Variance Constrained Distribution Alignment in Few Shot Models

Xiaohong Cai **Yi SUN*** **Zhaowen Lin*** **Tianwei Cai**
School of Computer Science (National Pilot Software Engineering School),
Beijing University of Posts and Telecommunications, Beijing, China
*Corresponding authors:{sybupt, Linzw}@bupt.edu.cn

Abstract

Learning generative models from the limited samples remains challenging due to unstable estimation of class conditional representations. Such instability often leads to intra-class distribution drift and degraded generalization under few sample regimes. To address these challenges, we propose a method that can model class level latent distributions for flexible and efficient few shot synthesis. Specifically, each input is represented by a learnable conditional latent distribution. Metric based statistical modeling effectively disentangles latent variables, contracts intra-class variance, and enlarges inter-class margins while enforcing cross task distributional alignment. Furthermore, we provide a variance based generalization analysis, showing that controlling class conditional variance tightens generalization bounds under few sample regimes. Experiments on the benchmark datasets demonstrate that our method surpasses prior works in visual quality and diversity, highlighting the benefit of statistical alignment for robust few shot generative modeling.

1 INTRODUCTION

Few shot generative modeling offers a promising way to mitigate data scarcity by synthesizing diverse samples of rare categories from only a handful of examples (Clouâtre and Demers, 2019; Liang et al., 2020). However, learning reliable generative models in this regime is fundamentally constrained by the statistical instability of class conditional estimation. With only limited samples available per class, empirical representations often exhibit high variance and distributional drift, which distort latent structures and hinder generalization across tasks. The key challenge is therefore to control class conditional variance and align distributions which requires a principled statistical treatment of latent representations, rather than task-specific architectural modifications (Clouâtre and Demers, 2019) or heuristic regularization (Hong et al., 2020b).

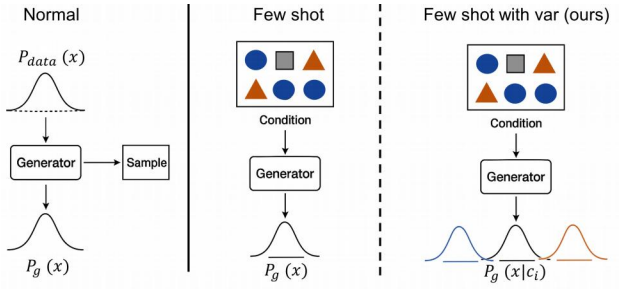


Figure 1: General framework of generative modeling.

Existing approaches including meta-learning (Clouâtre and Demers, 2019; Liang et al., 2020), fusion (Hong et al., 2020a,b; Gu et al., 2021), transformation (Hong et al., 2022a,b), and latent space editing (Ding et al., 2022; Li et al., 2023; Ding et al., 2025) offer a complementary way to augment scarce categories by synthesizing additional samples. However, these methods largely overlook the statistical variance inherent in class level representations, making them vulnerable to distribution drift and overfitting, which degrades visual fidelity, reduces diversity, and hinders generalization. At its core, few shot image generation can be viewed as estimating the underlying class distribution from limited observations. Prior approaches implicitly attempt this through feature fusion, latent transformation, or architectural heuristics, but without an explicit statistical formulation they lack both stability and theoretical guarantees. This motivates us to directly formulate few shot generation as class level

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

distribution estimation, enabling principled variance control and stronger generalization.

In this work, we explicitly model each category as a class conditional latent distribution $P(z|c)$, which captures the structural variability of samples belonging to class c in the latent space. Unlike classical generative models (Kingma and Welling, 2013; Goodfellow et al., 2014; Ho et al., 2020) that assume abundant training data, our approach accounts for the high variance nature of class conditional distributions in few shot settings. By treating few shot generation as a problem of statistical distribution estimation, our approach provides a principled way to stabilize class level representations, reduce overfitting to limited samples, and improve generalization across tasks. This perspective not only clarifies the source of instability in existing methods, but also establishes a foundation for generating diverse and realistic samples while offering theoretical insights into generalization under extreme data scarcity. Empirically, our framework demonstrates that stable estimation of class level distributions leads to consistent improvements in both visual quality and diversity across multiple benchmarks.

Our main contributions are summarized as follows: (1) We propose class level distribution estimation for few-shot generation, offering a statistical perspective that pinpoints instability sources and emphasizes the role of variance control in generalization. (2) We constrain class conditional variance and derive generalization guarantees in few shot regimes, effectively mitigating distribution drift. (3) We establish stable class representations without costly fine-tuning or task specific architectures, significantly improving generation fidelity and diversity. (4) We validate approach we proposed on multiple few-shot benchmarks, consistently outperforming prior methods and confirming the effectiveness of our statistical modeling paradigm.

2 RELATED WORK

2.1 Few Shot Generative Modeling

Few shot image generation addresses the problem of learning to generate diverse and high-fidelity images from only few samples of unseen categories. A central challenge lies in balancing distribution generalization with category-specific adaptation, while maintaining structural fidelity and semantic coherence. Existing methods include four categories. Meta-learning approaches like FIGR (Clouâtre and Demers, 2019) and DAWSON (Liang et al., 2020) enhance generalization through cross-task training but neglect structure, leading to unstable quality and limited to simple datasets like MNIST (Yann, 2010) and Omniglot (Lake et al., 2011). F2GAN (Hong et al., 2020b) combines feature fusion with region filling, while MatchingGAN (Hong et al., 2020a) uses feature matching to improve image consistency. LoFGAN (Gu et al., 2021) integrates local features for better detail expression. While these methods improve quality, the fusion processes may introduce structural artifacts or semantic shifts, compromising image authenticity. Transformation transfer methods model category transitions by capturing transformation increments. DeltaGAN (Hong et al., 2022a) allows more flexible category transformation, while Disco-FUNIT (Hong et al., 2022b) improves image diversity by learning discrete content representations. However, these methods lack semantic interpretability and structural consistency. Latent space editing approaches including Attribute Group Editing (AGE) (Ding et al., 2022) and its stable version SAGE (Ding et al., 2025) improve semantic control by grouping attributes. HAE (Li et al., 2023) shifts attribute space from Euclidean to hyperbolic space for better alignment of few-shot sample distributions. While effective in semantic modeling, existing methods often depend on manual directions and lack structural awareness, causing inconsistencies or deformation. In earlier work, GPN (Fort, 2017) also pointed out that instance level mapping may introduce bias and limit sample diversity. To address this, they introduces additional variation through a covariance matrix to broaden the coverage of the latent space. In contrast, we propose category aware distribution modeling which directly treats novel categories as class-conditional distributions.

2.2 Latent Distribution Adaptation

Pre-trained generators in few shot settings often suffer from limited coverage and poor reconstruction. Existing solutions either fine-tune the generator parameters (Roich et al., 2022) or adapt latent codes while keeping generators fixed (Li et al., 2023; Abdal et al., 2021; Wu et al., 2021). However, the latent space structure is neither optimal nor static, and its adaptability can be exploited for distributional alignment across categories. Recent approaches introduce lightweight adaptation modules or semantic regularization to shift representations from instance level to class level, thereby improving generalization and controllability. We emphasize class level latent alignment as a statistical mechanism to stabilize class distributions while enabling directional controllability and localized editing.

2.3 Representation Disentanglement

Building on the aligned class conditional distributions, representation disentanglement and factorization can be performed in a statistically grounded way.

Prior works include supervised approaches using annotated attributes (Yang et al., 2021; Shen et al., 2020; Goetschalckx et al., 2019), unsupervised factorization on instance-level or global latent spaces (Härkönen et al., 2020; Shen and Zhou, 2021; Park et al., 2023; Zhu et al., 2021, 2022), and multimodal-guided approaches (Radford et al., 2021; Park et al., 2023; Xia et al., 2021; Gal et al., 2022; Kawar et al., 2023) such as CLIP (Radford et al., 2021). However, these methods typically ignore class level distributions and regional constraints, which may lead to entangled factors and inconsistent edits. By conditioning factorization on learned class distributions and using adaptive CLIP-guided masks, our approach achieves category-consistent, localized, and robust semantic editing, while further validating the quality of the learned distributions.

2.4 Regional Constraints

Regional constraints improve localized control in semantic editing and help maintain consistency with class level distributions. Prior works, such as EditGAN (Ling et al., 2021) and RSeFa (Zhu et al., 2022), introduced explicit masks or localized semantic directions to refine attribute edits, while DiffEdit (Couairon et al., 2023) applied similar ideas to diffusion models. However, these approaches typically rely on manual or weak supervision and do not leverage class distribution information. We incorporate zero-shot segmentation from SAM (Kirillov et al., 2023) to guide semantic direction discovery on top of learned class distributions, enhancing local accuracy, distributional consistency, and structural stability in few shot generation.

3 PROPOSED METHOD

In this section, we first describe the problem setting (Subsection 3.1). We then introduce a method to estimate class conditional latent distributions from few shot data (Subsection 3.2), followed by a variance guided analysis that provides theoretical justification for our approach (Subsection 3.3). Next, we present the loss functions and training procedure for distribution learning under variance constraints (Subsection 3.4). Finally, we show how class-aware latent manipulation enables principled semantic editing and validates the reliability of the estimated distributions (Subsection 3.5).

3.1 Problem Setting

We consider the few-shot generative modeling problem, where the goal is to generate diverse and class-consistent images from only a limited number of examples per class. To address this, we adopt a meta-learning paradigm with episodic training. Unlike traditional metric learning (Guo et al., 2022), which constructs episodes through dataset sampling, we explicitly build task subsets in each episode consisting of a support set S (comprising K_S real image samples per class) to build the category center, and a query set Q (comprising K_Q generated samples) whose latent codes are optimized to align with the corresponding category center, promoting a clearer and more consistent class-level latent structure. Each episode contains N classes, represented as:

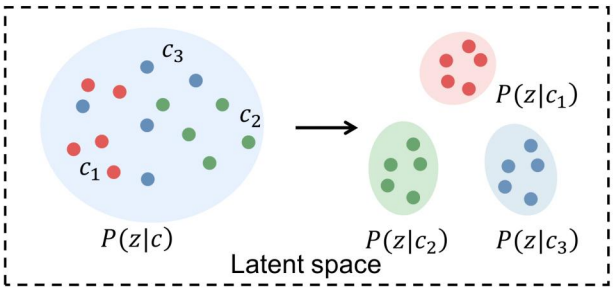


Figure 2: Illustration of class distribution modeling in latent space. Three class c_1, c_2, c_3 are shown here.

$$S = \{(x_i^c, l_i^c)\}_{i=1}^{K_S}, Q = \{(\hat{x}_j^c, l_j^c)\}_{j=1}^{K_Q}, c = 1, ..., N \quad (1)$$

where x_i^c is the i-th image from class c in support set S, with label l_i^c. Q is the query set, where $\hat{x}_j^c$ is the j-th generated image for class c, with label l_j^c.

The objective is to learn a generative model G such that:

$$\hat{x}_j^c \sim G(z|c) \quad (2)$$

where each generated sample $\hat{x}_j^c$ should faithfully reflect the semantic content of class c while exhibiting sufficient diversity. By explicitly modeling class conditional distributions, the framework moves beyond single prototype reconstruction and enables the generation of multiple semantically consistent and diverse instances per class.

3.2 Few Shot Class Distribution Estimation

In few-shot generative modeling, our goal is to generate diverse and class-consistent images given only a few samples per class. We formalize the latent representations of input image $x \in \mathbb{R}^{H\times W\times 3}$ as random variables in the $\mathcal{W}+$ (Abdal et al., 2019) space of StyleGAN2 (Karras et al., 2019, 2020). Given an input image x, we first obtain its instance-level latent code with pSp (Richardson et al., 2021):

$$w = pSp\,(x) \quad (3)$$

Here, $w \in \mathbb{R}^{18\times512}$ is a realization of the underlying class-conditional latent distribution $w^c \sim P(w|c)$ for class c, and pSp denotes the pre-trained encoder.

A single image provides limited information about $P(w|c)$, and thus directly using w may fail to capture the full class-level semantics, limiting diversity and generalization.

To address this, we introduce a class level *Latent Adapter* (LA) that shifts instance alignment toward shared category representation, enhancing semantic modeling and sample diversity. This alignment allows multiple instances from the same class to aggregate toward a unified class-level representation, enhancing semantic modeling and enabling the generation of diverse yet class consistent outputs. The LA maps instance level latents into a shared class aligned space:

$$z_i^c = LA(pSp(x_i^c)), \quad i = 1, \ldots, K_S \tag{4}$$

where K_S is the number of support samples per class and LA is essentially a lightweight multi-layer perceptron. This module infers the class level distribution parameters implicitly from the support samples and amortizes the estimation across tasks, providing stable results even when only a single example is available. This unified design means that the same estimator is applicable for $K_s = 1$ and $K_s \geq 2$, avoiding the instability of empirical covariance and enabling consistent distribution modeling across all regimes. The support set $\{z_i^c\}_{i=1}^{K_S}$ provides empirical observations to estimate the class mean and covariance:

$$\hat{\mu}_c = \frac{1}{K_S}\sum_{i=1}^{K_S} z_i^c \tag{5}$$

$$\hat{\Sigma}_c = \frac{1}{K_S - 1}\sum_{i=1}^{K_S}(z_i^c - \hat{\mu}_c)(z_i^c - \hat{\mu}_c)^\top \tag{6}$$

These estimates form the class level latent distribution $\mathcal{N}(\hat{\mu}_c, \hat{\Sigma}_c)$, from which we can later sample to generate diverse examples. Eq.(6) presents an ideal class-covariance formulation when $K_s \geq 2$, but we do not use it to compute covariance in our few-shot experiments (especially for $K_s = 1$). Instead, we use a learnable estimator to infer class distribution parameters.

Each query latent is adapted and then reconstructed by the generator:

$$\hat{z}_j^c = LA\left(pSp\left(\hat{x}_j^c\right)\right), \quad i = 1, \ldots, K_Q \tag{7}$$

Here, $\hat{z}_j^c \in \mathbb{R}^{18\times512}$. The reconstructed image can be expressed as:

$$\hat{x}_j^c = G\left(\ \hat{z}_j^c\ \right) \tag{8}$$

To encourage distributional alignment, we leverage a metric model $\mathcal{M}$ (Guo et al., 2022) to evaluate class consistency of query samples with the support set. It embeds each image into feature space, forms class prototypes by averaging support features, and outputs query scores based on the distances between query features and prototypes. We follow the original setting in (Guo et al., 2022) without modifying its architecture. Its output can be interpreted as a soft approximation of the posterior $P(c|\hat{x}_j^c, S)$:

$$y_{c,j} = \mathcal{M}\left(\ S,\ \hat{x}_j^c\ \right) \tag{9}$$

where $y_{c,j}$ represents the score of the j-th sample belonging to class c.

We then minimize the category-aware objective:

$$\mathcal{L}_{class} = \frac{1}{NK_Q}\sum_{c=1}^{N}\sum_{j=1}^{K_Q}\left(\left(\ y_{c,j} - l_j^c\ \right)\right)^2 \tag{10}$$

which regularizes query latents to cluster around the estimated class distributions, ensuring semantic consistency and inter-class separation.

Finally, once class latent distributions, including class mean μ_c and covariance Σ_c are estimated, we can generate diverse samples by sampling from distributions:

$$\hat{\mathbf{z}} \sim \mathcal{N}(\mu_c, \Sigma_c), \quad \hat{x} = G(\hat{\mathbf{z}}), \quad \forall c = 1, \ldots, N. \tag{11}$$

This probabilistic formulation bridges instance-level encoding and distribution-level generation, providing a principled mechanism to capture intra-class variability and synthesize multiple plausible samples per class. Building on this observation, we next provide a theoretical foundation, explaining how intra-class variance control and inter-class separation stabilize the estimation of class-conditional distributions and justify the effectiveness of our few-shot generation approach.

3.3 Variance Guided Generalization

Learning class-conditional distributions from only a few samples is inherently unstable, as empirical estimates can exhibit large variance. To enable a tractable analysis, we approximate the latent distribution of each class c by a Gaussian characterized by its first- and second-order statistics:

$$p(z \mid c) \approx \mathcal{N}(\mu_c, \Sigma_c) \tag{12}$$

where μ_c and Σ_c denote the class mean and covariance. Importantly, our method does not require the true latent distribution to be exactly Gaussian. This assumption is used to quantify estimation uncertainty and motivate variance guided distribution modeling.

Given n i.i.d. samples $\{z_i\}_{i=1}^n \sim p(z \mid c)$, the empirical mean and covariance are

$$\hat{\mu}_c = \frac{1}{n}\sum_{i=1}^{n} z_i, \quad \hat{\Sigma}_c = \frac{1}{n-1}\sum_{i=1}^{n}(z_i - \hat{\mu}_c)(z_i - \hat{\mu}_c)^\top \tag{13}$$

The empirical mean is unbiased and satisfies:

$$\mathrm{Cov}[\hat{\mu}_c] = \frac{1}{n}\Sigma_c \tag{14}$$

which implies that small n leads to high uncertainty in class prototypes and inaccurate approximation of the true class-conditional distribution.

Under the Gaussian approximation, the empirical mean further admits a direct concentration bound:

$$\|\hat{\mu}_c - \mu_c\|_2 \leq \mathcal{O}\left(\sqrt{\frac{\mathrm{Tr}(\Sigma_c)\log(1/\delta)}{n}}\right) \tag{15}$$

with probability at least $1-\delta$. Therefore, the estimation error decreases as n increases and becomes smaller when the intra-class variance (e.g., $\mathrm{Tr}(\Sigma_c)$) is controlled. This variance-guided view motivates our objective in Section 3.2, which explicitly contracts intra-class dispersion and stabilizes category-level latent representations.

To quantify inter-class separability relative to class variability, we define a normalized distance:

$$\Delta_{ij} = \frac{\|\hat{\mu}_i - \hat{\mu}_j\|_2}{\sqrt{\mathrm{Tr}(\hat{\Sigma}_i) + \mathrm{Tr}(\hat{\Sigma}_j)}} \tag{16}$$

A larger Δ_{ij} indicates that two empirical class distributions are well-separated compared to their intra-class variability, reducing potential overlap in the latent space and improving category identifiability.

By explicitly controlling intra-class variance and promoting inter-class separation, the empirical latent distributions can reliably approximate the true class-conditional distributions:

$$\hat{\mu}_c \approx \mu_c, \quad \hat{\Sigma}_c \approx \Sigma_c, \quad \forall c \tag{17}$$

This ensures that the learned latent representations are statistically aligned with the underlying class distributions, allowing the generator to produce class-consistent samples with controlled diversity even under few-shot conditions. Complete proofs of the concentration bound and related results are provided in the Supplementary Materials (Appendix A).

Our proposed objective (Section 3.4) enforces variance contraction and distributional alignment across tasks, thereby stabilizing latent representations under limited samples. This variance-guided perspective provides a principled justification for learning reliable class-conditional distributions and achieving stronger generalization in few-shot generation.

3.4 Loss Functions and Training Procedure

As mentioned above, we insert a learnable latent adapter between the pre-trained encoder and generator, mapping instance-level codes to a shared category latent space without finetuning the generator. This adapter serves as a distributional bridge, aligning individual latent embeddings with their class distributions while preserving category structure.

To optimize the latent adapter, we formulate a multi-objective loss grounded in statistical principles. A category aware loss $\mathcal{L}_{class}$ encourages latent embeddings to follow compact class conditional distributions, minimizing intra-class variance. At the same time, it enlarges the expected separation between distributions of different classes. From a statistical perspective, this shapes the latent space such that within-class scatter is reduced and between-class scatter is increased, yielding well-separated category-level distributions.

A perceptual loss $\mathcal{L}_{vgg}$ constrains the second-order statistics of generated images with respect to the input, ensuring that fine-grained structures and textures are preserved. Meanwhile, an adversarial loss $\mathcal{L}_{adv}$ calibrates the generated distribution against the empirical data distribution, enforcing realism and semantic consistency. During training, both the encoder and generator are kept fixed, and we jointly optimize LA and a discriminator D, which evaluates the fidelity and class alignment of $\hat{x}_i^c = G(LA(pSp(x_i^c)))$. This driving LA to learn statistically coherent mappings that capture the category latent distributions.

The overall training objective is:

$$\begin{aligned}\min_{\theta_{LA}} \mathcal{L}_{LA} &= \mathcal{L}_{class} + \lambda_{vgg}\mathcal{L}_{vgg} - \lambda_{adv}\mathcal{L}_{adv} \\ &= \mathcal{L}_{class} + \lambda_{vgg}\frac{1}{NK_S}\sum_{c=1}^{N}\sum_{i=1}^{K_S}\big\|F\left(x_i^c\right) \\ &\quad - F\left(\hat{x}_i^c\right)\big\|_2 - \lambda_{adv}\frac{1}{NK_S}\sum_{c=1}^{N}\sum_{i=1}^{K_S}\left[D\left(\hat{x}_i^c\right)\right]\end{aligned} \tag{18}$$

$$\begin{aligned}\min_{\theta_D} \mathcal{L}_D &= \mathcal{L}_{adv} - \frac{1}{NK_S}\sum_{c=1}^{N}\sum_{i=1}^{K_S}\left[D\left(x_i^c\right)\right] + \lambda_{gp}\mathcal{L}_{gp} \\ &= \frac{1}{NK_S}\sum_{c=1}^{N}\sum_{i=1}^{K_S}\left[D\left(\hat{x}_i^c\right)\right] - \frac{1}{NK_S}\sum_{c=1}^{N}\sum_{i=1}^{K_S} \\ &\quad \left[D\left(x_i^c\right)\right] + \lambda_{gp}\frac{1}{NK_S}\sum_{c=1}^{N}\sum_{i=1}^{K_S}\left\|\nabla_{x_i^c} D\left(x_i^c\right)\right\|_2\end{aligned} \tag{19}$$

$F(\cdot)$ denotes the VGG feature extractor. λ_{gp} is hyper-parameters about gradient regularization. λ_{vgg}, λ_{adv} are loss weights for perceptual consistency and adversarial training.

3.5 Class Aware Latent Manipulation

After obtaining class-level latent representations, we further validate their statistical coherence by exploring controllable semantic directions. From a distributional perspective, stable class latent structures should permit localized perturbations that preserve global category consistency. If the learned latent space indeed captures class distributions, semantic variations preserve intra-class coherence and inter-class separation.

To this end, we explore controllable semantic directions to refine editing. However, global modeling often suffers from background interference, reducing controllability. Therefore, we propose a region-constrained CLIP-guided discovery method that uses CLIP's cross-modal alignment and a region-aware mechanism to optimize directions in target areas.

Specifically, we design an automated region mask generation mechanism to constrain the spatial scope of semantic direction discovery. Using SAM, synthesized image $\hat{x}_j^c$ are segmented to automatically select structurally clear and semantically relevant regions (e.g., flowers, faces) as binary masks. To ensure semantic accuracy, we first retain a set of candidate masks about regions with high prediction confidence:

$$\mathcal{M} = \{M_1, M_2, ..., M_K\}, M_k \in \{0,1\}^{H\times W} \tag{20}$$

We then apply heuristic rules to select the final mask. Assuming target object is the largest instance in image, we choose the region with the largest pixel count to obtain the binary mask for direction discovery.

$$M^* = \arg\max_{M_k\in\mathcal{M}} \sum_{i,j} M_k(i,j) \tag{21}$$

Next, to discover the latent direction Δz^* aligned with a given text prompt t such as *"a flower with red petals"*, we perform CLIP-guided direction optimization within the adapted latent space, constrained by the region mask M^*. The optimization objective is:

$$\mathcal{L}_{\text{mask-CLIP}} = -\cos\left(f_{\text{I}}\left(G(\hat{z}_j^c + \Delta z)\odot M^*\right), f_{\text{T}}(t)\right) \tag{22}$$

Here $f_I(\cdot)$ denotes CLIP image encoder, $\odot$ is element-wise multiplication, and $f_T(t)$ is text embedding. This optimization produces a region focused semantic direction that localizes edits and improves precision.

$$\Delta z^* = \arg\min_{\Delta z} \mathcal{L}_{\text{mask-CLIP}} \tag{23}$$

Finally, we perform linear interpolation along the discovered direction Δz^* on the category-level latent code $\hat{z}_j^c$, generating diverse images $\hat{x}^c$ with semantic variations while preserving consistent category structure.

$$\hat{x}^c = G\left(\hat{z}_j^c + \alpha_\tau \Delta z^*\right), \quad \alpha_\tau \in [a,b], \quad \tau = 1,\dots,T \tag{24}$$

Here, α_τ is the interpolation coefficient controlling semantic change, with step size sampled from a predefined range (e.g., $[-1,\ 1]$). As Δz^* is learned under region masks, the semantic variation is localized within target regions while preserving background consistency, enabling precise and localized editing.

This region guided strategy ensures that latent perturbations induce low intra-class variance while preserving inter-class separation, thus validating the statistical consistency of class-level distributions and enabling diverse yet coherent few shot generation.

4 EXPERIMENTS

4.1 Experiments Settings

Datasets. We conduct experiments on 102Flowers (Liu et al., 2019) , Animal Faces (Nilsback and Zisserman, 2008) and VGGFace (Parkhi et al., 2015). Following the settings in (Hong et al., 2022a), each dataset is split into training and test classes, as summarized in Table 2. All test classes are unseen during training, enabling evaluation of the model's generalization and generation capability on novel categories.

Implementation details. We first pretrain the *pSp* encoder and StyleGAN2 generator on seen categories and then freeze them in all subsequent stages. The Latent Adapter is a 4-layer multi-layer perceptron with an output dimension of 18×512, matching the latent codes produced by *pSp*. A category-aware metric model is also pretrained on seen classes to provide supervision, guiding the adapter to align instance codes with class-level structure. To compatible with the few-shot learning, we adopt an episodic training strategy, where each episode contains 4 classes with 1-shot support samples. We use 1-shot as the primary setting to evaluate whether the proposed distribution modeling can remain stable under extremely limited supervision. The adapter is optimized with Adam (lr=0.001, batch size=4) for 20,000 steps using episodes sampled from the training set. For generalization evaluation, we apply the trained model to unseen classes. All experiments are conducted on an NVIDIA GeForce RTX 4090 GPU. More details can be found in the supplementary. Our code is available at CAE.

Metric. We evaluate our methods in terms of image quality, diversity, and semantic consistency. For each unseen class, 128 images are generated from a single real image to compute FID (Heusel et al., 2017) and LPIPS (Zhang et al., 2018) scores, following (Gu et al., 2021; Hong et al., 2022a, 2020a). More details are provided in the supplementary materials. Additionally, we report classification accuracy to quantify semantic alignment with target class labels.

Table 1: The FID and LPIPS scores of images generated by different methods for unseen categories on three benchmark datasets.

Method	Fine-tune GAN	Extra Modules	102Flowers		Animal Faces		VGGFace	
			FID ↓	LPIPS ↑	FID ↓	LPIPS ↑	FID ↓	LPIPS ↑
FreezeD†	✓	×	52.92	-	78.88	-	-	-
DeltaGAN†	✓	✓	109.78	0.3912	89.81	0.4418	80.12	0.3146
Disco-FUNIT†	✓	✓	90.12	0.4436	71.44	0.4511	-	-
LoFGAN	✓	✓	96.38	0.3912	101.32	0.5016	**19.13**	0.3115
AGE	✓	✓	64.47	0.6462	57.64	0.7042	22.58	0.5668
SAGE	✓	✓	75.96	0.527	66.19	0.5063	90.14	0.3338
HAE	✓	✓	85.32	0.6007	69.86	0.5591	35.65	0.4463
Ours	×	✓(LA)	**52.89**	**0.6825**	**29.98**	**0.7306**	30.43	**0.5955**

Note. Fine-tune GAN: Indicates whether fine-tuning of the pre-trained generator is required (✓ for required, × for not required). Extra Modules: Indicates whether additional modules are introduced. † Results reported from original paper.

Table 2: Category splits for Animal Faces, 102Flowers, and VGGFace.

Dataset	Training	Testing	Total
Animal Faces	119	30	149
102Flowers	85	17	102
VGGFace	1802	572	2374

Table 3: Top-1/Top-5 classification accuracy (in %) of generated images using pretrained classifiers.

Method	102Flowers	Animal	VGG
Model acc	96.83/100.00	59.69/89.51	67.57/81.69
LoFGAN	63.69/95.40	33.31/69.11	30.82/55.58
AGE	59.79/94.16	42.71/77.89	26.69/50.67
SAGE	51.42/92.97	44.51/82.55	31.48/57.08
HAE	66.91/95.82	**69.33**/86.23	37.56/62.80
Ours	**79.09**/**98.99**	60.60/**91.85**	**52.75**/**74.81**

Note. We use ResNet50 (He et al., 2016) for VGGFace, and MobileNetV2 (Sandler et al., 2018) for 102Flowers and Animal Faces. Classifiers are trained on real test classes and evaluated on generated images from unseen categories to assess semantic consistency.

4.2 Few shot image generation

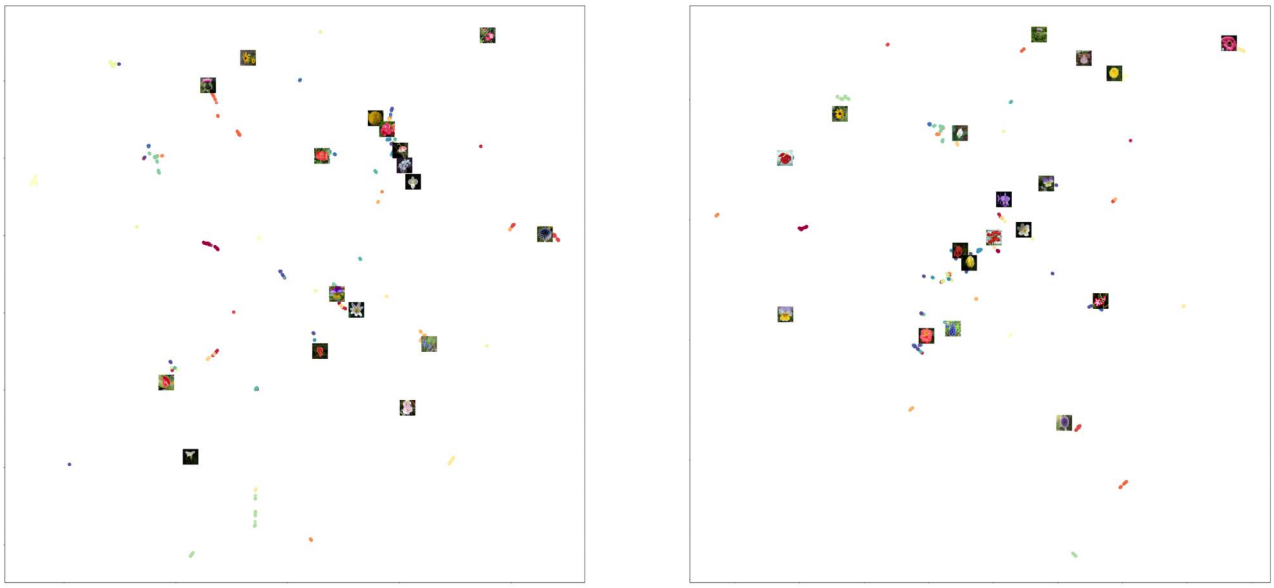
Baselines. We compare our method with representative few-shot image generation approaches. FreezeD (Mo et al., 2020) is included to validate our ability to generate high-quality images by adding other module. DeltaGAN, Disco-FUNIT, LoFGAN, AGE, SAGE, and HAE represent mainstream techniques that improve image diversity and category consistency under limited data. Comprehensive comparisons across multiple metrics demonstrate the effectiveness and advantages of our approach.

Main Results. We evaluate all baselines on 102Flowers, Animal Faces, and VGGFaces using FID, LPIPS, and classification accuracy. For fair comparison, we follow the same dataset split as Table 2. Results (†) is taken directly from their paper, while the others are obtained using our implementation under the same setting. As shown in Table 1 and Table 3, our method consistently achieves superior performance across all datasets, with the best or competitive scores in FID and test accuracy, indicating strong capability in preserving semantic alignment while maintaining high image quality. Meanwhile, our LPIPS scores remain high, demonstrating diverse generation under semantic constraints. Unlike most baselines that require fine-tuning the generator, our approach introduces only a lightweight latent adapter and keeps the GAN frozen, significantly reducing training cost. Overall, our method achieves a better balance between quality, diversity, and semantic control, showing strong generalization and practical efficiency.

Effect of Latent Adapter. The $\mathcal{L}_{class}$ is designed to enforce intra-class compactness and inter-class separation in the latent space. Without $\mathcal{L}_{class}$ degenerates into instance-level mapping, which is equivalent to removing LA. We investigate the impact of the latent adapter on class level latent structure and diversity. As shown in Table 4, introducing the adapter consistently improves per-class FID across all datasets, with a notable 11.43% reduction on Animal Faces, indicating enhanced category consistency and semantic controllability. The LPIPS score also increases, reflecting improved intra-class diversity. In contrast, the adapter enables more diverse outputs in style and pose while maintaining class semantics. These results empirically confirm that the latent adapter guides instance level representations toward a shared class level latent distribution, enhancing both controllability and diversity.

Table 4: Effect of the *Latent Adapter* (LA) on generation quality and diversity. We report FID (↓) and LPIPS (↑) scores for unseen categories on three datasets.

Method	LA	102Flowers		Animal Faces		VGGFace	
		FID ↓	LPIPS ↑	FID ↓	LPIPS ↑	FID ↓	LPIPS ↑
Ours	×	59.08	0.6165	33.85	0.6482	32.14	0.4518
Ours	✓	**52.89**	**0.6825**	**29.98**	**0.7306**	**30.43**	**0.5955**



(a) No-var latent (102F) (b) Var latent (102F)

Figure 3: Visualization of latent code distributions in Euclidean space for the 102Flowers dataset: (a) baseline, and (b) our proposed method.

Qualitative Evaluation. Figure 3 visualizes the latent codes projected in Euclidean space on 102Flowers. The baselines (a) yield latent spaces where samples from the same class are loosely grouped and often overlap with other classes. This indicates insufficient intra-class compactness and poor class-level separation. In contrast, our method (b) produces latent embeddings that form tight, well-defined clusters within each class and exhibit clear margins between categories. Such variance control enhances both the reliability and interpretability of the latent space across all three datasets. This clustering demonstrates that the latent adapter aggregates instance level codes toward a shared class level latent distribution, capturing the underlying category structure. The inter-class separation further confirms that different categories occupy distinct regions in the latent space, reducing semantic ambiguity and supporting robust classification. These results provide functional evidence that the learned latent space supports class consistent, disentangled semantic directions and generalizes from few examples to robust class level representations. Additional results on the other two datasets are provided in the supplementary materials.

5 Conclusion

In this work, we present a statistical distribution driven approach for few shot image generation that learns class level latent distributions from limited samples. By modeling shared latent representations, our method captures category structure and enables sampling of diverse yet semantically consistent examples. We provide theoretical insights linking intra-class variance control and inter-class separation to generalization, and demonstrate superior performance over existing methods in image quality, diversity, and category consistency. These results highlight the effectiveness of class level latent disentanglement and manipulation under scarce data, with future work extending to multimodal settings for enhanced generalization.

References

Abdal, R., Qin, Y., and Wonka, P. (2019). Image2stylegan: How to embed images into the style-

gan latent space? In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4432–4441.

Abdal, R., Zhu, P., Mitra, N. J., and Wonka, P. (2021). Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. *ACM Transactions on Graphics (ToG)*, 40(3):1–21.

Clouâtre, L. and Demers, M. (2019). Figr: Few-shot image generation with reptile.

Couairon, G., Verbeek, J., Schwenk, H., and Cord, M. (2023). Diffedit: Diffusion-based semantic image editing with mask guidance. In *ICLR 2023 (Eleventh International Conference on Learning Representations)*.

Ding, G., Han, X., Wang, S., Jin, X., and Huang, Q. (2025). Stable attribute group editing for reliable few-shot image generation. *IEEE Transactions on Circuits and Systems for Video Technology*.

Ding, G., Han, X., Wang, S., Wu, S., Jin, X., Tu, D., and Huang, Q. (2022). Attribute group editing for reliable few-shot image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11194–11203.

Fort, S. (2017). Gaussian prototypical networks for few-shot learning on omniglot. *arXiv preprint arXiv:1708.02735*.

Gal, R., Patashnik, O., Maron, H., Bermano, A. H., Chechik, G., and Cohen-Or, D. (2022). Stylegannada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13.

Goetschalckx, L., Andonian, A., Oliva, A., and Isola, P. (2019). Ganalyze: Toward visual definitions of cognitive image properties. In *Proceedings of the ieee/cvf international conference on computer vision*, pages 5744–5753.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.

Gu, Z., Li, W., Huo, J., Wang, L., and Gao, Y. (2021). Lofgan: Fusing local representations for few-shot image generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8463–8471.

Guo, Y., Du, R., Li, X., Xie, J., Ma, Z., and Dong, Y. (2022). Learning calibrated class centers for few-shot classification by pair-wise similarity. *IEEE Transactions on Image Processing*, 31:4543–4555.

Härkönen, E., Hertzmann, A., Lehtinen, J., and Paris, S. (2020). Ganspace: Discovering interpretable gan controls. *Advances in neural information processing systems*, 33:9841–9850.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.

Hong, Y., Niu, L., Zhang, J., and Zhang, L. (2020a). Matchinggan: Matching-based few-shot image generation. In *2020 IEEE International conference on multimedia and expo (ICME)*, pages 1–6. IEEE.

Hong, Y., Niu, L., Zhang, J., and Zhang, L. (2022a). Deltagan: Towards diverse few-shot image generation with sample-specific delta. In *European conference on computer vision*, pages 259–276. Springer.

Hong, Y., Niu, L., Zhang, J., and Zhang, L. (2022b). Few-shot image generation using discrete content representation. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 2796–2804.

Hong, Y., Niu, L., Zhang, J., Zhao, W., Fu, C., and Zhang, L. (2020b). F2gan: Fusing-and-filling gan for few-shot image generation. In *Proceedings of the 28th ACM international conference on multimedia*, pages 2535–2543.

Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119.

Kawar, B., Zada, S., Lang, O., Tov, O., Chang, H., Dekel, T., Mosseri, I., and Irani, M. (2023). Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6007–6017.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg,

A. C., Lo, W.-Y., et al. (2023). Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026.

Lake, B., Salakhutdinov, R., Gross, J., and Tenenbaum, J. (2011). One shot learning of simple visual concepts. In *Proceedings of the annual meeting of the cognitive science society*, volume 33.

Li, L., Zhang, Y., and Wang, S. (2023). The euclidean space is evil: Hyperbolic attribute editing for few-shot image generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22714–22724.

Liang, W., Liu, Z., and Liu, C. (2020). Dawson: A domain adaptive few shot generation framework.

Ling, H., Kreis, K., Li, D., Kim, S. W., Torralba, A., and Fidler, S. (2021). Editgan: High-precision semantic image editing. *Advances in Neural Information Processing Systems*, 34:16331–16345.

Liu, M.-Y., Huang, X., Mallya, A., Karras, T., Aila, T., Lehtinen, J., and Kautz, J. (2019). Few-shot unsupervised image-to-image translation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10551–10560.

Mo, S., Cho, M., and Shin, J. (2020). Freeze the discriminator: a simple baseline for fine-tuning gans.

Nichol, A., Achiam, J., and Schulman, J. (2018). On first-order meta-learning algorithms.

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

Park, Y.-H., Kwon, M., Jo, J., and Uh, Y. (2023). Unsupervised discovery of semantic latent directions in diffusion models.

Parkhi, O., Vedaldi, A., and Zisserman, A. (2015). Deep face recognition. In *BMVC 2015-Proceedings of the British Machine Vision Conference 2015*. British Machine Vision Association.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., and Cohen-Or, D. (2021). Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2287–2296.

Roich, D., Mokady, R., Bermano, A. H., and Cohen-Or, D. (2022). Pivotal tuning for latent-based editing of real images. *ACM Transactions on graphics (TOG)*, 42(1):1–13.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520.

Shen, Y., Yang, C., Tang, X., and Zhou, B. (2020). Interfacegan: Interpreting the disentangled face representation learned by gans. *IEEE transactions on pattern analysis and machine intelligence*, 44(4):2004–2018.

Shen, Y. and Zhou, B. (2021). Closed-form factorization of latent semantics in gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1532–1540.

Vershynin, R. (2018). *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press.

Wu, Z., Lischinski, D., and Shechtman, E. (2021). Stylespace analysis: Disentangled controls for stylegan image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12863–12872.

Xia, W., Yang, Y., Xue, J.-H., and Wu, B. (2021). Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2256–2265.

Yang, C., Shen, Y., and Zhou, B. (2021). Semantic hierarchy emerges in deep generative representations for scene synthesis. *International Journal of Computer Vision*, 129(5):1451–1466.

Yann, L. (2010). Mnist handwritten digit database. *ATT Labs*.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018). 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.

Zhu, J., Feng, R., Shen, Y., Zhao, D., Zha, Z.-J., Zhou, J., and Chen, Q. (2021). Low-rank subspaces in gans. *Advances in Neural Information Processing Systems*, 34:16648–16658.

Zhu, J., Shen, Y., Xu, Y., Zhao, D., and Chen, Q. (2022). Region-based semantic factorization in gans. In *International Conference on Machine Learning*, pages 27612–27632. PMLR.

Checklist

The checklist follows the references. For each question, choose your answer from the three possible options: Yes, No, Not Applicable. You are encouraged to include a justification to your answer, either by referencing the appropriate section of your paper or providing a brief inline description (1-2 sentences). Please do not modify the questions. Note that the Checklist section does not count towards the page limit. Not including the checklist in the first submission won't result in desk rejection, although in such case we will ask you to upload it during the author response period and include it in camera ready (if accepted).

In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes/No/Not Applicable]
   Answer: Yes
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes/No/Not Applicable]
   Answer: Not Applicable
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes/No/Not Applicable]
   Answer: No. We have not submitted the source code and corresponding libraries at this stage. Upon acceptance of the paper, we will release our implementation to the public.
2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes/No/Not Applicable]
   Answer: Yes
   (b) Complete proofs of all theoretical results. [Yes/No/Not Applicable]
   Answer: Yes
   (c) Clear explanations of any assumptions. [Yes/No/Not Applicable]
   Answer: Yes
3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes/No/Not Applicable]
   Answer: Yes. To ensure the reproducibility of our results, we include detailed pseudocode in the supplementary material.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes/No/Not Applicable]
   Answer: Yes. The details can be seen at Section4.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes/No/Not Applicable]
   Answer: Yes
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes/No/Not Applicable]
   Answer: Yes
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes/No/Not Applicable]
   Answer: Yes
   (b) The license information of the assets, if applicable. [Yes/No/Not Applicable]
   Answer: Yes
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes/No/Not Applicable]
   Answer: Not Applicable
   (d) Information about consent from data providers/curators. [Yes/No/Not Applicable]
   Answer: Yes
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Yes/No/Not Applicable]
   Answer: Not Applicable
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Yes/No/Not Applicable]
   Answer: Not Applicable
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Yes/No/Not Applicable]
   Answer: Not Applicable

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Yes/No/Not Applicable]

Answer: Not Applicable

Supplementary Materials

A Proofs for Section 3.3 Variance Guided Generalization

A.1 Notation and Assumptions

We consider a fixed class c and assume latent codes are i.i.d. samples:

$$z_1, \dots, z_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mu_c, \Sigma_c) \tag{25}$$

where $\Sigma_c \geq 0$ and $\mathrm{Tr}(\Sigma_c) < \infty$. Define the empirical mean

$$\hat{\mu}_c = \frac{1}{n} \sum_{i=1}^{n} z_i \tag{26}$$

A.2 Lemma 1: Unbiasedness and Covariance of the Sample Mean

Lemma 1. The sample mean $\hat{\mu}_c$ satisfies

$$\mathbb{E}[\hat{\mu}_c] = \mu_c, \quad \mathrm{Cov}[\hat{\mu}_c] = \frac{1}{n}\Sigma_c \tag{27}$$

Proof. By linearity of expectation:

$$\mathbb{E}[\hat{\mu}_c] = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[z_i] = \frac{1}{n} \cdot n\mu_c = \mu_c \tag{28}$$

For the covariance, independence gives

$$\mathrm{Cov}[\hat{\mu}_c] = \mathrm{Cov}\Big(\frac{1}{n} \sum_{i=1}^{n} z_i\Big) = \frac{1}{n^2} \sum_{i=1}^{n} \mathrm{Cov}[z_i] = \frac{1}{n^2} \cdot n\Sigma_c = \frac{1}{n}\Sigma_c \tag{29}$$

□

A.3 Lemma 2: Gaussian Concentration of the Sample Mean

Lemma 2. For any $\delta \in (0, 1)$, with probability at least $1 - \delta$:

$$\|\hat{\mu}_c - \mu_c\|_2 \leq \sqrt{\frac{2\,\mathrm{Tr}(\Sigma_c)\log(1/\delta)}{n}} \tag{30}$$

Proof. Since $z_i \sim \mathcal{N}(\mu_c, \Sigma_c)$ are i.i.d.,

$$\hat{\mu}_c \sim \mathcal{N}\Big(\mu_c, \frac{1}{n}\Sigma_c\Big) \tag{31}$$

hence $\hat{\mu}_c - \mu_c \sim \mathcal{N}(0, \Sigma_c/n)$. The expected squared norm is

$$\mathbb{E}\big[\|\hat{\mu}_c - \mu_c\|_2^2\big] = \mathrm{Tr}\Big(\frac{1}{n}\Sigma_c\Big) = \frac{1}{n}\mathrm{Tr}(\Sigma_c) \tag{32}$$

By a standard Gaussian concentration inequality for the Euclidean norm of a Gaussian vector (Vershynin, 2018), with probability at least $1 - \delta$:

$$\|\hat{\mu}_c - \mu_c\|_2 \leq \sqrt{\frac{2\,\mathrm{Tr}(\Sigma_c)\log(1/\delta)}{n}} \tag{33}$$

□

A.4 Proposition 1: Variance-Dependent Estimation Error

Proposition 1. Under the same assumptions, the estimation error of the class prototype satisfies

$$\|\hat{\mu}_c - \mu_c\|_2 = \mathcal{O}\Big(\sqrt{\frac{\mathrm{Tr}(\Sigma_c)\log(1/\delta)}{n}}\Big) \quad \text{with probability at least } 1-\delta \tag{34}$$

Therefore, reducing intra-class variance (i.e., $\mathrm{Tr}(\Sigma_c)$) improves the stability of class-conditional latent estimation in few-shot regimes.

Proof. This follows directly from Lemma 2. □

A.5 Probabilistic Guarantee for Inter-Class Separation

Define a normalized inter-class distance between classes i and j:

$$\Delta_{ij} = \frac{\|\hat{\mu}_i - \hat{\mu}_j\|_2}{\sqrt{\mathrm{Tr}(\hat{\Sigma}_i) + \mathrm{Tr}(\hat{\Sigma}_j)}} \tag{35}$$

By Lemma 2, with probability at least $1-\delta$:

$$\|\hat{\mu}_c - \mu_c\|_2 \le \sqrt{\frac{2\,\mathrm{Tr}(\Sigma_c)\log(1/\delta)}{n}}, \quad c \in \{i, j\} \tag{36}$$

Applying the triangle inequality yields

$$\|\hat{\mu}_i - \hat{\mu}_j\|_2 \ge \|\mu_i - \mu_j\|_2 - \sqrt{\frac{2\,\mathrm{Tr}(\Sigma_i)\log(1/\delta)}{n}} - \sqrt{\frac{2\,\mathrm{Tr}(\Sigma_j)\log(1/\delta)}{n}} \tag{37}$$

holding with probability at least $1-2\delta$.

Thus, if true class means are well-separated relative to intra-class variances, the empirical means remain separated with high probability, ensuring Δ_{ij} is large and latent categories are identifiable under few-shot conditions.

B RELATED WORK

Few-shot image generation aims to generate images with semantic consistency and diversity from few samples of unseen categories. This task imposes demands on a model's generalization capability, intra-class structural preservation, and semantic modeling. Existing methods can be broadly classified into four categories. Meta-learning approaches, such as FIGR (Clouâtre and Demers, 2019) and DAWSON (Liang et al., 2020), improve generalization to new classes through cross-task training. FIGR transfers meta-parameters across tasks using the Reptile algorithm (Nichol et al., 2018), while DAWSON incorporates domain adaptation for fast adjustment to the target class. Although these methods enhance generalization, they often neglect structural information, leading to unstable image quality, and are mainly applied to simpler datasets like MNIST (Yann, 2010) and Omniglot (Lake et al., 2011). Fusion modeling methods improve representation by fusing features or completing images. F2GAN (Hong et al., 2020b) combines feature fusion with region filling, while MatchingGAN (Hong et al., 2020a) uses feature matching to improve image consistency. LoFGAN (Gu et al., 2021) integrates local features for better detail expression. While these methods improve quality, the fusion processes may introduce structural artifacts or semantic shifts, compromising image authenticity. Transformation transfer methods model category transitions by capturing transformation increments. DeltaGAN (Hong et al., 2022a) allows more flexible category transformation, while (Hong et al., 2022b) improves image diversity by learning discrete content representations. However, these methods lack semantic interpretability and structural consistency. Latent space editing approaches including Attribute Group Editing (AGE) (Ding et al., 2022) and its stable version SAGE (Ding et al., 2025) improve semantic control by grouping attributes. HAE (Li et al., 2023) shifts attribute space from Euclidean to hyperbolic space for better alignment of few-shot sample distributions. While excelling in semantic modeling, these methods often rely on manually defined attribute directions and lack structural awareness, which can lead to inconsistency or deformation in the generated images.

C THE FRAMEWORK OF OUR METHOD

Our framework for few-shot image generation adopts a statistical distribution driven approach. The overall framework is shown as Figure4. Given a single input image x_i^c, we first encode it using a pretrained encoder and map it through the latent adapter to obtain a class-level latent distribution $P(z|c)$. This distribution captures the underlying category structure, allowing us to sample multiple latent codes that represent plausible variations within the same class. Each sampled code $z^{'}$ is then refined via a CLIP-guided optimization, constrained by region masks generated from the reconstructed images, to align semantic attributes with a target text prompt. Finally, we perform linear interpolation along the optimized directions to generate diverse yet class-consistent images $\hat{x}_i^c$. This pipeline enables controlled, high-fidelity synthesis from limited examples while preserving both intra-class diversity and semantic coherence.

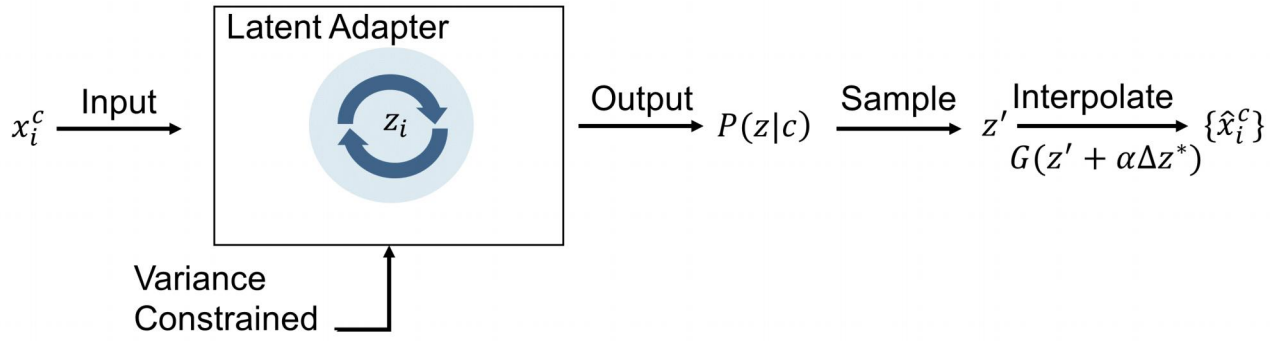


Figure 4: The overall framework of our method.

D THE TRAINING AND SAMPLING PROCEDURE

Algorithm 1 describes the training procedure of our method. In this stage, instance-level latent representations are extracted using a pre-trained encoder and generator, and then mapped to a shared class-level latent space through a lightweight Latent Adapter. During training, a category-aware loss guides the adapter to aggregate instance codes toward the corresponding class distribution, while perceptual and adversarial losses preserve image quality and detail, enabling the learning of class-level latent distributions that generalize from few examples.

Algorithm 2 describes how new samples are generated based on the learned class-level latent distributions. It is noted that the model does not compute covariance directly but relies on the learnable latent adapter to infer class distribution parameters implicitly. In Algorithm 2, where LA is written as producing $(\mu_c, \sum_c)$, was only a conceptual shorthand to express that LA implicitly learns a class latent distribution during training. It only refines the notation to better reflect the behavior of the existing LA module. Latent codes are first sampled from the estimated class distributions and then optimized along semantic directions under region-constrained CLIP guidance. Linear interpolation along these directions produces diverse, high-quality images that remain consistent with the original class. This procedure demonstrates the effectiveness of the learned class-level distributions and achieves controllable few-shot image generation.

E IMPLEMENTATION DETAILS AND ANALYSIS

In our framework, the *pSp* encoder is pretrained only on the training split following the standard few-shot setting, ensuring that the encoder never sees any test-time classes. The encoder is trained to invert real images into the StyleGAN2 latent space via reconstruction losses contained pixel and perceptual, following the standard pSp training protocol in (Ding et al., 2022). We keep the architecture unchanged and do not finetune it on novel classes. Its purpose is to provide a stable mapping to the StyleGAN latent space so that our method can focus on class-level distribution learning rather than encoder optimization.

During the training process, the hyper-parameters in Eq.(18) and (19) are set as $\lambda_{vgg} = 0.05$, $\lambda_{adv} = 0.1$, and $\lambda_{gp} = 5$. To prevent the LA from introducing unpredictable disruptions to the latent space, we adopt a staged

Algorithm 1 Training procedure of the proposed method

Require: Pre-trained encoder pSp, generator G, metric model $\mathcal{M}$, training episodes $\mathcal{S}$, learning rates η_{LA}, η_D, loss weights $\lambda_{vgg}, \lambda_{adv}$
Ensure: Trained latent adapter LA and discriminator D
Initialize LA, D; freeze pSp and G
for each episode sampled from $\mathcal{S}$ **do**
  Encode support images: $w_i^c = pSp(x_i^c)$
  for $t = 1$ to T **do**
    Map to shared class space: $z_i^c = LA(w_i^c)$
    Reconstruct images: $\hat{x}_i^c = G(z_i^c)$
    Compute class-consistency loss:

$$\mathcal{L}_{class} = \frac{1}{NK_S} \sum_{c,i} \left(\mathcal{M}(S, \hat{x}_i^c) - l_i^c \right)^2$$

    Compute perceptual and adversarial losses: $\mathcal{L}_{vgg}$, $\mathcal{L}_{adv}$
    Update adapter: $\theta_{LA} \leftarrow \theta_{LA} - \eta_{LA} \nabla_{\theta_{LA}} (\mathcal{L}_{class} + \lambda_{vgg} \mathcal{L}_{vgg} + \lambda_{adv} \mathcal{L}_{adv})$
  end for
  Update discriminator: $\theta_D \leftarrow \theta_D - \eta_D \nabla_{\theta_D} (\mathcal{L}_{adv} - \frac{1}{NK_S} \sum D(x_i^c) + \lambda_{gp} \mathcal{L}_{gp})$
end for
return LA, D

training strategy. In the initial stage, the module is trained without $\mathcal{L}_{class}$ to reproduce the original w latent codes, effectively learning an identity mapping. This ensures that the adapter starts from a stable initialization aligned with the pre-trained pSp encoder. In the second stage, all loss terms are activated to fully optimize the adapter, guiding the adapter to explicitly model class latent distributions by variance guided objectives. This allows align instance level latents with class conditional distributions while preserving semantic consistency and diversity.

F DETAILS OF METRICS

FID. The Fréchet Inception Distance (FID) (Heusel et al., 2017) assesses the distribution similarity between generated and real images. Specifically, both synthetic and real images are typically fed through an inception network pretrained on ImageNet, and the distance between their feature distributions is computed using the embeddings extracted from the final layer. A lower FID score indicates that the generated samples are closer to the real samples in the feature space, suggesting higher generation quality and better distribution alignment.

LPIPS. The Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al., 2018) measures perceptual distance between two images using deep neural network features. Lower LPIPS indicates higher similarity, while higher values reflect larger visual differences. In image generation, LPIPS is commonly used to quantify intra-class diversity, as perceptual distance correlates with variation of visual attributes within the same category.

G ADDITIONAL EXPERIMENTS

G.1 Main Results

Qualitative Evaluation. To further validate the generative capability, we present qualitative comparisons across multiple datasets and semantic attributes. As shown in Figure 5, we compare our method with representative few-shot generation approaches including LoFGAN, AGE, and HAE under the same category and semantic conditions. Our approach demonstrates stable semantic consistency while producing richer variations in style, structure, and pose. In contrast, LoFGAN exhibits instability in certain categories with attribute drift and mode collapse; AGE generates clear images but lacks diversity in semantic edits across samples; HAE improves training stability but struggles with fine-grained semantic control. Figure 6 illustrates our method's generation results on 102Flowers and Animal Faces datasets along semantic directions such as quantity and pose. The results show that our method effectively controls semantic variations while maintaining category consistency, yielding natural and diverse outputs. These qualitative findings further confirm the superior semantic control, diversity,

Algorithm 2 Few-Shot Image Generation from Input via Class-Level Distribution

Require: Input image x, trained latent adapter LA, generator G, text prompt t, number of samples K, number of interpolation steps T, CLIP model f_I, f_T, learning rate η
Ensure: Generated images $\{\hat{x}^c_{k,\tau}\}$

Encode input: $w = pSp(x)$
Map to class-level latent space: $\mu_c, \Sigma_c = LA(w)$
for $k = 1$ to K **do**
    Sample latent code: $\hat{z}^c_k \sim \mathcal{N}(\mu_c, \Sigma_c)$
    Initialize latent direction: $\Delta z \leftarrow 0$
    Initial reconstruction: $\hat{x}^c_k = G(\hat{z}^c_k)$
    Generate region mask: M^* using SAM on $\hat{x}^c_k$
    for CLIP optimization steps **do**
        Compute masked CLIP loss:

$$\mathcal{L}_{\text{mask-CLIP}} = -\cos(f_I(\hat{x}^c_k \odot M^*), f_T(t))$$

        Update latent direction: $\Delta z \leftarrow \Delta z - \eta \nabla_{\Delta z} \mathcal{L}_{\text{mask-CLIP}}$
        Update reconstruction: $\hat{x}^c_k = G(\hat{z}^c_k + \Delta z)$
    end for
    for $\tau = 1$ to T **do**
        Interpolate along optimized direction:

$$\hat{x}^c_{k,\tau} = G(\hat{z}^c_k + \alpha_\tau \Delta z), \quad \alpha_\tau \in [a, b]$$

    end for
end for
return $\{\hat{x}^c_{k,\tau}\}$

and category consistency of our approach.

Figure 7 and 8 further visualize the latent code in Euclidean space across Animal Faces and VGGFace datasets. Our method produces more clustered and class-discriminative representations, providing functional evidence that the learned latent space captures class-level distributions and disentangled semantic directions. the latent embeddings of partial samples from Animal Faces and VGGFace datasets in Euclidean space, with the color of the points representing the class. The overlaid images are for visual aids; the focus should be on the clustering structure. Compared to baseline methods, our method produces clearer inter-class separation. Figure 7 and 8(b) both show greater distances between different classes compared to the baseline, indicating improved inter-class separation. This demonstrates that the latent space captures discriminative, class-consistent representations that generalize effectively from few samples.

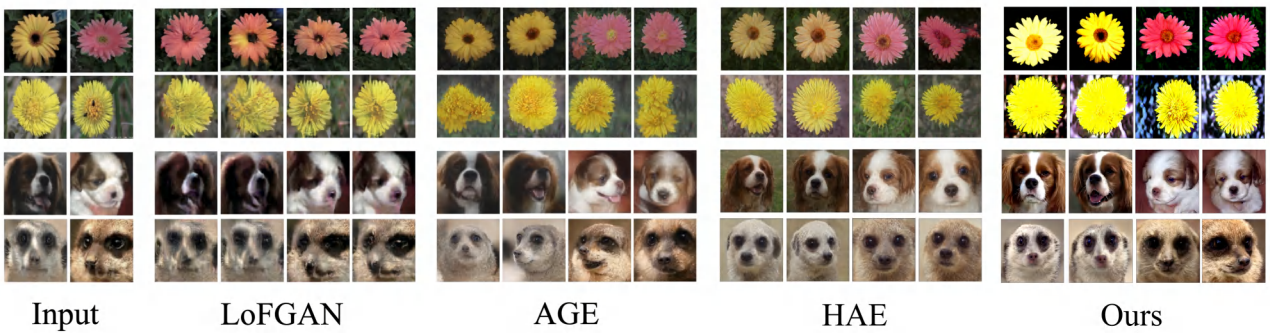


Figure 5: Generated images by our method compared with LoFGAN, AGE and HAE. Note that: SAGE has not released code and pre-trained models.

Latent Space Interpolation. We conducted linear interpolation experiments in the latent space to evaluate

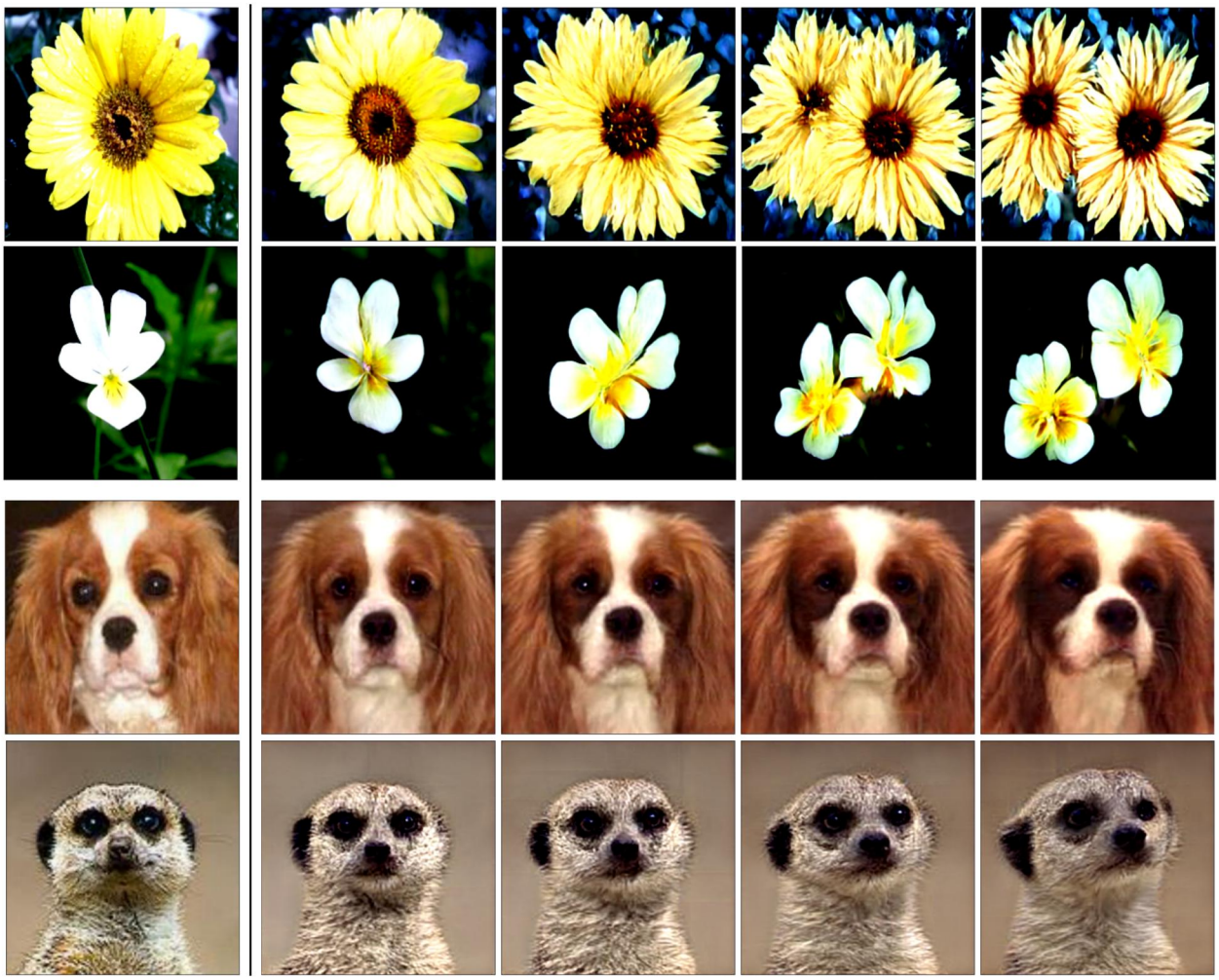


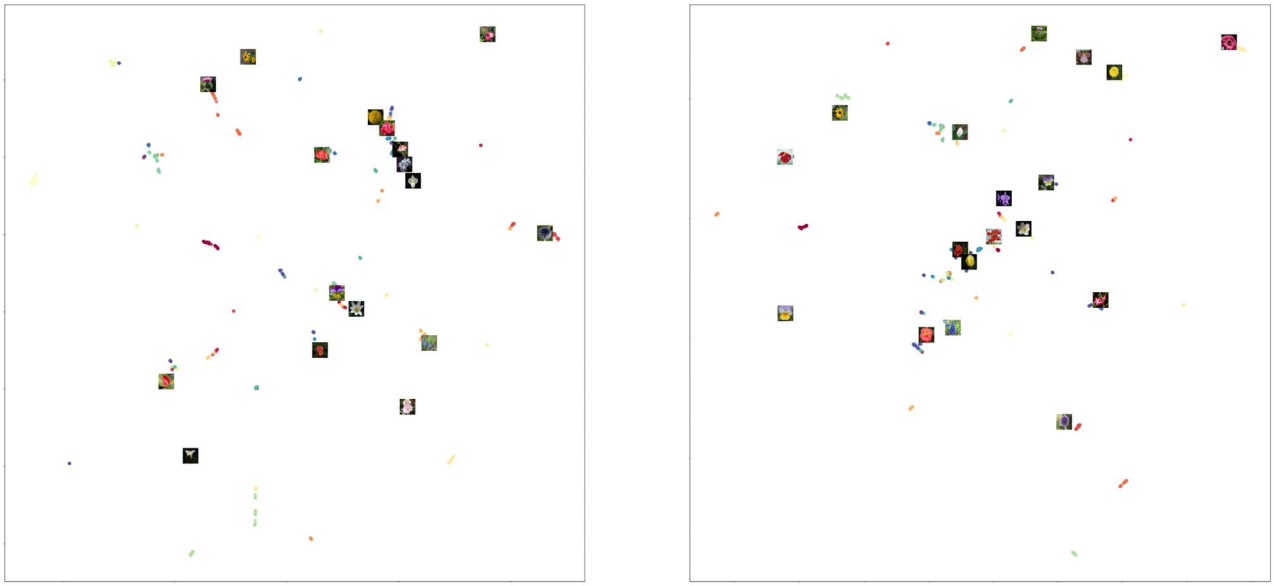
Figure 6: Examples of semantic-controlled image generation on 102Flowers and Animal Faces.

the consistency and stability of the discovered semantic directions cross samples. By sampling from the same class and interpolating along the shared semantic direction, this experiment reveals the continuity and generality of semantic directions in the latent space. Figure 9 illustrates the semantic evolution along the interpolation paths generated by our method. The results show smooth and natural transitions of target attributes with increasing interpolation steps, demonstrating strong semantic consistency. Meanwhile, non-target attributes such as background and pose remain stable, highlighting the local focus and disentanglement capability of the semantic directions. This confirms the semantic directions possess clear directionality, continuity, and controllability. It effectively supports structured latent space representation and manipulation under few-shot settings, providing a solid foundation for subsequent editing and generation tasks.

G.2 Ablation Study

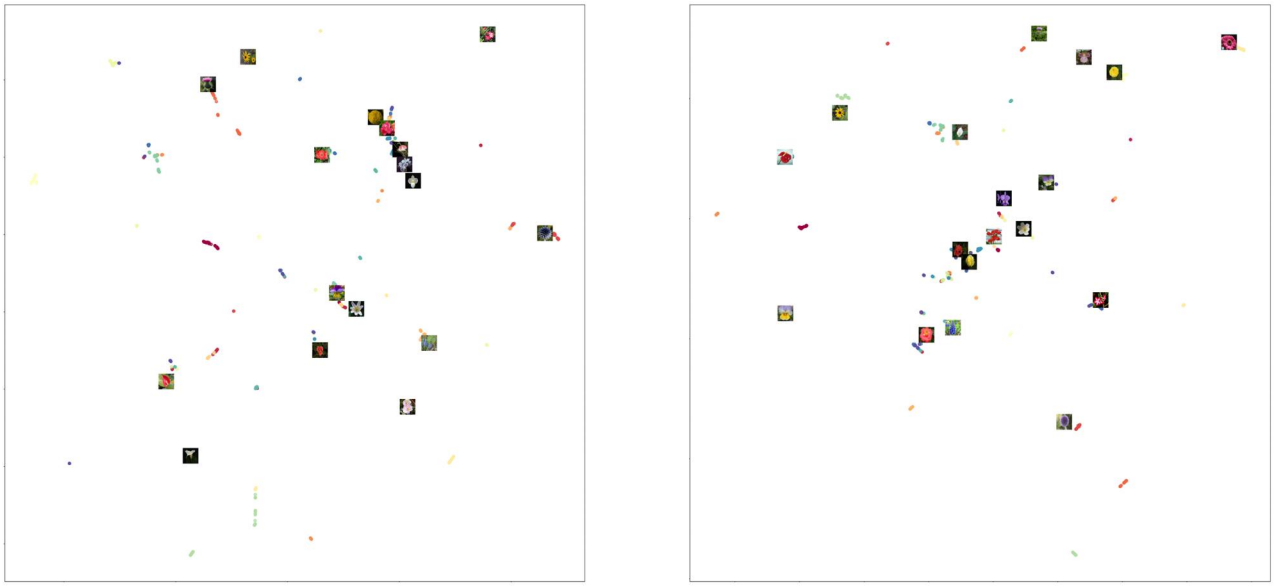
Effect of Latent Adapter. We investigate the impact of the latent adapter on class-level semantic disentanglement and generation diversity. Figure 10 presents qualitative comparisons under few-shot settings. Without the adapter, outputs show limited variation and repetitive attributes. In contrast, the adapter enables more diverse outputs in style and pose while maintaining class semantics. These results demonstrate that the latent adapter refines the latent space, guiding representations toward a shared class-level direction and improving both diversity and category consistency.

Effect of Region Mask in Semantic Direction Discovery. We further explore the role of region masks



(a) No-var latent (102F) (b) Var latent (102F)

Figure 7: Visualization of latent code distributions in Euclidean space for the Animal Faces dataset: (a) baseline, and (b) our proposed method.



(a) No-var latent (102F) (b) Var latent (102F)

Figure 8: Visualization of latent code distributions in Euclidean space for the VGGFace dataset: (a) baseline, and (b) our proposed method.

in guiding semantic direction discovery. Specifically, we compare results with and without applying flower or face region masks during CLIP-guided optimization. As shown in Figure 11, region-constrained directions yield more focused and stable edits for target attributes (e.g., petal color, expression), while reducing interference from irrelevant areas such as background. In contrast, directions discovered without masks often result in vague or entangled changes. These results demonstrate that region masks enhance the precision and interpretability of semantic control by focusing optimization on task-relevant areas, improving fine-grained editing under few-shot settings.

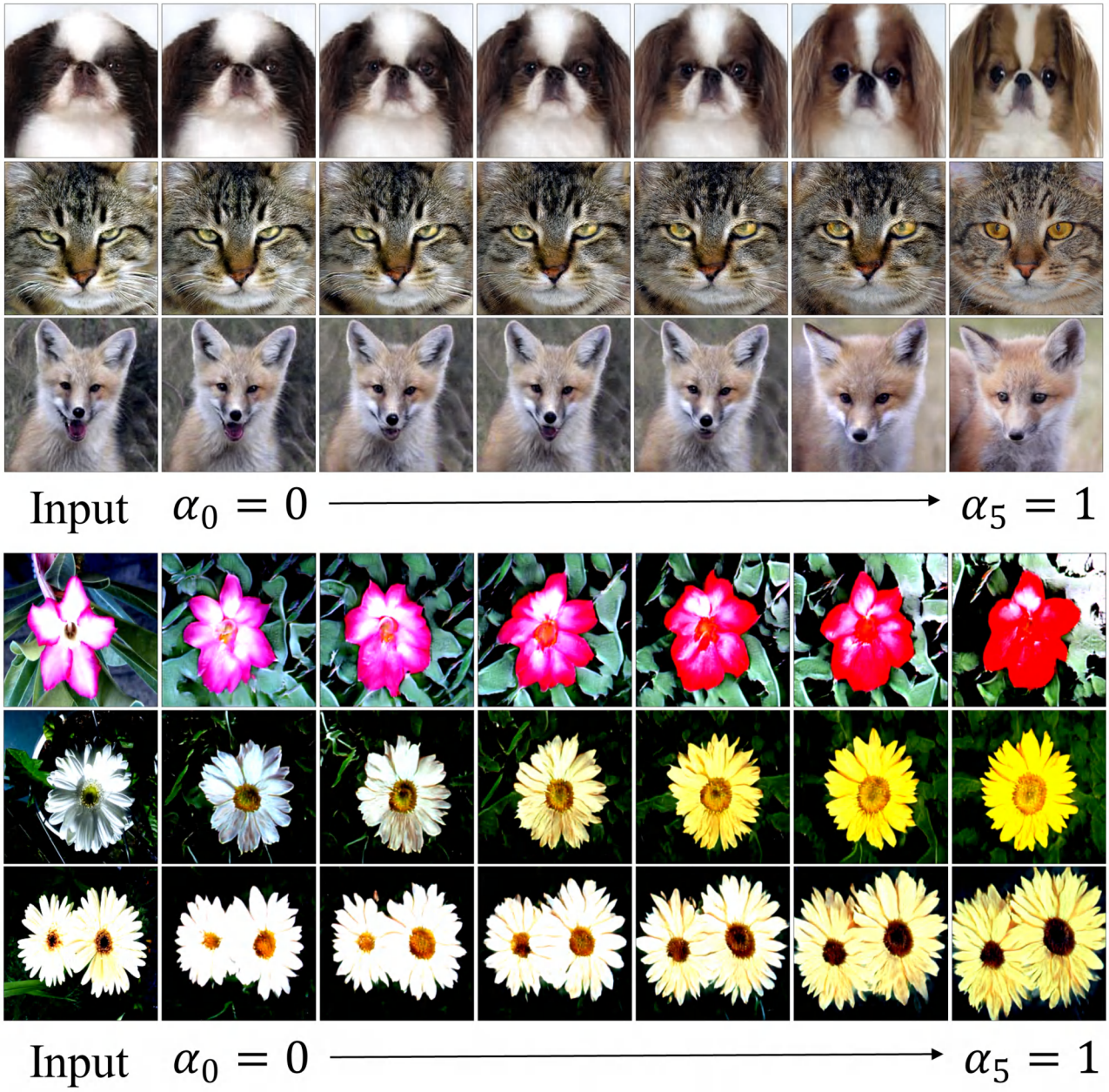


Figure 9: Latent interpolation along discovered semantic direction

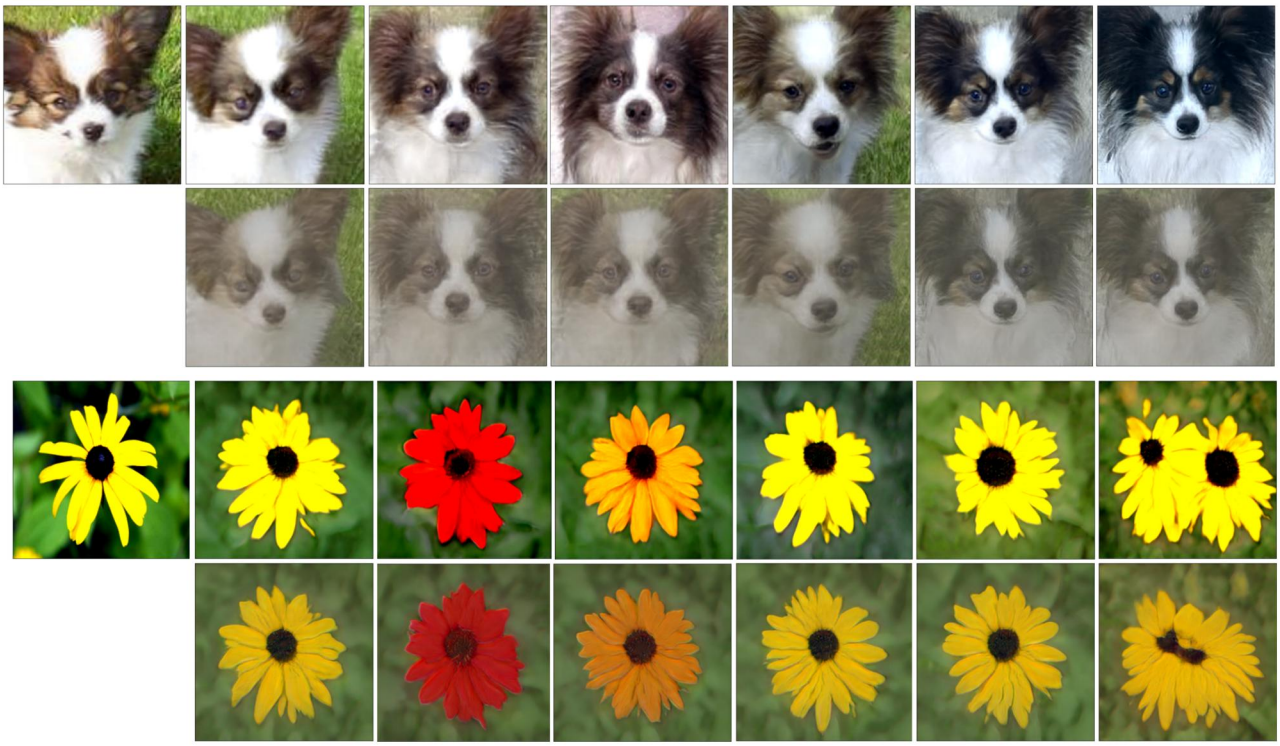


Figure 10: Qualitative comparison of few-shot generation with (Top) and without (Bottom) Latent Adapter.

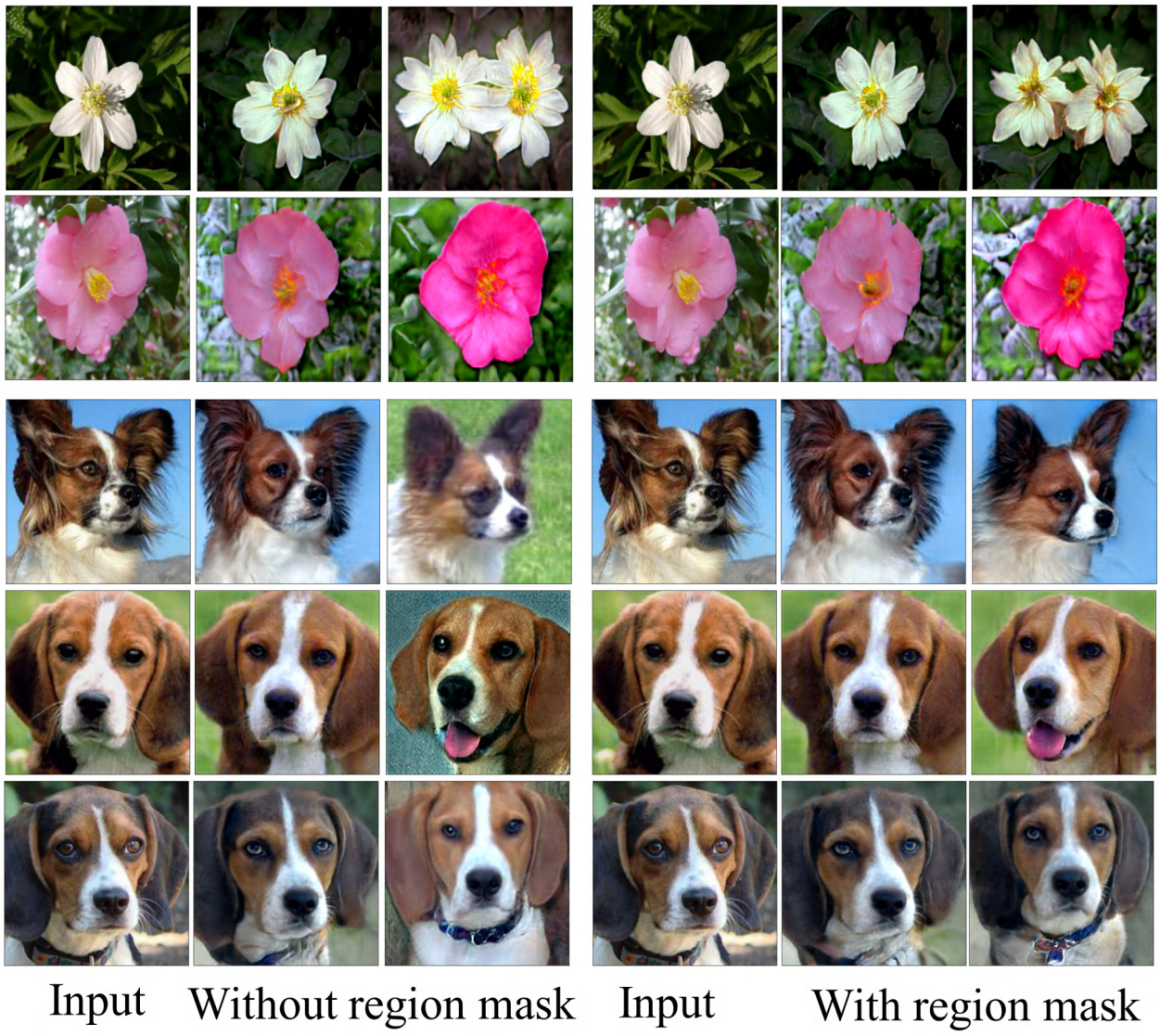


Figure 11: Comparison of semantic edits with and without region masks.