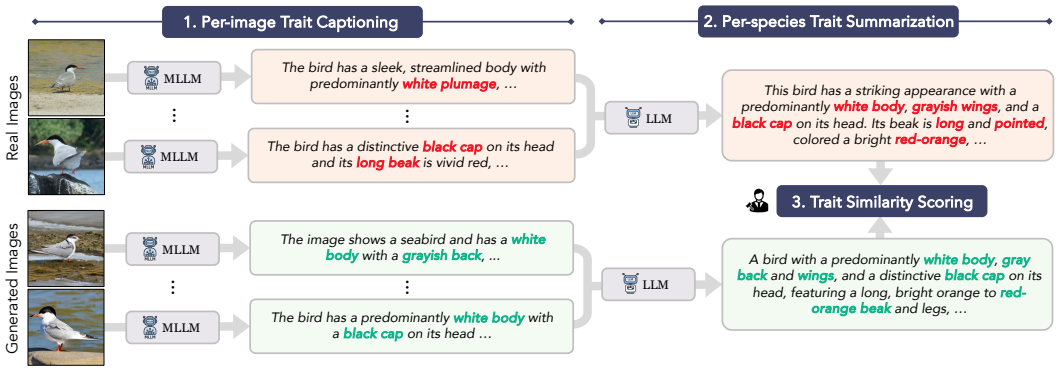


Figure 4: Caption-based trait fidelity evaluation. We leverage an MLLM to generate trait captions for real and generated images, summarize each set into a species-level trait description with an LLM, and compute text similarity between the two summaries. Our metric provides an interpretable, trait-level measure of morphological fidelity that complements standard image metrics.

Step 1: Per-image Trait Captioning. For each species, we prompt an MLLM (namely, InternVL3-8B [57]) to describe only visible traits such as shape, color, and texture, while explicitly excluding species names to avoid bias (see Appendix A.4 for prompt details). In our experiments, we sample 10 images per species for the generated set, resulting in 10 corresponding trait captions. For real images, we caption all the available images of that species.

Step 2: Per-species Trait Summarization. Next, we employ an LLM (namely LLaMa3.1-8B-Instruct [5]) to consolidate the per-image captions into a single "trait caption" for each species. We instruct the model to extract and merge only morphological attributes (e.g., "yellow crown", "short beak", "dark wings") while discarding contextual elements such as background or lighting (see Appendix A.4 for prompt details). The same summarization procedure is applied to real images, producing one reference trait caption per species. Our key insight is that per-species summarization helps recover the full set of visual traits, since distinctive features may be occluded or absent in individual species images due to variations in pose, orientation, or visibility.

Step 3: Trait Similarity Scoring. Finally, we measure the similarity between the summarized captions of real and generated images using several text-based metrics, including BERTScore [53], CLIPScore [9], ROUGE [23], and Mistral-7B text similarity [13], providing a quantitative assessment of how faithfully the generated species preserve authentic visual traits.

Remark. Since expert trait annotations are difficult to obtain at scale, we use real images as references and apply the same captioning and summarization pipeline to both real and generated images, which helps reduce systematic bias and makes the evaluation scalable. We further validate this protocol through a user study (Appendix C.2) and by comparing against BioCAP [55] reference captions, which provide Wikipedia-grounded, trait-rich descriptions (Appendix C.3). Additional implementation details are provided in Appendix A.

5 Experiments

5.1 Experimental Setup

Implementation Details. We repurpose the IP-Adapter [48] training paradigm for taxonomy-text dual conditioning. We replace the image prompt encoder with VTM's taxonomic encoder and pass the full taxonomic hierarchy from `Kingdom` to `Species` to obtain the taxonomic embeddings. These embeddings are projected into the diffusion model's conditioning interface and injected through the decoupled cross-attention as shown in Figure 2, while keeping the VTM encoder, original text encoder, and diffusion backbone frozen. We consider three VTM backbones: BioCLIP [6], BioTrove-CLIP [47], and TaxaBind [38] and Stable Diffusion v1.5 [35] as the frozen diffusion backbone. Additional results with SDXL backbone are provided in Appendix D.3.

Training Details. We train our model for 100 epochs with a batch size of 64 per GPU. We use AdamW with a learning rate of 1×10^{-4}, weight decay of 0.01, and mixed-precision training. At inference, we

Table 1: Results on iNat-mini over 1,103 species (one from each `Family` level in the taxonomy). ZS denotes zero-shot results. We show TaxaAdapter results using two VTM text encoders, BioCLIP and TaxaBind. **Best** is bolded, second-best underlined.

Model	Image Quality		Classification Accuracy		T2I Alignment	Diversity		Caption-based Trait Fidelity			
	FID↓	LPIPS↓	CAS@1↑	CAS@5↑	CLIPScore↑	Prec.↑	Rec.↑	BERT↑	CLIP ↑	ROUGE-L ↑	Mistral-7B ↑
SD 1.5 (ZS) [35]	31.43	0.768	1.24%	2.86%	18.96	0.45	0.32	0.22	0.77	0.20	0.89
SD 3.5-Large (ZS) [3]	70.63	0.806	1.86%	4.48%	19.46	0.38	0.15	0.23	0.78	0.21	0.89
FLUX.1-dev (ZS) [18]	74.14	0.801	1.78%	4.21%	18.94	0.55	0.05	0.29	0.79	0.24	0.91
SD 1.5 (LoRA [11])	15.14	**0.724**	30.69%	56.01%	22.84	0.66	0.51	0.32	0.80	0.25	0.92
SD 3.5-Large (LoRA [11])	31.42	0.762	3.47%	10.18%	21.69	0.62	0.28	0.29	0.79	0.24	0.91
FLUX.1-dev (LoRA [11])	49.72	0.765	1.81%	5.10%	20.06	0.57	0.26	0.26	0.77	0.23	0.90
TaxaAdapter (w/BioCLIP)	13.31	0.742	**61.36%**	**81.99%**	**23.03**	**0.71**	0.52	**0.33**	**0.82**	**0.26**	**0.93**
TaxaAdapter (w/TaxaBind)	**13.04**	0.741	59.98%	79.85%	22.89	**0.71**	**0.53**	0.32	0.81	**0.26**	0.92

Table 2: Results on TREEOFLIFE-1M over 500 species.

Model	Image Quality		Classification Accuracy		T2I Alignment	Diversity		Caption-based Trait Fidelity			
	FID↓	LPIPS↓	CAS@1↑	CAS@5↑	CLIPScore↑	Prec.↑	Rec.↑	BERT↑	CLIP ↑	ROUGE-L ↑	Mistral-7B ↑
SD 1.5 (ZS) [35]	46.89	0.7973	9.32%	25.54%	19.19	0.54	0.33	0.28	0.75	0.24	0.90
SD 3.5-Large (ZS) [3]	79.26	0.8179	11.00%	29.32%	19.43	0.55	0.18	0.28	0.74	0.24	0.90
FLUX.1-dev (ZS) [18]	90.09	0.8003	12.44%	31.14%	19.00	0.66	0.06	0.30	0.75	0.25	0.90
SD 3.5-Large (LoRA [11])	64.21	0.7977	20.46%	50.36%	20.97	0.46	0.28	0.28	0.78	0.24	0.91
FLUX.1-dev (LoRA [11])	59.13	0.7703	13.06%	34.70%	20.01	**0.69**	0.32	0.28	0.77	0.25	0.91
TaxaAdapter (w/BioCLIP)	**29.87**	**0.7345**	**61.98%**	**82.96%**	**21.76**	0.67	0.51	**0.33**	**0.81**	**0.28**	**0.92**
TaxaAdapter (w/TaxaBind)	33.78	0.7476	52.22%	75.24%	20.90	0.59	**0.57**	0.27	0.77	0.25	0.91

generate images at 512×512 resolution using 50 denoising steps. We set the taxonomy-conditioning scale to $\lambda = 1$ for all quantitative experiments and vary it only in the free-form text-control analysis. Additional experiments with varying hyperparameters are provided in Appendix A.

Baselines. We use pre-trained text-to-image models including SD v1.5 [35], SD v3.5-Large [3], and FLUX.1-dev [18] as baselines. For every model, we either apply them zero-shot (ZS) or perform LoRA fine-tuning [11]. We also consider a state-of-the-art taxonomy-aware generative model, TaxaDiffusion [25] as a baseline. See Appendix A for additional details about baselines.

Datasets. We consider three biodiversity datasets for training: (1) iNat-mini [42], with 500K images spanning 10K species from plants, animals, and fungi (covering all seven taxonomic levels), (2) TREEOFLIFE-1M, a subset of TREEOFLIFE-10M dataset [40] with 1M images spanning 200K+ species, and (3) FishNet [14], with 95K images spanning 17.4K fish species. For every image in these datasets, we use their full taxonomic hierarchy string (`Kingdom` $\rightarrow$ `Species`).

Metrics. We evaluate all models across five criteria: **(1) Image Quality:** We use standard metrics of Fréchet Inception Distance (FID) [39] and Learned Perceptual Image Patch Similarity (LPIPS) [52]. **(2) Classification Accuracy:** To evaluate whether generated images preserve species-level identity, we report top-k Classification Accuracy Score (CAS@k) [34] using a DINOv2 [29] linear-probe classifier trained on a separate 500K subset of real images taken from remaining iNaturalist [42]/ TOL-10M [40]. **(3) Text-to-Image Alignment:** We report CLIPScore [9] to measure alignment between generated images and input prompts. **(4) Diversity**: We report Precision and Recall [17] to assess fidelity and coverage. **(5) Caption-based Trait Fidelity:** We use our proposed MLLM-driven trait similarity pipeline from Section 4 to measure morphological trait fidelity in generated images. See Appendix A and B for implementation details of these metrics.

Remark. While stronger backbones such as SD3.5-Large and FLUX.1-dev improve open-domain generation, they still struggle with fine-grained species identity under generic text prompting. This suggests that scaling the generator alone is insufficient for biodiversity synthesis. We use SD1.5 as the primary backbone, as it allows lightweight training of VTM conditioning (22M params) while keeping the T2I model frozen. Applying the same adapter design to large diffusion-transformer backbones, such as FLUX.1-dev, would require substantially more compute (500M params [46]).

5.2 Results

We summarize results across the three benchmarks in Tables 1, 2, and 3, where TaxaAdapter demonstrates consistent gains in image quality, taxonomy-alignment, species identity, and trait

Table 3: Quantitative comparison with TaxaDiffusion [25] on 156 species of iNat-mini and 200 species of FishNet following the evaluation protocol of [25]. TaxaDiffusion numbers directly copied from [25].

Model	iNat-mini		FishNet	
	FID↓	LPIPS↓	FID↓	LPIPS↓
SD 1.5 (ZS) [35]	66.91	0.779	61.93	0.7737
TaxaDiffusion [25]	46.39	0.747	31.87	**0.7319**
TaxaAdapter	**30.36**	**0.741**	**30.68**	0.7818

Table 4: Long-tailed species generation results on ToL-1M. We report results for 5K species with only 1 training image and fewer than 5 training images.

Model	Single image		< 5 images	
	FID↓	LPIPS↓	FID↓	LPIPS↓
SD 1.5 [35]	43.44	0.797	43.07	0.799
SD 3.5-Large [3]	68.66	0.817	71.42	0.819
FLUX.1-dev [18]	89.87	0.821	80.24	0.811
TaxaAdapter	**32.19**	**0.733**	**35.43**	**0.724**

fidelity. While zero-shot T2I models generate visually realistic images (high precision for FLUX.1-dev), but they substantially underperform on species identity (CAS), failing to preserve fine-grained species identity. LoRA fine-tuning yields only marginal CAS improvements. In contrast, TaxaAdapter achieves strong identity preservation (up to ∼83%) while also improving the other metric categories. Table 3 compares TaxaAdapter against TaxaDiffusion on the same subset of iNat-mini and FishNet species as reported in [25]. Figure 3 provides visual comparisons across species drawn from multiple taxonomy groups. Appendix D.5 further shows that providing detailed trait descriptions of species appearance, traits, and morphology from BioCAP [55] that are grounded in Wikipedia as input text to T2I models does improve the baseline generation; however, the models still fall behind TaxaAdapter. Please see Appendix C for additional quantitative and qualitative results.

5.3 Generalization Capabilities of TaxaAdapter

Can TaxaAdapter Generate Rare Species? Biodiversity datasets are intrinsically long-tailed, with many species under-observed due to geography, seasonality, and sampling bias, resulting in very few images for a broad range of taxa. For biological use, a generative model should remain species-faithful in this *rare-but-seen* regime, where only a handful of examples exist to anchor morphology. Table 4 reports results on 5000 TREEOFLIFE-1M species with only a single training image and with fewer than five training images, where we see TaxaAdapter outperform baselines. We do not report CAS here because reliable species classifiers cannot be trained for species with only one or a few images.

Can TaxaAdapter Synthesize Unseen Species? Many species remain under-documented in existing image datasets, and new observations continually arrive from the field. Unlike long-tailed evaluation (rare but seen species), this setting probes whether TaxaAdapter enables accurate generation of species never seen during training. We identify 51 species from the CUB-200 dataset [43] that have no overlap with iNat-mini and use models trained on iNat-mini to synthesize images for these unseen classes. Figure 5 shows that TaxaAdapter generates images that closely resemble the real species despite never seeing them during training. Table 5 also confirms this observation: TaxaAdapter achieves competitive similarity scores compared with zero-shot FLUX.1-dev and SD 3.5, demonstrating that coupling diffusion models with VTMs enables Out-of-Distribution (OOD) species generation. Appendix C.6 reports additional results on the FishNet [14] datasets and separates the species seen during VTM pretraining.

Table 5: Quantitative OOD evaluation on 51 CUB-200-2011 species unseen during training. Models are trained on iNat-mini and tested on unseen CUB classes.

Model	FID ↓	LPIPS ↓	CAS@1 ↑	CAS@5 ↑
SD 1.5 [35]	**36.13**	0.7853	9.61%	25.88%
SD 3.5-Large [3]	44.21	0.8016	8.24%	23.92%
FLUX.1-dev [18]	60.32	0.7901	6.67%	20.39%
TaxaAdapter	42.36	**0.7777**	**38.82%**	**68.24%**

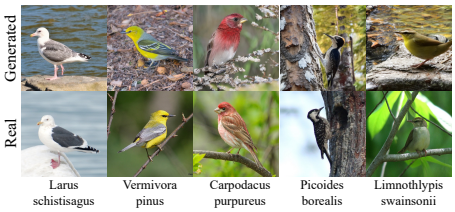
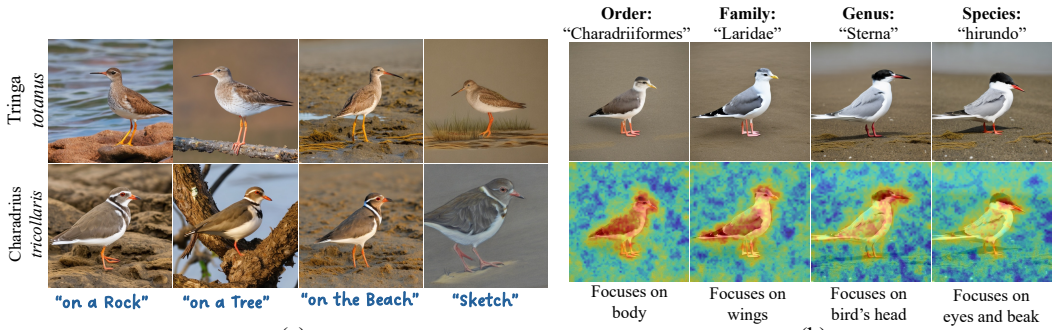


Figure 5: Qualitative results on OOD species generation. TaxaAdapter demonstrates strong zero-shot morphological consistency on unseen classes.

Does Dual Conditioning Preserve Text Control? Biologists often rely on standardized illustrations, such as schematic sketches, that are commonly used to document and study trait variations [4, 45]. Previous methods [25, 30, 15] use taxonomy to fine-tune the diffusion models, which disrupts their



(a) (b)

Figure 6: (a) Free-form text prompting. Our dual-conditioning design allows flexible control over contextual attributes while preserving species-level morphology. (b) Attention maps across different taxonomic tokens. TaxaAdapter gradually focuses on fine-grained information from body structure to head at `Genus` level to eyes and beak at `Species` level.

Table 6: Ablation study on Dual Conditions (DC) and Decoupled Cross-Attention (DCA) on TREEOFLIFE-1M.

	DC	DCA	FID↓	LPIPS↓	CAS@1↑	CAS@5↑
(a)	✗	✓	35.52	0.735	56.42%	78.68%
(b)	✓	✗	35.41	0.740	48.36%	72.86%
(c)	✓	✓	**29.87**	**0.734**	**61.98%**	**82.96%**

Table 7: Analysis of LoRA fine-tuning of SD 1.5, replacing CLIP with BioCLIP. Models are trained on iNat-mini-birds and evaluated on 650 bird species.

	Model	FID ↓	LPIPS ↓
(a)	SD1.5 (ZS)	55.38	0.769
(b)	SD1.5 (CLIP) + LoRA	24.74	0.745
(c)	SD1.5 (BioCLIP) + LoRA	74.60	0.745
(d)	TaxaAdapter	17.08	0.745

ability to follow free-form text prompts. In contrast, TaxaAdapter retains the text pathway of the pretrained diffusion model. As shown in Figure 6-(a) and Appendix C.4, TaxaAdapter generates morphologically correct species across diverse contexts such as environment, pose, and styles.

Can TaxaAdapter Enable Trait Discovery? We visualize cross-attention maps of TaxaAdapter for images generated at each taxonomic level [8] in Figure 6-(b). We observe a coarse-to-fine progression where coarse-taxonomic tokens (e.g., `Order`) emphasize global body structure, (`Family`) emphasizes on wing pattern, mid-level tokens (`Genus`) focus more on the head region, and `Species`-level tokens attend to fine-grained traits such as eyes and beak, generating novel scientific hypotheses about taxonomy-aligned traits that can be subsequently validated in future biological studies.

5.4 Ablation Analyses

Benefits of Dual Conditions. We examine the effects of removing the text branch and conditioning only on VTM's taxonomy features. As shown in Table 6 row (a), this single-stream variant underperforms the dual-conditioning design of row (c), suggesting complementary contextual information from the text pathway that boosts generation performance.

Effect of Decoupled Cross-Attention. We evaluate the role of decoupled cross-attention by replacing it with a single cross-attention layer that averages the text and taxonomy embeddings. Table 6 row (b) shows a clear performance drop, suggesting that a unified attention stream cannot disentangle morphology from contextual cues.

Replacing CLIP with BioCLIP As highlighted in Section 3.3, we show that naïve fine-tuning with Taxonomy (Table 7-(b)) and the replace SD 1.5 CLIP's text encoder with a BioCLIP (Table 7-(c)) is non-trivial as diffusion backbones are heavily pretrained for text inputs and not VTM embeddings and this interface mismatch that leads to inferior generation quality. See Appendix D.2

Why do we need Domain-Specific Encoders? We replace the VTM encoder in TaxaAdapter with a generic CLIP text encoder (while retaining the dual-conditioning design) and observe a performance drop in Table 15 row (a). This highlights that taxonomy–image aligned embeddings are critical for fine-grained species generation, and fine-tuning alone is not sufficient. Table 15 also shows compatibility with Different VTMs.

6 Conclusion and Future Work

We introduce TaxaAdapter, a lightweight plug-in that injects VTM-derived taxonomy embeddings into frozen diffusion backbones, enabling scalable fine-grained species synthesis while preserving morphology fidelity, species identity, and full controllability through free-form text prompts. Future directions include extending TaxaAdapter to diffusion-transformer architectures and leveraging our MLLM-driven caption framework to interpret the semantic traits and distinctions across species and higher taxonomic groups. TaxaAdapter offers a scalable path toward taxonomy-aware generative modeling over the Tree of Life that continually improves as generative models evolve. We discuss the **Limitations and Failure Cases** of TaxaAdapter in Appendix E.

Acknowledgments and Disclosure of Funding

Our research is supported in part by NSF OAC 2118240, NSF Award 2330423, NSERC Award 585136, the MIT Generative AI Consortium, and the NVIDIA Academic Grant Program. The authors acknowledge the Ohio Supercomputer Center and Advanced Research Computing at Virginia Tech for providing computational resources.

References

[1] Krzysztof Adamkiewicz, Brian B Moser, Stanislav Frolov, Tobias Christian Nauen, Federico Raue, and Andreas Dengel. When pretty isn't useful: Investigating why modern text-to-image models fail as reliable training data generators. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 36660–36669, 2026.

[2] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023.

[3] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

[4] Ray Fisher and Ashley Dowling. Modern methods and technology for doing classical taxonomy. *Acarologia*, 50:395–409, 10 2010. doi: 10.1051/acarologia/20101981.

[5] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[6] Jianyang Gu, Samuel Stevens, Elizabeth G Campolongo, Matthew J Thompson, Net Zhang, Jiaman Wu, Andrei Kopanev, Zheda Mai, Alexander E White, James Balhoff, et al. Bioclip 2: Emergent properties from scaling hierarchical contrastive learning. *arXiv preprint arXiv:2505.23883*, 2025.

[7] Inhwa Han, Serin Yang, Taesung Kwon, and Jong Chul Ye. Highly personalized text embedding for image manipulation by stable diffusion. *arXiv preprint arXiv:2303.08767*, 2023.

[8] Alec Helbling, Tuna Han Salih Meral, Ben Hoover, Pinar Yanardag, and Duen Horng Chau. Conceptattention: Diffusion transformers learn highly interpretable features. *arXiv preprint arXiv:2502.04320*, 2025.

[9] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 7514–7528, 2021.

[10] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[11] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

[12] Yi Huang, Jiancheng Huang, Yifan Liu, Mingfu Yan, Jiaxi Lv, Jianzhuang Liu, Wei Xiong, He Zhang, Shifeng Chen, and Liangliang Cao. Diffusion model-based image editing: A survey. *arXiv preprint arXiv:2402.17525*, 2024.

[13] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

[14] Faizan Farooq Khan, Xiang Li, Andrew J Temple, and Mohamed Elhoseiny. Fishnet: A large-scale dataset and benchmark for fish recognition, detection, and functional trait prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 20496–20506, 2023.

[15] Mridul Khurana, Arka Daw, M Maruf, Josef C Uyeda, Wasila Dahdul, Caleb Charpentier, Yasin Bakış, Henry L Bart Jr, Paula M Mabee, Hilmar Lapp, et al. Hierarchical conditioning of diffusion models using tree-of-life for studying species evolution. In *European Conference on Computer Vision*, pages 137–153. Springer, 2024.

[16] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2426–2435, 2022.

[17] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in neural information processing systems*, 32, 2019.

[18] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

[19] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

[20] Ming Li, Taojiannan Yang, Huafeng Kuang, Jie Wu, Zhaoning Wang, Xuefeng Xiao, and Chen Chen. Controlnet++: Improving conditional controls with efficient consistency feedback. In *European Conference on Computer Vision*, pages 129–147. Springer, 2025.

[21] Xiaoming Li, Xinyu Hou, and Chen Change Loy. When stylegan meets stable diffusion: a w+ adapter for personalized image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2187–2196, 2024.

[22] Zhen Li, Mingdeng Cao, Xintao Wang, Zhongang Qi, Ming-Ming Cheng, and Ying Shan. Photomaker: Customizing realistic human photos via stacked id embedding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8640–8650, 2024.

[23] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013/.

[24] Naoki Matsunaga, Masato Ishii, Akio Hayakawa, Kenji Suzuki, and Takuya Narihira. Fine-grained image editing by pixel-wise guidance using diffusion models. *arXiv preprint arXiv:2212.02024*, 2022.

[25] Amin Karimi Monsefi, Mridul Khurana, Rajiv Ramnath, Anuj Karpatne, Wei-Lun Chao, and Cheng Zhang. Taxadiffusion: Progressively trained diffusion model for fine-grained species generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8579–8589. IEEE, 2025.

[26] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 4296–4304, 2024.

[27] Pouyan Navard, Amin Karimi Monsefi, Mengxi Zhou, Wei-Lun Chao, Alper Yilmaz, and Rajiv Ramnath. Knobgen: controlling the sophistication of artwork in sketch-based diffusion models. *arXiv preprint arXiv:2410.01595*, 2024.

[28] A. A. Nemudryi, K. R. Valetdinova, S. P. Medvedev, and S. M. Zakian. Talen and crispr/cas genome editing systems: Tools of discovery. In *Acta naturae, 6(3)*, page 19–40, 2014.

[29] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[30] Ziying Pan, Kun Wang, Gang Li, Feihong He, and Yongxuan Lai. Finediffusion: Scaling up diffusion models for fine-grained image generation with 10,000 classes. *arXiv preprint arXiv:2402.18331*, 2024.

[31] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[33] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[34] Suman Ravuri and Oriol Vinyals. Classification accuracy score for conditional generative models. *Advances in neural information processing systems*, 32, 2019.

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[36] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023.

[37] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Wei Wei, Tingbo Hou, Yael Pritch, Neal Wadhwa, Michael Rubinstein, and Kfir Aberman. Hyperdreambooth: Hypernetworks for fast personalization of text-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6527–6536, 2024.

[38] Srikumar Sastry, Subash Khanal, Aayush Dhakal, Adeel Ahmad, and Nathan Jacobs. Taxabind: A unified embedding space for ecological applications. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1765–1774. IEEE, 2025.

[39] Maximilian Seitzer. pytorch-fid: FID Score for PyTorch. https://github.com/mseitzer/pytorch-fid, August 2020. Version 0.3.0.

[40] Samuel Stevens, Jiaman Wu, Matthew J Thompson, Elizabeth G Campolongo, Chan Hee Song, David Edward Carlyn, Li Dong, Wasila M Dahdul, Charles Stewart, Tanya Berger-Wolf, et al. Bioclip: A vision foundation model for the tree of life. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19412–19424, 2024.

[41] Grant Van Horn and Pietro Perona. The devil is in the tails: Fine-grained classification in the wild. *arXiv preprint arXiv:1709.01450*, 2017.

[42] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018.

[43] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. Caltech-ucsd birds-200-2011 (cub-200-2011). Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[44] Fu-Yun Wang, Zhaoyang Huang, Xiaoyu Shi, Weikang Bian, Guanglu Song, Yu Liu, and Hongsheng Li. Animatelcm: Accelerating the animation of personalized diffusion models and adapters with decoupled consistency learning. *arXiv preprint arXiv:2402.00769*, 2024.

[45] Barbara Wittmann. Outlining species: Drawing as a research technique in contemporary biology. *Science in Context*, 26(2):363–391, 2013. doi: 10.1017/S0269889713000094.

[46] XLabs AI. Flux ip-adapter. https://huggingface.co/XLabs-AI/flux-ip-adapter, 2024. Hugging Face model repository.

[47] Chih-Hsuan Yang, Benjamin Feuer, Talukder Jubery, Zi Deng, Andre Nakkab, Md Zahid Hasan, Shivani Chiranjeevi, Kelly Marshall, Nirmal Baishnab, Asheesh Singh, et al. Biotrove: A large curated image dataset enabling ai for biodiversity. *Advances in Neural Information Processing Systems*, 37:102101–102120, 2024.

[48] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.

[49] Dongxu Yue, Qin Guo, Munan Ning, Jiaxi Cui, Yuesheng Zhu, and Li Yuan. Chatface: Chat-guided real face editing via diffusion latent space manipulation. *arXiv preprint arXiv:2305.14742*, 2023.

[50] Cheng Zhang, Xuanbai Chen, Siqi Chai, Chen Henry Wu, Dmitry Lagun, Thabo Beeler, and Fernando De la Torre. Iti-gen: Inclusive text-to-image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3969–3980, 2023.

[51] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023.

[52] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[53] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

[54] Xulu Zhang, Xiao-Yong Wei, Wengyu Zhang, Jinlin Wu, Zhaoxiang Zhang, Zhen Lei, and Qing Li. A survey on personalized content synthesis with diffusion models. *arXiv preprint arXiv:2405.05538*, 2024.

[55] Ziheng Zhang, Xinyue Ma, Arpita Chowdhury, Elizabeth Campolongo, Matthew Thompson, Net Zhang, Samuel Stevens, Hilmar Lapp, Tanya Berger-Wolf, Yu Su, et al. Biocap: exploiting synthetic captions beyond labels in biological foundation models. In *International Conference on Learning Representations*, volume 2026, pages 125415–125451, 2026.

[56] Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 36:11127–11150, 2023.

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

Appendix

A Additional Implementation Details **14**

A.1 Background and Additional Method Details 14

A.2 Backbone and Conditioning Modules 15

A.3 Training and Hyperparameter Details 16

A.4 Trait Fidelity Evaluation 16

B Revisiting Main Results **17**

C Additional Results **18**

C.1 Trait Fidelity Evaluation 18

C.2 User Study for Trait Fidelity Evaluation. 18

C.3 Trait Fidelity Evaluation with BioCAP captions as biologically informed external references. 19

C.4 Free-form Text Conditioning 19

C.5 Additional Qualitative Results 19

C.6 Additional OOD results 20

C.7 Mixed or Hybrid Species 21

D Additional Ablations and Analyses **22**

D.1 Compatibility with Different VTMs. 22

D.2 Replacing CLIP with BioCLIP 22

D.3 Compatibility with Other Diffusion Backbones 23

D.4 Taxonomic Name vs. Common Name 23

D.5 T2I baseline using detailed morphological descriptions/ retrieval-augmented prompts 24

D.6 Isolating the contribution of hierarchical taxonomy 24

E Limitations **24**

F Ethics and Social Impact **25**

A Additional Implementation Details

A.1 Background and Additional Method Details

Text-to-Image Diffusion Models. Diffusion models generate samples by progressively denoising a noisy latent variable $\mathbf{z}_t$ at timestep t. In conditional generation, the denoising model is additionally conditioned on an auxiliary signal $\mathbf{c}$, such as a class label, text prompt, or image embedding, to guide the generation process [35]. At each timestep, a denoising network predicts the noise added to the clean latent $\mathbf{z}_0$. The standard training objective is

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathbf{z}_0,\boldsymbol{\epsilon},\mathbf{c},t}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, \mathbf{c}, t)\|_2^2\right], \tag{1}$$

where $\boldsymbol{\epsilon}$ is sampled Gaussian noise and $\boldsymbol{\epsilon}_\theta$ is the denoising network. In text-to-image diffusion models, the conditioning signal $\mathbf{c}$ is typically produced by a pretrained text encoder, such as CLIP [32], and injected into the denoising U-Net through cross-attention layers.

Vision Taxonomy Models. Vision Taxonomy Models (VTMs) are trained to align visual observations with taxonomic information, producing embeddings that capture biological and taxonomic structure. Given a taxonomic hierarchy string y_{taxa} from `Kingdom` to `Species`, for example "*Animalia, Chordata, Aves, Passeriformes, Passerellidae, Spizella, breweri*", a VTM taxonomy encoder $\mathcal{E}_{\text{VTM}}$ produces a taxonomy-aware embedding:

$$\mathbf{c}_{\text{taxa}} = \mathcal{E}_{\text{VTM}}(y_{\text{taxa}}). \tag{2}$$

Taxonomy-Text Dual Conditioning. To combine species-level taxonomic information with the flexible controllability of natural-language prompts, we use a taxonomy-text dual-conditioning design. The model receives two complementary conditioning signals:

$$\mathbf{c}_{\text{taxa}} = \mathcal{E}_{\text{VTM}}(y_{\text{taxa}}), \qquad \mathbf{c}_{\text{text}} = \mathcal{E}_{\text{CLIP}}(y_{\text{text}}), \tag{3}$$

where y_{text} is a natural-language prompt such as "*a photo of a bird*". In practice, we incorporate taxonomy and adopt the template '*a photo of* y_{taxa}" to maintain semantic consistency with the introduced embeddings.

Decoupled Cross-Attention. We repurpose the decoupled cross-attention mechanism from IP-adapter [48] that assigns a separate attention to each conditioning source. Specifically, the taxonomy embedding $\mathbf{c}_{\text{taxa}}$ is first passed through a projection network consisting of a linear layer and LayerNorm. Let $\boldsymbol{Q}$ be the query from the U-Net features, and $(\boldsymbol{K}_{\text{text}}, \boldsymbol{V}_{\text{text}})$ and $(\boldsymbol{K}_{\text{taxa}}, \boldsymbol{V}_{\text{taxa}})$ be key-value pairs from the taxonomy and text branches, respectively. The dual-attention update is written as follows:

$$\boldsymbol{Z} = \text{Att}(\boldsymbol{Q}, \boldsymbol{K}_{\text{text}}, \boldsymbol{V}_{\text{text}}) + \text{Att}(\boldsymbol{Q}, \boldsymbol{K}_{\text{taxa}}, \boldsymbol{V}_{\text{taxa}}) \tag{4}$$

Our use of decoupled cross-attention allows the taxonomy branch to focus on species-level morphology, while the text branch controls contextual cues such as style and background, enabling complementary conditioning.

Parameter-Efficient Training. During training, we update only the projection network and the decoupled cross-attention layers, keeping all parameters of the pretrained diffusion model frozen. The model is trained on large-scale taxonomy–image pairs using a similar denoising objective as in Equation 1:

$$\mathbb{E}_{\mathbf{z}_0, \boldsymbol{\epsilon}, \mathbf{c}_{\text{text}}, \mathbf{c}_{\text{taxa}}, t}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, \mathbf{c}_{\text{text}}, \mathbf{c}_{\text{taxa}}, t)\|_2^2\right], \tag{5}$$

To enable Classifier-Free Guidance [10] at inference, we randomly drop the dual-conditioning branch during training. The parameter-efficient strategy minimizes computational cost and prevents forgetting of the pretrained text–image alignment, allowing the model to retain its ability to be controlled by text prompts while incorporating biological knowledge.

Inference. At inference time, both conditioning streams operate jointly to guide the denoising process. Since the taxonomy and text cross-attention modules are decoupled, their contributions can be independently controlled by a weighting factor λ:

$$\boldsymbol{Z} = \text{Att}(\boldsymbol{Q}, \boldsymbol{K}_{\text{text}}, \boldsymbol{V}_{\text{text}}) + \lambda \cdot \text{Att}(\boldsymbol{Q}, \boldsymbol{K}_{\text{taxa}}, \boldsymbol{V}_{\text{taxa}}) \tag{6}$$

where λ modulates taxonomy conditioning strength. Setting $\lambda = 0$ recovers the original text-to-image diffusion model, while a larger λ emphasizes taxonomy-specific traits derived from VTMs. This provides intuitive control between visual appearance and biological precision for fine-grained species generation.

Free-form Text Prompting. Thanks to the taxonomy-text dual conditioning design, we observe that the text branch remains fully functional and can be freely customized to specify background, pose, lighting, or artistic style. For example, given a taxonomy sequence y_{taxa}, users can prompt "y_{taxa} *on a tree branch*", "y_{taxa} *under cloudy sky*" or "y_{taxa} *as a schematic sketch*" to produce morphologically accurate yet stylistically diverse outputs. This flexibility makes TaxaAdapter suitable for both scientific visualization, while the taxonomy branch ensures trait fidelity.

A.2 Backbone and Conditioning Modules

Diffusion Backbone. All experiments were conducted using a publicly available Stable Diffusion v1.5 [35] latent diffusion model backbone, and all the weights are frozen throughout training. Images

Table 8: Training and inference hyperparameter for TaxaAdapter. These settings are used across all datasets unless otherwise noted.

Parameter	Value
Base model	Stable Diffusion v1.5 (frozen)
VTM encoder	BioCLIP / TaxaBind / BioTrove-CLIP (frozen)
Image resolution	512×512
Latent shape	$64 \times 64 \times 4$
Trainable parameters	$\approx$ 22M (proj + attn)
Epochs	100
Batch size per GPU	64
Optimizer	AdamW
Learning rate	1×10^{-4}
Weight decay	0.01
Precision	`fp16`
Classifier-free drop prob.	0.05 (dual conditioning)

are resized and center-cropped to a resolution of 512×512, and encoded into a latent tensor of shape $64 \times 64 \times 4$. We also evaluate using Stable Diffusion XL [31] and present its results in Section D.3.

Taxonomy Encoder. Unless specified, the taxonomy encoder $\mathcal{E}_{\text{VTM}}$ is instantiated from BioCLIP 2 [6]. We also evaluate our framework using TaxaBind [38] and BioTrove-CLIP [47] as alternative Vision Taxonomy Models (VTMs). In all cases, the VTMs' taxonomy encoders are kept frozen. We feed the full 7-level hierarchical Linnaean taxonomy paths defined as `Kingdom`, `Phylum`, `Class`, `Order`, `Family`, `Genus` and `Species` as a string (e.g., "*Animalia, Chordata, Aves, Passeriformes, Passerellidae, Spizella, breweri*") into the taxonomy encoder $\mathcal{E}_{\text{VTM}}$ of the VTM.

A.3 Training and Hyperparameter Details

Model Parameters. Our parameter-efficient training strategy (Section A.1) only updates the following: (i) the projection network applied to $\mathbf{c}_{\text{taxa}}$ (Eq. 3), and (ii) the additional weights introduced by the taxonomy-decoupled cross-attention. All other parameters of the diffusion backbone and VTMs remain frozen. In total, this adds approximately 22M trainable parameters, which is a small fraction of the 895M ($\sim$ 2.45%) parameters of the full Stable Diffusion model. This keeps memory and compute overhead modest while enabling strong taxonomy-aware control.

Training Setup. Table 8 summarizes the main hyperparameters used for TaxaAdapter. All models are trained for 100 epochs with a batch size of 64 images per GPU. We use 4 NVIDIA H100 GPUs with mixed-precision (`fp16`) for training the models. Optimization is performed with AdamW, using a learning rate of 1×10^{-4} and weight decay of 0.01 for all adapter parameters. For classifier-free guidance, we follow the standard ϵ-prediction objective in Eq. 1 and randomly drop the conditioning during training with probability 0.05 and replace both $\mathbf{c}_{\text{text}}$ and $\mathbf{c}_{\text{taxa}}$ with null tokens (i.e., an unconditional branch). All training images are resized to 512×512 and taxonomic labels are taken from the dataset metadata.

Compute TaxaAdapter takes $\sim$1 day to train on the FishNet dataset, whereas full fine-tuning of SD1.5 requires updating $\sim$860M parameters, which takes $\sim$6 days. For LoRA baselines, we train FLUX.1-dev and SD3.5-Large with LoRA rank 16 for 50k iterations with the same batch size of 64 per GPU.

Inference Setup During inference, we use the dual-conditioning formulation from Eq. 6. We fix $\lambda = 1$ for all quantitative results. This setting provides a good balance between preserving species-level morphology (taxonomy branch). For free-form text prompting, we use $\lambda = 0.5$ to allow flexible control as justified in Section 5.3 of the main paper. We use a standard deterministic sampler with 50 denoising steps for all models, including baselines and a classifier-free guidance scale of 3.5. We run all our experiments with a fixed seed of 42.

A.4 Trait Fidelity Evaluation

Per-image trait captioning. For each species, we prompt an InternVL3-8B [57] to describe only visible traits. We explicitly exclude species names to avoid bias. In our experiments, we sample 10

images per species for the generated set, resulting in 10 corresponding trait captions. For real images, we caption all the available images of that species. We use the following prompt for per-image trait captions:

```
<image> Describe this image in detail.  Focus on the main subject, its
appearance, colors, shape, pattern, texture, position, and any distinctive
features.  Ignore the background.
```

Per-species Trait Summarization. For generating a per-species trait summarization, we employ LLaMA3.1-8B-Instruct [5] to consolidate the per-image captions into a single "trait caption" for each species. We use the prompt:

```
<captions> Each line is a caption from one image of this species.  Produce one
combined caption that reflects the shared visible traits, including colors,
patterns, shapes, markings, textures, and other distinctive parts.  Ignore the
background.
```

B Revisiting Main Results

Results on iNat-mini. Table 1 of the main paper presents results on the iNat-mini dataset. TaxaAdapter achieves consistent improvements across all evaluation metrics. For classification accuracy, we train a linear classifier using pre-trained DINOv2 features using a separate 500k image subset sampled from the remaining 2.2M iNat images to avoid data leakage; the base classifier achieves 72.26% top-1 accuracy on its validation split. For caption-based trait fidelity, we report BERTScore, CLIPScore, and Mistral-7B similarity for semantic alignment, and ROUGE-L for word-level overlap of species-defining traits between real and generated caption sets. TaxaAdapter achieves stronger trait alignment than all baselines.

We observe that SD1.5 outperforms the larger baselines. FID measures distributional similarity between generated images and the biodiversity reference dataset, whereas CAS (Classification Accuracy Score) measures how accurately the generated images preserve the prompted species identity. Consequently, lower FID does not necessarily imply stronger species fidelity. Across our results, SD3.5-Large and FLUX.1-dev generally achieve higher CAS than SD1.5, indicating better species-identity preservation despite worse FID. For example, on TREEOFLIFE-1M, FLUX achieves higher CAS@1 than SD1.5 (12.44% vs. 9.32%) but substantially worse FID (90.09 vs. 46.89).

This interpretation is further supported by generative precision and recall. FLUX achieves higher precision than SD1.5 (0.66 vs. 0.54), indicating stronger sample fidelity, but substantially lower recall (0.06 vs. 0.33), indicating limited coverage of the reference distribution. One possible explanation is a compositional mismatch: TREEOFLIFE-1M contains organisms across highly diverse natural settings, whereas larger open-domain models often produce polished, portrait-style, foreground-centric images with limited variation in pose, scale, and environment (see Figure 3). K. Adamkiewicz et al. [1] also show a similar finding that most recent text-to-image models collapse to generating a narrow, aesthetic-centric distribution that undermines diversity and real data distribution coverage. This mismatch possibly explains the larger models' poorer FID despite their stronger CAS.

Prompt familiarity further contributes to the results because generic models may not be explicitly trained to interpret complete Linnaean taxonomy strings. As shown in Appendix D.4 and Table 17, replacing these strings with common species names improves FLUX's FID from 90.09 to 68.29 and CAS@1 from 12.44% to 21.28%, confirming that familiarity with the prompt formulation substantially affects performance. Similar observations are made in Appendix D.5 and Table 18 where BioCAP captions are used for prompting. We also tried prompting all models to generate organisms in natural scenes. It produced similar results, suggesting that this behavior cannot be resolved through natural-scene prompting alone.

Results on TREEOFLIFE-1M. We further evaluate on the larger TREEOFLIFE-1M dataset, which covers a broader taxonomy span and species diversity, making it a substantially harder benchmark than iNat-mini. As observed in Table 2, TaxaAdapter maintains strong performance and generalizes effectively at scale, achieving substantial gains in both image quality and taxonomy–image alignment.

Notably, the results on TREEOFLIFE-1M surpass those on iNat-mini, suggesting that more diverse training data enhances fine-grained species generation.

Results on FishNet. Table 3 of the main paper reports results on the fine-grained FishNet dataset. TaxaAdapter improves taxonomy–image alignment while attaining the lowest FID. We note a slight increase in LPIPS compared with TaxaDiffusion, which we hypothesize reflects the model's stronger emphasis on fine-scale trait fidelity rather than global texture smoothness. This trade-off diminishes as training scales to larger datasets, consistent with the trend observed in Table 1 of the main paper.

C Additional Results

C.1 Trait Fidelity Evaluation

Results on TREEOFLIFE-1M. Table 2 in the main paper reports trait-fidelity scores on TREEOFLIFE-1M using the MLLM-based pipeline described in Section 4 of the main paper. For each species, we first generate image-level captions from InternVL3-8B and then summarize them into a single trait caption with LLaMA3.1-8B-Instruct. This per-species summarization is important for capturing stable morphological signatures: distinctive colors, patterns, or body parts may be occluded or absent in any single image due to pose, orientation, or partial visibility. Aggregating over multiple captions yields a more robust description of species-level traits and reduces noise from background clutter.

To disentangle the effect of this summarization step, Table 9 evaluates an instance-wise variant of the same pipeline. Here, we directly compare captions of real and generated images without summarization, compute similarity scores for each image pair, and then average over images within each species. Despite being more sensitive to per-image variability, this image-to-image setting shows the same ordering as the summarized variant. TaxaAdapter consistently outperforms all baselines across BERT Score, CLIP Score, ROUGE-L, and Mistral-7B text similarity. We additionally experimented with a one-to-many matching scheme, where each generated image caption is compared against all real image captions of the same species; this produced similar relative performance.

Table 9: Quantitative results on instance-wise trait-fidelity for TREEOFLIFE-1M. Results are evaluated using 500 species on TREEOFLIFE-1M.

Model	BERT↑	CLIP↑	ROUGE-L↑	Mistral-7B↑
SD1.5	0.30	0.61	0.13	0.87
SD3.5-Large	0.28	0.61	0.13	0.87
FLUX.1-dev	0.30	0.62	0.13	0.87
TaxaAdapter	**0.35**	**0.67**	**0.31**	**0.89**

Results on iNat-mini. Table 10 reports the instance-wise variant trait similarity as seen above on the iNat-mini dataset. Again, we observe that TaxaAdapter achieves the highest scores on all four caption-based metrics, indicating that its improvements are not limited to a particular dataset or evaluation variant. Taken together, the summarized and instance-wise results across TREEOFLIFE-1M and iNat-mini provide consistent evidence that taxonomy-aware conditioning leads to captions that more faithfully capture species-defining visual traits than those produced by strong off-the-shelf text-to-image models.

Examples of Trait Fidelity Evaluation. Figure 7 shows trait captioning results for species "*Animalia, Chordata, Aves, Charadriiformes, Laridae, Sterna hirundo*" with the common name 'Common tern' and "*Animalia, Chordata, Aves, Passeriformes, Passerellidae, Amphispiza, bilineata*" with the common name 'Black-throated Sparrow', respectively. We show the captions generated for different images and the trait-summarized captions across all the images for each species.

C.2 User Study for Trait Fidelity Evaluation.

To validate whether our generated text summaries are faithful to visual traits, we conduct a small human evaluation guided by curated Wikipedia descriptions. We sampled 10 species randomly from the iNat-mini dataset and, for each species, the evaluators were shown (i) a small set of real images for reference, and (ii) a ground-truth textual description extracted from Wikipedia that emphasizes visual

Table 10: Quantitative results on instance-wise trait-fidelity for iNat-mini.

Model	BERT↑	CLIP↑	ROUGE-L↑	Mistral-7B↑
SD1.5	0.32	0.66	0.29	0.89
SD3.5-Large	0.32	0.66	0.29	0.89
FLUX.1-dev	0.34	0.69	0.30	0.90
TaxaAdapter	**0.38**	**0.73**	**0.33**	**0.92**

traits (e.g., coloration patterns, shapes, etc.). We then show the MLLM-generated trait summary for that species and ask evaluators to rate its correctness and completeness with respect to the reference images and Wikipedia text. We had 12 evaluators, including evolutionary biologists and experts actively working at the intersection of machine learning and ecology/biology. The study yielded an average rating of $4.25/5.0$, suggesting that the generated summaries capture salient, visually grounded traits.

C.3 Trait Fidelity Evaluation with BioCAP captions as biologically informed external references.

To further validate whether our caption-based trait-fidelity evaluation captures biologically meaningful visual traits, we compare generated-image trait summaries against BioCAP [55] reference captions. Unlike our main evaluation, where real-image references are produced using the same MLLM-based pipeline, BioCAP provides independent trait-rich descriptive captions grounded in Wikipedia context and taxon-specific examples. This allows us to evaluate whether generated images align with external biologically informed trait descriptions rather than only with MLLM-generated summaries from real images. As shown in Table 11, TaxaAdapter achieves the best performance across all four metrics, suggesting that TaxaAdapter produces images whose inferred visual traits better align with biologically informed external references

This consistency also supports our main evaluation protocol: using real images as reference trait ground truth through the same captioning pipeline yields rankings consistent with independent BioCAP references, making the proposed metric scalable to taxa for which curated trait descriptions are unavailable.

Table 11: Trait Fidelity Evaluation using BioCAP captions as biologically informed external references.

Model	BERT↑	CLIP↑	ROUGE-L↑	Mistral-7B↑
SD 1.5 (ZS)	0.28	0.80	0.24	0.88
SD 3.5-Large (ZS)	0.31	0.81	0.26	0.90
FLUX.1-dev (ZS)	0.34	0.83	0.29	0.91
SD 3.5-Large (LoRA)	0.33	<u>0.87</u>	0.28	0.91
FLUX.1-dev (LoRA)	0.31	0.85	0.28	0.91
TaxaAdapter (w/BioCLIP)	**0.41**	**0.88**	**0.32**	**0.93**
TaxaAdapter (w/TaxaBind)	<u>0.36</u>	0.86	<u>0.30</u>	<u>0.92</u>

C.4 Free-form Text Conditioning

Figure 8 shows additional free-form prompting results enabled by TaxaAdapter. We set $\lambda = 1$ during training (Eq. 4) and use $\lambda = 1$ at inference for our default biologically-grounded generation setting, where the text branch is fixed to "a photo of y_{taxa}" and biological precision is the primary goal. We vary λ *only at inference* to enable free-form text control. With $\lambda = 0.5$, the model maintains species-faithful morphology while supporting diverse contextual prompts. We do not tune λ beyond these settings. Figure 9 shows a quantitative analysis characterizing the effect of λ on image quality and text alignment and motivates $\lambda = 0.5$ for free-form prompting.

C.5 Additional Qualitative Results

Figures 11, 12 and 13 show additional qualitative results of TaxaAdapter compared to the baselines.



Figure 7: Qualitative results for trait fidelity analysis using our captioning-based evaluation method. The upper case shows the trait captions generated using MLLM for real and generated images for species "*Sterna hirundo*" along with the summarized captions. The bottom case shows the trait-fidelity captions for species "*Amphispiza bilineata*".

C.6 Additional OOD results

We additionally evaluate TaxaAdapter on 100 fish species from the FishNet [14] datasets that were absent from both VTM pretraining and TaxaAdapter training, using the model trained only on the TreeOfLife-1M dataset. Table 12 shows that TaxaAdapter substantially improves species-level fidelity, achieving 8.30% CAS@1 and 24.60% CAS@5, compared with the strongest baseline results of 1.90% and 10.70%, respectively. It also achieves the best LPIPS (0.773), although SD3.5 obtains a better FID. These results extend the OOD evaluation beyond birds to a taxonomically distinct group and show a consistent improvement in species fidelity for previously unseen fish species.

Table 12: OOD results of TaxaAdapter on 100 FishNet species.

Model	FID ↓	LPIPS ↓	CAS@1 ↑	CAS@5 ↑
SD1.5	<u>73.65</u>	<u>0.799</u>	1.80%	5.70%
SD3.5	**64.83**	0.811	<u>1.90%</u>	<u>10.70%</u>
FLUX.1-dev	85.32	0.802	1.30%	7.30%
TaxaAdapter	82.48	**0.773**	**8.30%**	**24.60%**

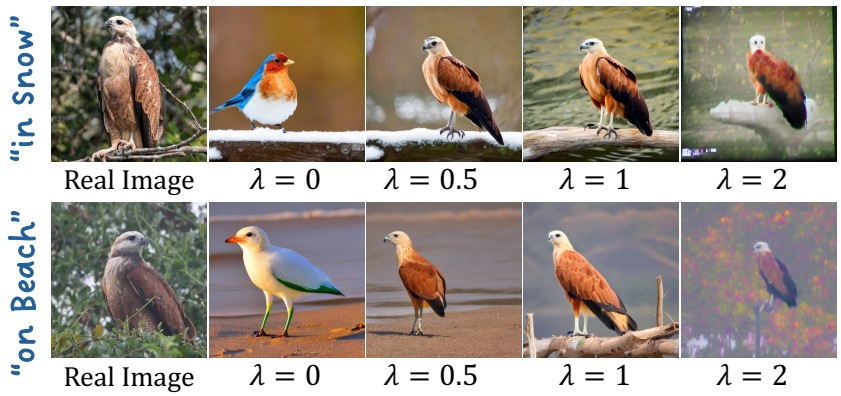


Figure 8: Different weighting factors λ during inference for free-form text generation. Intermediate values (e.g., $\lambda = 0.5$) balance both conditions and yield morphology-faithful generations. Note that the setting of this figure is different from Figure 6 in the main paper, where we show general generation results.

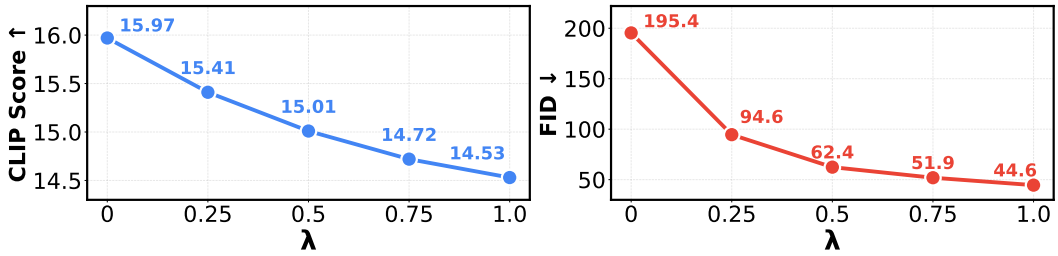


Figure 9: FID indicates species alignment; CLIP Score represents free-form text alignment (w.r.t. pose, background, etc.).

For OOD on birds, as in Section 5.3, the 51 CUB test species are unseen during TaxaAdapter training for the model trained on the iNat-mini dataset. Among them, 16 species were present during VTM pretraining, whereas 35 were also absent from VTM pretraining after matching the taxonomic strings. The latter subset evaluates generalization to species unseen by both the adapter and the pretrained VTM, while the former evaluates whether TaxaAdapter can transfer knowledge already represented by the VTM to species excluded from adapter training, as shown in Tables 14 and 13. We can further characterize the 35 species by their taxonomic proximity to the TaxaAdapter training distribution: 14 classes share the same genus, whereas the remaining 21 share a family but not a genus+species with their closest training relative.

Table 13: Results on 35 CUB species unseen during both VTM pretraining and TaxaAdapter training.

Model	FID ↓	LPIPS ↓	CAS@1 ↑	CAS@5 ↑
SD1.5	**37.03**	0.792	7.14%	25.43%
SD3.5	45.24	0.794	4.29%	18.57%
FLUX.1-dev	57.71	**0.769**	3.43%	16.57%
TaxaAdapter	<u>42.61</u>	<u>0.788</u>	**29.43%**	**60.57%**

C.7 Mixed or Hybrid Species

Beyond synthesizing known taxa, we show that TaxaAdapter exhibits compositional behavior in the VTM-conditioned embedding space. We do this by modifying the taxonomic name by mixing the names of two different species, for example, given "Oenanthe familiaris" and "Ploceus ocularis", we construct the hybrid species "Oenanthe *ocularis*" and synthesize samples for this hybrid taxon. As shown in Figure 10, the mixed species inherits the black eye line trait from the reference species, demonstrating that VTM-aligned taxonomy embeddings can generate images that combine fine-grained morphological traits associated with different taxonomic levels. We, however, note that

Table 14: Results on 16 CUB species unseen during TaxaAdapter training only.

Model	FID ↓	LPIPS ↓	CAS@1 ↑	CAS@5 ↑
SD1.5	82.28	0.783	15.00%	26.88%
SD3.5	80.21	0.806	16.88%	35.62%
FLUX.1-dev	90.16	0.816	13.75%	28.75%
TaxaAdapter	**73.23**	**0.753**	**59.38%**	**85.00%**

Table 15: Analysis of taxa encoder in TaxaAdapter. Models are trained on iNat-mini-birds (1500 species) and evaluated on 650 bird species.

	Taxa Encoder	FID↓	LPIPS↓
(a)	CLIP [32]	29.16	0.747
(b)	BioTrove-CLIP [47]	21.51	0.745
(c)	BioCLIP [6]	17.08	0.745
(d)	TaxaBind [38]	16.61	0.742

such mixed or hybrid species do not exist and are hence difficult to validate. We present them as an interesting direction for future studies for CRISPR-style [28] experiments for trait/gene editing.

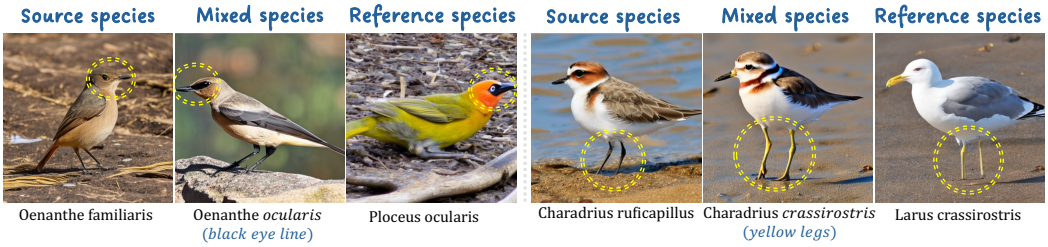


Figure 10: Synthetic "mixed or hybrid" species via taxonomic name mixing.

D Additional Ablations and Analyses

D.1 Compatibility with Different VTMs.

Tab. 15 shows the effect of varying the VTM backbone from BioCLIP to TaxaBind [38] and BioTrove-CLIP [47]. We see that all VTM backbones achieve comparable gains over the CLIP baseline, demonstrating that TaxaAdapter works robustly with different VTMs and real benefits stem from domain-matched taxonomy encoders and not just fine-tuning.

D.2 Replacing CLIP with BioCLIP

In Section 5.4, we studied the key design choices of TaxaAdapter. Here, we provide further justification for the taxonomy encoder design.

Naïve fine-tuning with taxonomy (Table 7-(b)). In the main paper, we describe that directly fine-tuning generative models with taxonomy information leads to suboptimal performance. Specifically, we take a subset of bird species from the iNat-mini dataset and train the model on 1500 species while evaluating on 650 species, sampling one species per `Genus` level in the taxonomy. We use zero-shot SD 1.5 as the baseline (row (a) in Table 7). We then fine-tune SD 1.5 with LoRA by feeding taxonomy labels into its CLIP encoder. While this yields improvements in FID and LPIPS, it still underperforms TaxaAdapter.

Replacing CLIP with BioCLIP (Table 7-(c)). Next, we attempt the non-trivial substitution of the SD 1.5 CLIP text encoder with a VTM, BioCLIP (row (c)). Unlike the CLIP text encoder used in Stable Diffusion, which exposes a sequence of contextualized token embeddings of shape $[77 \times 768]$ for cross-attention, BioCLIP is typically used via its pooled and projected embeddings for classification tasks, having embeddings of shape $[1 \times 768]$. Directly feeding this pooled vector

Table 16: Results of TaxaAdapter using different diffusion backbones on the iNat-mini birds subset. Both models use the same BioCLIP taxonomy encoder.

SD backbone	FID ↓	LPIPS ↓
TaxaAdapter (SD1.5)	17.08	0.745
TaxaAdapter (SDXL)	13.39	0.743

into the diffusion model would break the expected interface and discard the token-level structure. To make BioCLIP compatible with the diffusion backbone, we bypass its pooling and projection head and get the final-layer token embeddings to obtain the conditioning sequence for cross-attention of size $B \times 77 \times 768$.

Despite these efforts, this variant achieves worse performance compared with naively fine-tuning the CLIP encoder. This suggests that the model requires a significant amount of training to achieve taxonomy-to-image alignment. Also, since Stable Diffusion is largely pre-trained with CLIP, it is not easily replaceable without pre-training on a large dataset. All the fine-tuning was done similarly for 100 epochs using the same hyperparameters as TaxaAdapter.

D.3 Compatibility with Other Diffusion Backbones

One of the goals of TaxaAdapter is to serve as a plug-in taxonomy adapter that can be attached to different text-to-image backbones with dual-text encoders without fine-tuning of the model. To test this, we now train the model on SDXL [31] for 512×512 images. In both cases, we use the original BioCLIP embeddings. Table 16 shows the results. We observe that upgrading the backbone from SD 1.5 to SDXL consistently improves both image quality and taxonomy–image alignment. FID drops from 17.08 to 13.39, while LPIPS remains essentially unchanged. These results indicate that the adapter cleanly factors out taxonomy control from low-level image synthesis: stronger backbones directly translate into better fine-grained generations, while the taxonomy branch continues to provide stable species-level guidance. Crucially, no additional tuning of the SDXL backbone is required beyond training the same lightweight adapter, demonstrating that TaxaAdapter is generalizable.

D.4 Taxonomic Name vs. Common Name

For state-of-the-art models, like FLUX.1-dev we don't know what data they were trained on. We evaluate its performance using both taxonomic names and the common names of the species for comparison. Concretely, we evaluate FLUX.1-dev on the TREEOFLIFE-1M subset of 500 species using two prompt variants per species: (i) the full Linnaean taxonomic string (*taxonomic name*, TN), for example, "*Animalia, Chordata, Aves, Passeriformes, Passerellidae, Spizella, Breweri*" and (ii) the common name (CN), for example, Brewer's sparrow. TaxaAdapter is evaluated with taxonomic-name prompts, which are also used as input to the VTM.

Table 17 shows that FLUX.1-dev is much more aligned with common-name prompts. Switching from TN to CN reduces FID (from 90.09 to 68.29), and nearly doubles CAS@1/5. However, even under the favorable setting of using common names, FLUX.1-dev (CN) remains far behind our taxonomy-aware TaxaAdapter (TN). With taxonomic-name prompts only, TaxaAdapter attains an FID of 29.87. These suggest that explicit taxonomy-aware conditioning is essential for robust species-level control in biodiversity synthesis.

Table 17: Quantitative comparison on TREEOFLIFE-1M. Results are evaluated using 500 species on TREEOFLIFE-1M. For FLUX.1-dev, we show the difference in performance when using the taxonomic name vs the common name. TN: taxonomic name as prompt. CN: common name as prompt.

Model	FID↓	LPIPS↓	CAS@1↑	CAS@5↑
FLUX.1-dev (TN)	90.09	0.800	12.44%	31.14%
FLUX.1-dev (CN)	68.29	0.803	21.28%	42.42%
TaxaAdapter (TN)	**29.87**	**0.734**	**61.98%**	**82.96%**

D.5 T2I baseline using detailed morphological descriptions/ retrieval-augmented prompts

We use trait-descriptive prompts in text-to-image models as a useful complementary approach to generating morphologically accurate images when such trait descriptions are available. We conduct experiments using BioCAP [55] captions as text prompts, which contain detailed descriptions of species appearance, traits, and morphology. This setting also represents a retrieval-based approach where a detailed species description is retrieved (from BioCAP) and supplied to a strong text-to-image model.

We extend Table 1 results in Table 18 and show that BioCAP prompting improves over the taxonomy-only baselines (Flux FID drops from 74.14 to 42.25), highlighting that richer morphological descriptions are beneficial for generation. However, these models still fall behind TaxaAdapter, particularly in species identity, where FLUX.1-dev achieves CAS@1 of 11.69% vs 61.36% for TaxaAdapter.

Table 18: Quantitative comparison of TaxaAdapter with zero-shot (ZS) text-to-image generation models using BioCAP captions.

Model	FID ↓	LPIPS ↓	CAS@1 ↑	CAS@5 ↑
SD 1.5 (ZS w/ taxonomic name)	31.43	0.768	1.24%	2.86%
SD 3.5-Large (ZS w/ taxonomic name)	70.63	0.806	1.86%	4.48%
FLUX.1-dev (ZS w/ taxonomic name)	74.14	0.801	1.78%	4.21%
SD1.5 (ZS w/ BioCAP captions)	19.84	0.761	8.45%	18.23%
SD3.5 (ZS w/ BioCAP captions)	36.85	0.783	13.23%	27.19%
FLUX.1-dev (ZS w/ BioCAP captions)	42.25	0.778	11.69%	25.04%
TaxaAdapter (w/ BioCLIP)	**13.31**	**0.742**	**61.36%**	**81.99%**

This shows that text captions certainly provide an opportunity to increase morphologically faithful generations (when available), but supplying descriptive text alone is hard to scale and also insufficient to match the benefits of the image-aligned taxonomic features learned directly from images without any trait labels. TaxaAdapter does not involve training and curation of any corpus of textual descriptions, making it trait label-free. A promising future direction is to explore combining descriptive text with VTM conditioning in TaxaAdapter.

D.6 Isolating the contribution of hierarchical taxonomy

To isolate the contribution of hierarchical taxonomic conditioning, we train TaxaAdapter models on 1103 iNaturalist species, selecting one species from each family, and altering the taxonomic string supplied to the VTM. Specifically, we compare the canonical seven-level taxonomy against two controls: (i) randomly permuting the same taxonomic ranks, thereby disrupting their canonical root-to-leaf order, and (ii) replacing the higher-level ranks with those of a different species, thereby producing a biologically inconsistent hierarchy.

Table 19 shows that the complete, correctly ordered seven-level taxonomy achieves the highest species-identity performance, with 73.64% CAS@1 and 86.70% CAS@5. Randomizing the order produces the worst results, reducing CAS@1 to 43.44% and degrading FID from 13.12 to 16.87. More importantly, corrupting the higher-level ranks while retaining the lower-level identity also decreases CAS@1 to 71.11% and CAS@5 to 84.73%. This indicates that biologically consistent higher ranks provide complementary lineage context for preserving fine-grained species morphology.

Since, BioCLIP was pretrained using a mixture of scientific names (genus+species), common names, and complete taxonomic strings. Hence, sending a scientific name (genus+species) as input to BioCLIP is bound to implicitly recover the complete hierarchical information through the pretrained VTM. Hence, masking higher-level taxonomic information in the inputs to VTMs may not help us to evaluate the importance of higher levels of the taxonomy hierarchy.

E Limitations

While TaxaAdapter demonstrates strong taxonomy-aware generation across multiple biodiversity datasets, it has certain limitations. One limitation of the current setup is that the free-form text

Table 19: Ablation study on the effect of hierarchical taxonomic information in TaxaAdapter.

Hierarchical Taxonomy	FID ↓	LPIPS ↓	CAS@1 ↑	CAS@5 ↑
Random order	16.87	0.748	43.44%	62.03%
Corrupted higher levels	13.44	0.747	71.11%	84.73%
All 7 levels	**13.12**	**0.743**	**73.64%**	**86.70%**

conditioning is biased toward backgrounds and contexts that appear in the training data. When prompts describe species in rare or unseen environments (e.g., "a photo of *Spizella breweri* perched on the roof of a car"), the model often fails to generate accurate images.

Another limitation is that our model still inherits challenges from data quality and sparsity. Species or clades that occupy only a small fraction of the frame in the images remain challenging. In such cases, we observe failures where the generated samples blur fine-grained markings or allow background textures to dominate.

Lastly, our evaluation of visual trait fidelity relies on large vision–language models and LLM-based trait summarization. While these provide a scalable proxy in the absence of expert-annotated trait captions, they introduce their own biases and may under- or over-emphasize certain traits relative to expert-defined descriptors. A more thorough integration with expert-curated trait databases and human evaluations is an important direction for future work.

F Ethics and Social Impact

Our work focuses on species generation using public image datasets and taxonomy data, with the aim of accelerating scientific discovery, research, and usage. We do not use any human subjects or personally identifiable information, and we do not foresee any negative impacts or ethical concerns with our work. The primary intended use of TaxaAdapter is to support scientific exploration, education, and data augmentation for biodiversity and ecological research.



Figure 11: Additional qualitative results on TREEOFLIFE-1M.



Figure 12: Additional qualitative results on iNaturalist.

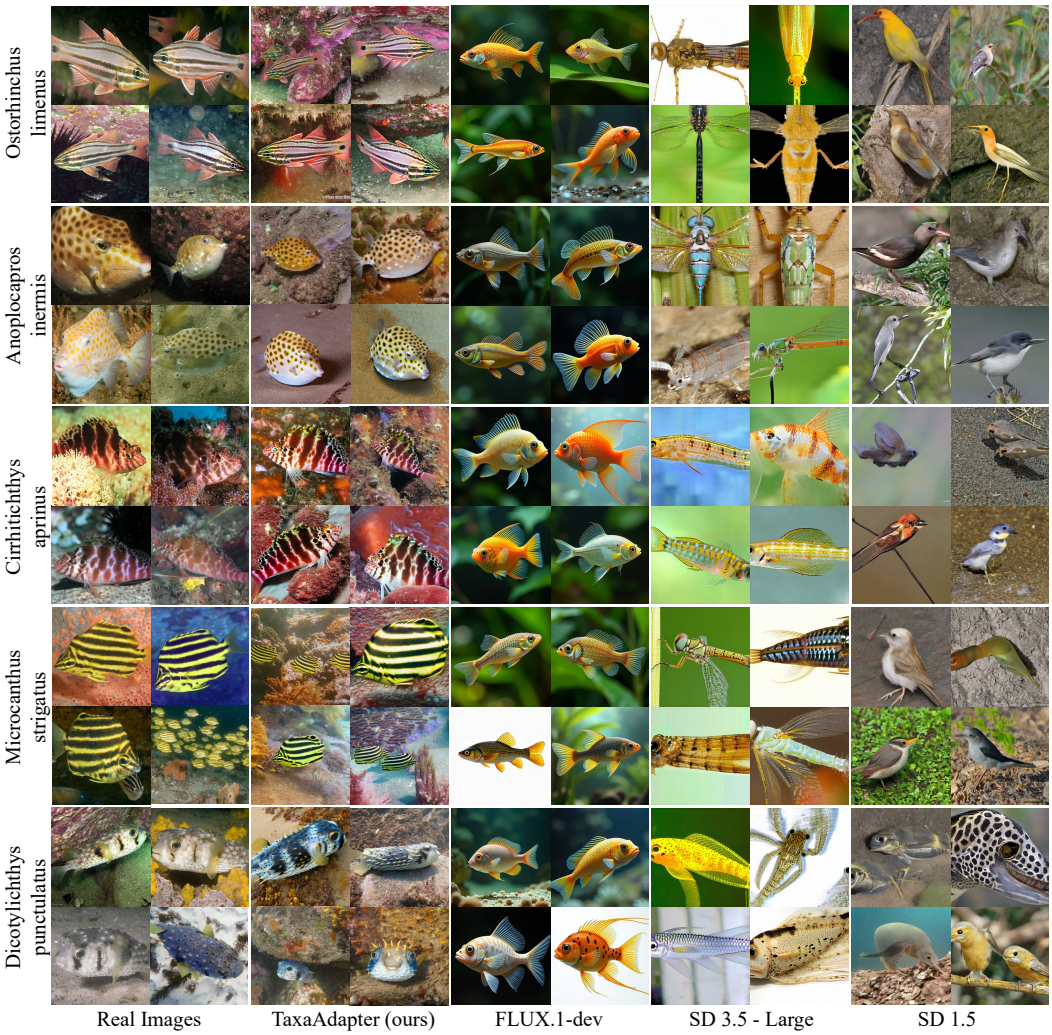


Figure 13: Additional qualitative results on FishNet.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: Yes, introduction is discussed in Sec. 1. Both the introduction and abstract state the main contribution as proposing TaxaAdapter as a simple and lightweight adapter that incorporates the taxonomic embeddings of Vision Taxonomy Models (VTMs) such as BioCLIP to perform scalable fine-grained species generation over the entire Tree of Life. This enables rare and unseen species generation and improves species-level fidelity while preserving flexible text control over attributes such as pose, style, and background.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Appendix E discusses the limitations and failure cases of our work with species occupying only a small fraction of frame in the images and occluded with background, which are difficult to learn.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [N/A]

   Justification: The paper does not claim any theoretical assumptions. Appendix A.1 provides all theoretical equations and formulas needed.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Yes, Section 5 lists the experimental setup, implementation details, and training details, with additional hyperparameter details necessary to reproduce results are reported in Appendix A.2 and A.3.

   Guidelines:

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

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: Datasets used are open access and are listed in Section 5. Source code is available on Github and models are released at https://imageomics.github.io/TaxaAdapter/

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Yes, Section 5 lists the experimental setup, implementation details, and training details, with additional hyperparameter details are reported in Appendix A.3.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [No]

   Justification: We do not report error bars for all of the main results. All benchmarks are done with a fixed seed mentioned in Appendix A.3 to reproduce the results.

   Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Yes, Appendix A.3 lists the compute needed to run the model. It lists the time taken to run the model, the number of GPUs used, and the type of GPUs used along with training epochs, batch size, etc..

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: Yes, the research is conducted, which conforms to and respects the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

Answer: [Yes]

Justification: Yes, Appendix F discusses the impacts. Our work focuses on species generation using public image datasets and taxonomy data, with the aim of accelerating scientific discovery, research, and usage.

Guidelines:

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

11. **Safeguards**

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

Answer: [N/A]

Justification: The paper does not pose such risks. The models that will be released are useful for biologists for scientific discovery and advancements, and do not pose risks.

Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: The paper cites the original owners and creators of models and architectures by citing and acknowledging the authors and build on their work.

Guidelines:

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

13. **New assets**

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

Answer: [Yes]

Justification: Source code is available on Github and models are released at https://imageomics.github.io/TaxaAdapter/

Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [Yes]

Justification: LLMs are not part of the TaxaAdapter generation method. However, our proposed trait-fidelity evaluation (Section 4) uses InternVL3-8B to caption real and generated images and LLaMA-3.1-8B-Instruct to summarize per-species trait descriptions. Mistral-7B is used as one of the text-similarity scorers. The models, prompts and protocol are described in Section 4 and Appendix A.4. We validate the protocol with an expert user study (Appendix C.2) and against BioCAP references (Appendix C.3), and discuss its limitations in Appendix E.

Guidelines:

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