LMF: LEARNING MINORITY FACTORS FOR DECOMPOSABLE MINORITY GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Minority generation synthesizes high-fidelity, uncommon samples to bolster synthetic data utility and advance creative AI. While current methods can successfully steer generation trajectories toward low-density regions, they remain restricted to pattern-agnostic minority guidance and fail to uncover the mechanisms underlying minority formation. To remedy this limitation, we reconceptualize minority samples from a novel factor-level perspective and introduce **Decomposable Minority Generation**, a new paradigm that discovers minority factors within a data distribution and leverages them for synthesis. We instantiate this paradigm with **LMF** (**L**earning **M**inority **F**actors), comprising a minority factor pool, a detail-aware modulator, and a factor routing module, together with anti-collapse regularization and Siamese factor disentanglement. Experiments show that the learned factors exhibit coherent semantics, cross-class reusability, continuous controllability, and compositionality, supporting pattern-specified minority generation and interpretation of minority characteristics. Quantitatively, LMF achieves more faithful minority distribution matching than prior methods across all benchmarks, reducing cFID and sFID by up to 30.2% and 34.2%, respectively. Furthermore, we demonstrate the utility of LMF generated minority samples beyond generation through improved downstream visuo-motor control.

1 INTRODUCTION

Existing text-to-image (T2I) models Rombach et al. (2022); Saharia et al. (2022) can achieve remarkable visual fidelity, but their generations often remain biased toward high-density regions of the data distribution, as frequent patterns are more readily captured during training Park et al. (2025). Consequently, low-density yet plausible modes are underrepresented, creating a gap between the generated and real distributions. To bridge this gap, minority generation Um & Ye (2024; 2025) aims to synthesize rare yet feasible samples, with benefits for synthetic data utility Ayromlou et al. (2025); Belagali et al. (2026) and creative AI Han et al. (2025); Golan et al. (2026).

Prior efforts in minority generation characterize minority samples by their reconstruction instability under noise perturbation, and steer generation trajectories toward low-density regions through latent-space manipulation Park & Um (2026); Um et al. (2024); Um & Ye (2024), condition-space modulation Um & Ye (2025); Wang et al. (2026), and sampling-protocol modification Um et al. (2025). Beyond reaching minority regions, understanding what makes a sample minority is valuable for both interpretability and targeted synthesis. However, existing methods remain *pattern-agnostic*. They can determine *whether* a sample lies in a minority region, while leaving *what* makes an individual sample minority intrinsically opaque. As a result, the mechanisms underlying minority formation remain unexplored, rendering the targeted synthesis of specific minority patterns unattainable.

These limitations motivate us to reconceptualize minority samples from a novel factor-level perspective. We posit that a minority sample can be interpreted as the common mode augmented with a composition of minority factors. Here, the common mode denotes the shared visual patterns within the high-density region of the same semantic class, while each factor captures a recurring deviation from this mode, such as handheld presentation, printed depiction, or circular arrangement. This formulation motivates a novel paradigm that performs minority generation in a decomposable manner, coined **Decomposable Minority Generation**. Instead of relying on pattern-agnostic guidance during generation, this paradigm requires the model to first uncover the minority factors underlying a

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

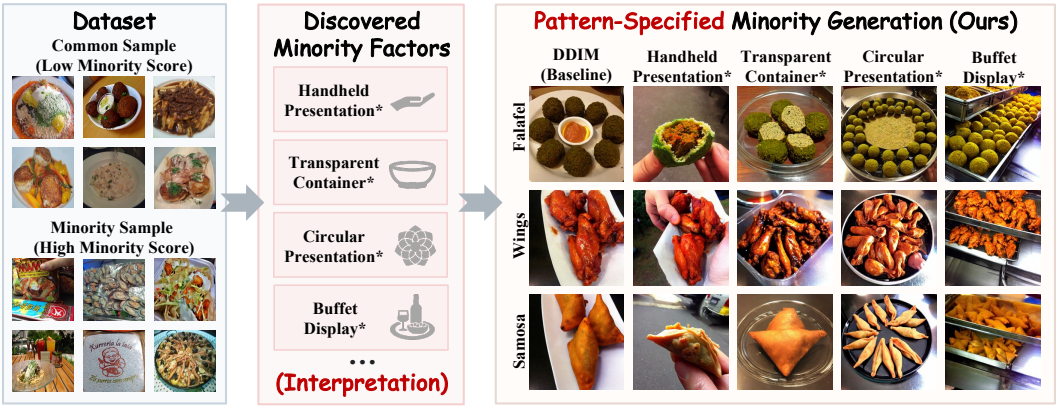


Figure 1: Illustration of decomposable minority generation. The discovered minority factors provide a factor-level interpretation of minority formation and enable pattern-specified minority generation. The shown samples are generated by individually activating minority factors learned by LMF on Foods dataset. **Factor labels are assigned post hoc by GPT-5.6 Sol. This notation is used consistently throughout all figures.*

data distribution and subsequently leverage them for synthesis, as shown in Fig. 1. Consequently, the discovered factors not only provide insight into minority formation, but also enable pattern-specified minority generation through explicit factor specification.

To instantiate this paradigm, we propose **L**earning **M**inority **F**actors (**LMF**), a pool-based framework that encompasses three components: a minority factor pool, a detail-aware modulator, and a factor routing module. Firstly, we construct a minority factor pool dedicated to learning and storing individual minority factors. To determine which minority factors are present in a given sample, we introduce a detail-aware modulator that retains the fine-grained visual details crucial for constituting minority features. The resulting representations are subsequently processed by the factor routing module to establish associations between the sample and its relevant minority factors, alongside an anti-collapse regularization to promote broader participation of factor entries. Eventually, we devise a Siamese factor disentanglement regularization to encourage different factor entries to capture distinct minority characteristics, enhancing the representational capacity of the minority factor pool.

In summary, our main contributions are as follows: (1) We devise a novel minority generation paradigm, termed decomposable minority generation, which uncovers minority factors within a distribution to enable pattern-specified minority generation and provide factor-level insights into minority formation. (2) We instantiate this paradigm with LMF (Learning Minority Factors), comprising a minority factor pool, a detail-aware modulator, and a factor routing module, together with anti-collapse regularization and Siamese factor disentanglement for distinct factor learning. (3) Comprehensive experiments demonstrate that the factors learned by LMF exhibit high semantic coherence, reusability, continuity, and compositionality. Quantitative evaluations show that LMF more faithfully models target minority distributions across multiple benchmarks, while the generated minority samples further improve downstream visuo-motor control.

2 Related Work

2.1 Minority Generation

Minority generation aims to guide the generation trajectory toward low-density regions of the data distribution Sehwag et al. (2022); Yu et al. (2020); Um et al. (2024). Current methods execute this through three interventions. Latent-space manipulation methods revise intermediate diffusion states using minority-related gradients derived from an auxiliary classifier Um et al. (2024), reconstruction-based likelihood estimation Um & Ye (2024), or a semantic density prior Park & Um (2026). Condition-space modulation methods encode minority information into the conditioning signal. MP Um & Ye (2025) dynamically optimizes an additional minority-aware token embedding during generation, while DCDM Wang et al. (2026) represents sample difficulty using a set of auxiliary conditioning tokens. Sampling-protocol modification methods broaden the initial noise

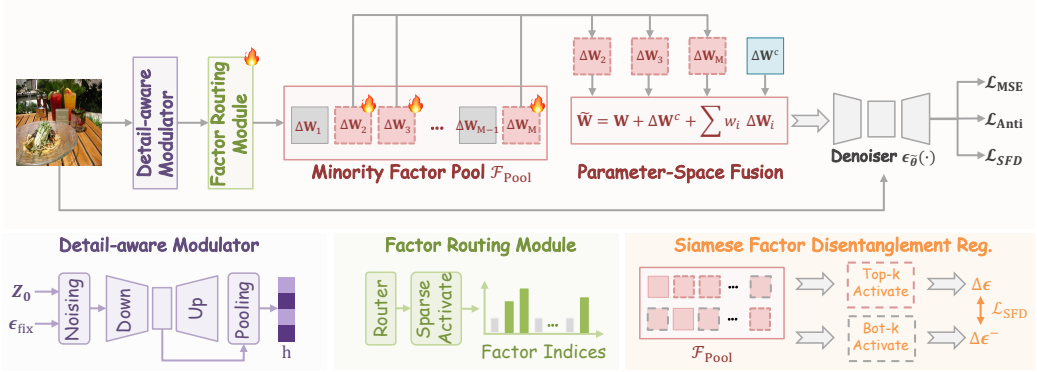


Figure 2: Overview of Learning Minority Factors (LMF) for decomposable minority generation. The minority factor pool stores the learned factors, while the detail-aware modulator and factor routing module determine the factor composition associated with each image. Training is jointly supervised by the MSE, anti-collapse, and Siamese factor disentanglement objectives.

distribution and reshape the denoising schedule to bias generation toward minority regions Um et al. (2025). However, existing methods are agnostic to the heterogeneous structure of minority regions, obscuring the mechanisms driving minority formation and precluding the targeted synthesis of specific minority patterns. To bridge this gap, we introduce a novel factor-level conceptualization of minority samples and a corresponding decomposable minority generation paradigm that uncovers their underlying minority factors, enabling pattern-specified minority generation.

2.2 Generative Concept Learning

Decomposable minority generation requires minority-specific visual information to be represented by individually identifiable and reusable components. This objective aligns closely with generative concept learning Avrahami et al. (2023); Kumari et al. (2023); Wei et al. (2023), which extracts semantically meaningful concepts from limited visual references using pretrained T2I models. Initial attempts distill a holistic object concept from reference images, represented either as a textual token Gal et al. (2023) or through model adaptation Ruiz et al. (2023). Subsequent works extend this paradigm to multiple concepts Vinker et al. (2023); Hao et al. (2024); Zhang et al. (2024), or organize visual concepts along predefined semantic axes such as category, color, and material Lee et al. (2024b); Xu et al. (2025). More recent approaches further integrate object-level concepts with intrinsic attributes to provide a more comprehensive decomposition of visual content Cendra & Han (2025); Shi et al. (2026). Albeit these advances, existing methods are detached from the global data distribution, providing no mechanism to discern minority-specific deviations from common characteristics. By contrast, our method identifies recurring minority factors across the target distribution and explicitly models them as deviations from the common mode.

3 Learning Minority Factors

To endow minority generation with pattern specification and interpretability, we propose **LMF** (**L**earning **M**inority **F**actors), which instantiates decomposable minority generation paradigm. Built upon a pretrained latent diffusion model, LMF consists of a minority factor pool, a detail-aware modulator, and a factor routing module, complemented by anti-collapse regularization and Siamese factor disentanglement, as illustrated in Fig. 2. We first introduce the necessary preliminaries, then formulate minority samples from a factor-level perspective, and subsequently detail the proposed LMF framework.

3.1 Preliminaries

We build our method on latent diffusion models, which operate in the latent space of a pretrained VAE with encoder $\mathcal{E}$ and decoder $\mathcal{D}$. Given an image $\mathbf{x}$, its clean latent is obtained as $\mathbf{z}_0 = \mathcal{E}(\mathbf{x})$.

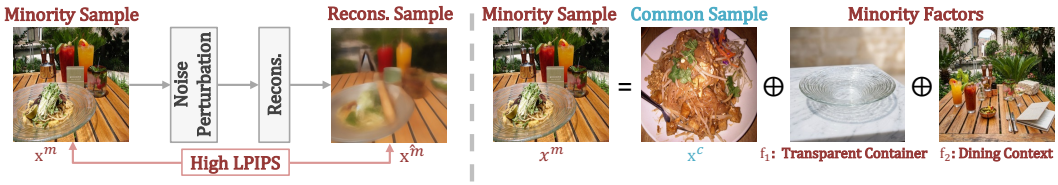


Figure 3: Comparison of prior and factor-level minority formulations. Prior formulations identify minority samples based on reconstruction discrepancy, while our formulation extends this characterization to the factor level by decomposing each minority sample into its constituent minority factors.

The forward diffusion process perturbs $\mathbf{z}_0$ at timestep t as:

$$\mathbf{z}_t = \sqrt{\bar{\alpha}_t}\,\mathbf{z}_0 + \sqrt{1-\bar{\alpha}_t}\,\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{1}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$ is determined by a predefined noise schedule. The denoiser $\boldsymbol{\epsilon}_\theta(\cdot)$, typically parameterized by a U-Net, is trained to predict the injected noise $\boldsymbol{\epsilon}$ from $(\mathbf{z}_t, t, \mathbf{c})$, where $\mathbf{c}$ denotes the text condition. The standard training objective is the mean squared error loss:

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

3.2 Factor-level Minority Formulation

Prior works characterize minority samples based on the observation that minority-specific features are susceptible to being removed during noise perturbation and reconstruction Um & Ye (2025; 2024); Song et al. (2025), as illustrated in Fig. 3 (a). Accordingly, the minority score is measured by the discrepancy between the original image and its reconstructed counterparts:

$$s(\mathbf{x}) = \frac{1}{R} \sum_{r=1}^{R} d_{\text{LPIPS}} \left(\mathbf{x}, \hat{\mathbf{x}}^{(r)} \right), \tag{3}$$

where $\hat{\mathbf{x}}^{(r)}$ denotes the image reconstructed from the r-th noise perturbation and denoising process, and d_{LPIPS} measures the perceptual discrepancy between the original and reconstructed images Zhang et al. (2018). A larger $s(\mathbf{x})$ therefore indicates stronger minority characteristics.

While reconstruction-based estimation indicates *how minority* a sample is, it does not reveal *what makes* the sample minority. Hence, we extend the conventional minority formulation to a novel factor-level perspective, as shown in Fig. 3 (b), where a minority sample is interpreted as a common sample augmented with several minority factors:

$$\mathbf{x}^m \approx \mathbf{x}^c \oplus \{\mathbf{f}_{i_1}, \mathbf{f}_{i_2}, \ldots, \mathbf{f}_{i_K}\}, \tag{4}$$

where $\mathbf{x}^m$ denotes a minority sample, $\mathbf{x}^c$ denotes the common sample of the same semantic class, and each $\mathbf{f}_{i_k}$ represents a minority factor that characterizes a particular visual deviation from the common mode. The operator $\oplus$ denotes semantic composition rather than pixel-wise addition. This factor-level view makes minority formation explicitly decomposable, providing a structured basis for decomposable minority generation.

3.3 Minority Factor Pool

Building upon our factor-level formulation, we view individual minority factors as recurring patterns shared across diverse samples and classes within a distribution. This motivates us to uncover shared minority factors directly from the target distribution. To this end, we instantiate each factor $\mathbf{f}_i$ as an independent LoRA update $\Delta\mathbf{W}_i = \mathbf{B}_i\mathbf{A}_i$ and organize them into a minority factor pool:

$$\mathcal{F}_{\text{pool}} = \{\Delta\mathbf{W}_i = \mathbf{B}_i\mathbf{A}_i\}_{i=1}^{M}, \tag{5}$$

where M denotes the pool size, forming a reusable set of candidate minority factors. Since each minority sample exhibits a distinct factor composition, factor learning is performed through sample-specific compositions of the shared factor pool. Accordingly, for an arbitrary minority sample,

let $\mathcal{I} = \{i_1, \ldots, i_K\}$ denote its associated factors and $\mathbf{w}$ the corresponding association weights, obtained by the factor routing module described in Sec. 3.5. These factors are then composed with the pretrained denoiser at the parameter level:

$$\widetilde{\mathbf{W}} = \mathbf{W} + \Delta\mathbf{W}^c + \sum_{i\in\mathcal{I}} w_i \Delta\mathbf{W}_i, \tag{6}$$

where $\mathbf{W}$ and $\widetilde{\mathbf{W}}$ denote the pretrained and composed weights, respectively. Here, $\Delta\mathbf{W}^c$ is a common-mode LoRA update learned from common samples and kept fixed during minority factor learning. The inclusion of this module not only fulfills Eq. 4, but also shields the learned factors from common mode contamination, ensuring they capture purer minority-specific characteristics. Therefore, the resulting weight defines the denoising prediction:

$$\hat{\epsilon} = \epsilon_{\widetilde{\mathbf{W}}}(\mathbf{z}_t, t, \mathbf{c}). \tag{7}$$

The corresponding denoising loss follows the standard diffusion objective $\mathcal{L}_{\mathrm{MSE}}$ defined in Eq. 2.

3.4 Detail-aware Modulator

The detail-aware modulator aims to capture fine-grained visual cues that characterize minority patterns. Conventional discriminative representations primarily emphasize category-level semantics and may suppress subtle variations that are less relevant to recognition but important for minority formation, such as presentation style or spatial arrangement. We therefore adopt intermediate diffusion features, which preserve richer fine-grained visual information Tang et al. (2023); Song et al. (2025). Given $\mathbf{z}_0$, we apply fixed noise $\boldsymbol{\epsilon}_{\mathrm{fix}}$ at timestep t_{fix} following Eq. 1, feed the resulting $\mathbf{z}_{t_{\mathrm{fix}}}$ into the frozen denoiser $\boldsymbol{\epsilon}_{\theta_{\mathrm{DAM}}}$, and pool its layer-$l$ feature to obtain the detail-aware representation:

$$\mathbf{h} = \mathrm{Pool}\left(\Phi_l^{\theta_{\mathrm{DAM}}}\left(\mathbf{z}_{t_{\mathrm{fix}}}, t_{\mathrm{fix}}, \mathbf{c}\right)\right), \tag{8}$$

where $\Phi_l^{\theta_{\mathrm{DAM}}}$ denotes the intermediate feature extracted from layer l of the denoiser. The resulting $\mathbf{h}$ is subsequently used for minority factor routing.

3.5 Factor Routing Module

Given the detail-aware representation $\mathbf{h}$ from Sec. 3.4, the factor routing module establishes the correspondence between the input image and the minority factor pool. Specifically, we utilize a lightweight router to map $\mathbf{h}$ to routing weights over the factor pool:

$$\mathbf{w} = \mathrm{Sigmoid}\big(\mathrm{Linear}_2(\mathrm{SiLU}(\mathrm{Linear}_1(\mathbf{h})))\big), \tag{9}$$

where $\mathbf{w} \in (0,1)^M$ denotes the routing weights over M minority factors. The factors with the K largest routing weights are selected, yielding the factor set $\mathcal{I} = \mathrm{TopK}(\mathbf{w}, K)$. The selected factors and their corresponding weights are then used for the parameter composition in Eq. 6.

Anti-Collapse Regularization. A potential limitation in factor routing is the emergence of *dead factors*, wherein routing selections collapse onto a small subset of entries. Such underutilization compromises the expressiveness of the minority factor pool. To counteract this routing degeneracy, we introduce anti-collapse regularization to encourage broader participation of factor entries during training. For a mini-batch of B samples, we normalize the routing weights of each sample and aggregate them to estimate the batch-level factor usage:

$$p_{b,i} = \frac{w_{b,i}}{\sum_{j=1}^{M} w_{b,j}}, \qquad q_i = \frac{1}{B}\sum_{b=1}^{B} p_{b,i}, \tag{10}$$

where q_i denotes the usage of factor i within the mini-batch, and $\mathbf{q} = [q_1, \ldots, q_M]$ denotes the corresponding usage distribution. We encourage the batch-level usage distribution to maintain high entropy during the early stage of training, ensuring sufficient participation across the factor pool. Thus, we match $\mathbf{q}$ to a uniform reference distribution $\mathbf{u}$ to discourage premature factor collapse:

$$\mathcal{L}_{\mathrm{Anti}} = D_{\mathrm{KL}}\left(\mathbf{q} \,\|\, \mathbf{u}\right) = \sum_{i=1}^{M} q_i \log \frac{q_i}{1/M}, \tag{11}$$

where $\mathbf{u} = [1/M, \ldots, 1/M]$. Importantly, $\mathcal{L}_{\mathrm{Anti}}$ is applied only during early training to promote broad factor participation and is subsequently removed, allowing the pool to specialize according to the natural factor usage of the dataset rather than enforcing a uniform final distribution.

3.6 SIAMESE FACTOR DISENTANGLEMENT

To maximize the representational capacity of the minority factor pool, different entries should capture distinct minority characteristics. Consequently, we introduce Siamese factor disentanglement to promote non-redundant minority effects across factor entries. For each sample, the activated factor composition $\mathcal{I}$ contains the most relevant factor entries, while the Siamese counterpart is constructed from the Bottom-K factors, $\mathcal{I}^- = \mathrm{BottomK}(\mathbf{w}, K)$. Here, the Siamese routing scores $\mathbf{w}$ are used only for selection, and the selected factors are assigned unit weights to avoid overly weak Siamese effects. In order to isolate the visual effects induced by the two factor compositions, we compute their residual predictions relative to the shared common-mode denoiser $\boldsymbol{\epsilon}_{\theta_c}$:

$$\Delta\boldsymbol{\epsilon} = \boldsymbol{\epsilon}_{\widetilde{\mathbf{W}}}(\mathbf{z}_t, t, \mathbf{c}) - \boldsymbol{\epsilon}_{\mathbf{W}^c}(\mathbf{z}_t, t, \mathbf{c}), \quad \Delta\boldsymbol{\epsilon}^- = \boldsymbol{\epsilon}_{\widetilde{\mathbf{W}}^-}(\mathbf{z}_t, t, \mathbf{c}) - \boldsymbol{\epsilon}_{\mathbf{W}^c}(\mathbf{z}_t, t, \mathbf{c}), \tag{12}$$

where $\mathbf{W}^c = \mathbf{W} + \Delta\mathbf{W}^c$ denotes the common-mode weights, and $\widetilde{\mathbf{W}}^-$ denotes the denoiser parameters corresponding to the Siamese factor compositions. Then, we impose a margin on the similarity between the two residual directions to encourage these effects to be sufficiently distinct:

$$\mathcal{L}_{\mathrm{SFD}} = \max\left(0,\, \cos\left(\Delta\boldsymbol{\epsilon}, \Delta\boldsymbol{\epsilon}^-\right) - \gamma\right), \tag{13}$$

where γ represents the similarity margin.

3.7 TRAINING OBJECTIVES

The overall objective combines the denoising loss, anti-collapse regularization, and Siamese factor disentanglement:

$$\mathcal{L} = \begin{cases} \mathcal{L}_{\mathrm{MSE}} + \lambda_{\mathrm{Anti}}\mathcal{L}_{\mathrm{Anti}} + \lambda_{\mathrm{SFD}}\mathcal{L}_{\mathrm{SFD}}, & \rho < \rho_{\mathrm{Anti}}, \\ \mathcal{L}_{\mathrm{MSE}} + \lambda_{\mathrm{SFD}}\mathcal{L}_{\mathrm{SFD}}, & \rho \geq \rho_{\mathrm{Anti}}, \end{cases} \tag{14}$$

where $\rho \in [0, 1]$ denotes the normalized training progress and ρ_{Anti} specifies the fraction of training during which anti-collapse regularization is applied.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets and Split. We conduct our main experiments on ImageNet-1K Deng et al. (2009) and Foods Bossard et al. (2014). For each dataset, we compute the minority score defined in Eq. 3 and rank all samples accordingly. The top 50% samples with higher minority scores are used for minority factor learning, while the bottom 50% samples serve as common samples for learning the common-mode update $\Delta\mathbf{W}^c$. For ablation studies, we use ImageNet-10 Kim et al. (2022), which offers a more compact benchmark for analyzing individual design choices. Full results on ImageNet-10 are provided in the Appendix.

Evaluation Protocol and Metrics. We evaluate LMF at both the factor and distribution levels. At the factor-level evaluation, we qualitatively assess the learned factors through factor-specified generation. Activating individual factors across classes evaluates semantic coherence and cross-class reusability, varying factor weights examines continuity of control, and jointly activating multiple factors verifies compositionality. We further perform post-hoc factor interpretation with GPT-5.6 Sol to reveal the semantics of the discovered factors and provide insight into minority formation. These evaluations characterize capabilities enabled by factor-level modeling that are not explicitly supported by prior minority generation methods. At the distribution level evaluation, we quantify how faithfully LMF models the target minority distribution by generating a set of minority samples and measuring cFID Parmar et al. (2022) and sFID Nash et al. (2021) against a predefined minority reference set following the evaluation protocol of prior works Park & Um (2026); Um et al. (2025). We further report Precision and Recall Kynkäänniemi et al. (2019) for fidelity and coverage, together with CLIPScore Hessel et al. (2021) and PickScore Kirstain et al. (2023) for semantic consistency and perceptual quality. For ablation studies, we additionally report Density and Coverage Naeem et al. (2020) to provide a more comprehensive assessment of model behavior. All metrics are averaged over three independent runs. Standard deviations are reported in Tab. 1 and omitted from the remaining tables for clarity.

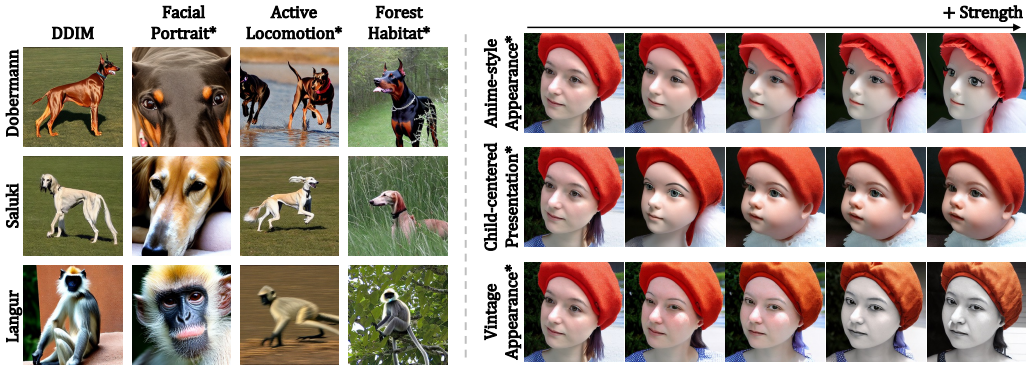


Figure 4: Single-factor-specified minority generation and factor-strength control on ImageNet-1K. The learned factors exhibit consistent semantics across classes while remaining distinct from one another. The arrows indicate progressively increasing factor LoRA weights, demonstrating continuous control over factor strength.

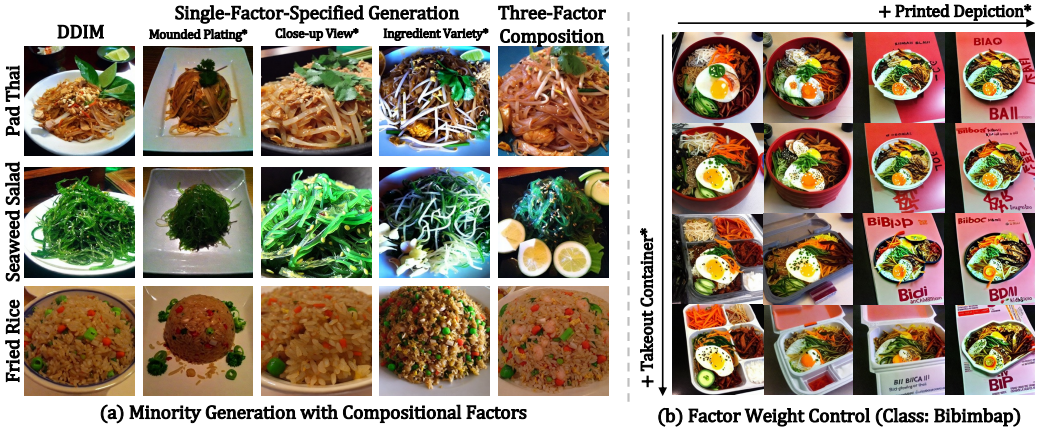


Figure 5: Multi-factor-specified minority generation and factor-strength control on Foods. In (a), the last column jointly activates the preceding three factors. In (b), the weights of the two factors are varied at fixed intervals of 0.25 along the two axes. These results demonstrate the compositionality of the learned factors, while preserving independent control over individual factor strengths.

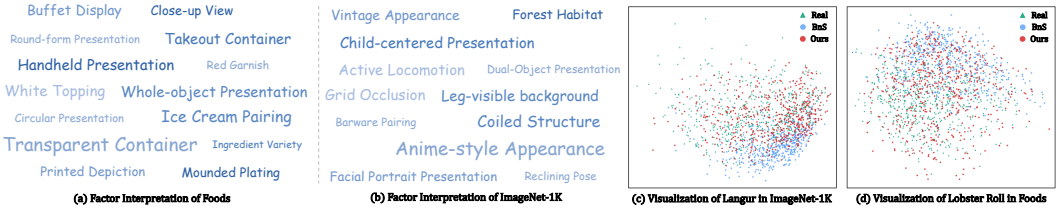
Implementation Details. We use SD v1.5 as the base T2I model. The factor pool contains 64 entries with Top-3 routing. We train for 100 epochs on ImageNet-10 and Foods and 10 epochs on ImageNet-1K, with a learning rate of 1×10^{-4}. We apply $\mathcal{L}_{\text{Anti}}$ during the first 30% of training and set $\lambda_{\text{Anti}} = 10^{-3}$, $\lambda_{\text{SFD}} = 10^{-4}$, and $\gamma = 0.3$. For generation, we adopt DDIM sampling with 50 steps, a CFG scale of 7.5, and a resolution of 512×512, following Um & Ye (2025). We use *"a photo of [class name]"* for both training and generation to prevent detailed captions from explicitly exposing minority factor cues and interfering with factor discovery. For factor-level evaluation, we activate individual factors or their compositions. For distribution-level evaluation, we follow the reference-set class distribution and randomly sample factor compositions from feasible combinations observed during training. All baselines use the same dataset-adapted SD v1.5 backbone and generation settings for fair comparison. Further details are provided in the Appendix.

4.2 Factor-Level Evaluation

Single-Factor-Specified Minority Generation. Fig. 1 and Fig. 4 (a) showcase minority generation with individual factors discovered from Foods and ImageNet-1K, respectively. Compared with standard DDIM sampling, different factors elicit distinct yet semantically meaningful visual variations, demonstrating semantic coherence and effective factor disentanglement. Moreover, the same factor preserves consistent semantics across diverse classes, revealing cross-class reusability. Beyond discrete factor specification, we further examine continuous control over factor expression by varying the weights of factors discovered from ImageNet-1K. As shown in Fig. 4 (b), increasing a factor weight gradually modulates the corresponding visual characteristic, exhibiting a smooth response to factor strength. Together, these results show that LMF learns semantically coherent, cross-class

Table 1: Distribution-level evaluation on ImageNet-1K and Foods. LMF achieves the lowest cFID and sFID among all baselines, demonstrating superior minority distribution modeling.

Dataset	Method	cFID ↓	sFID ↓	Recall ↑	Precision ↑	CLIPScore ↑	PickScore ↑
IN-1K	SGMS Um & Ye (2024)	29.81±0.007	42.26±0.068	0.1832±0.0040	0.4428±0.0017	79.19±0.015	20.40±0.003
	BnS Um et al. (2025)	25.27±0.048	36.18±0.099	0.1865±0.0064	**0.6931±0.0022**	79.00±0.010	20.33±0.002
	MP Um & Ye (2025)	25.24±0.090	29.13±0.126	0.2341±0.0054	0.5907±0.0023	78.31±0.023	20.39±0.002
	JEPA Park & Um (2026)	30.77±0.035	36.69±0.101	0.2897±0.0061	0.3378±0.0041	77.69±0.027	20.43±0.004
	LMF (Ours)	**17.95±0.079**	**19.17±0.062**	**0.4372±0.0030**	0.5851±0.0034	**80.34±0.036**	**20.88±0.000**
Foods	SGMS Um & Ye (2024)	31.46±0.089	32.66±0.043	0.3142±0.0036	0.2858±0.0036	**79.91±0.008**	21.69±0.003
	BnS Um et al. (2025)	25.56±0.106	25.13±0.041	0.3475±0.0022	**0.4752±0.0003**	79.77±0.012	21.73±0.003
	MP Um & Ye (2025)	24.55±0.033	18.19±0.040	0.4421±0.0084	0.3961±0.0029	79.27±0.014	21.77±0.000
	JEPA Park & Um (2026)	31.28±0.147	33.38±0.111	0.3906±0.0046	0.2324±0.0029	79.15±0.009	**21.85±0.004**
	LMF (Ours)	**17.14±0.064**	**17.39±0.029**	**0.5496±0.0054**	0.3935±0.0007	79.03±0.010	21.74±0.006



(a) Factor Interpretation of Foods (b) Factor Interpretation of ImageNet-1K (c) Visualization of Langur in ImageNet-1K (d) Visualization of Lobster Roll in Foods

Figure 6: Post-hoc interpretation of learned minority factors in (a) and (b), and distribution visualization of generated results in (c) and (d). Font sizes are adjusted for readability.

reusable, and continuously controllable factors that enable pattern-specified minority generation beyond the capabilities of prior methods.

Multi-Factor-Specified Minority Generation. We further evaluate the compositionality of the learned factors by jointly activating multiple factors. As illustrated in Fig. 5 (a), the compositional generation results in the last column jointly retain the semantic effects of the three constituent factors shown in the middle columns, demonstrating that multiple minority factors can be coherently integrated within a single sample. Fig. 5 (b) further shows that individual factor strengths remain independently adjustable under composition. The compositionality of the learned factors highlights the advantage of decomposable minority generation, supporting flexible specification of multiple minority patterns.

Factor Interpretation. The learned factors provide a factor-level view of minority formation. To make their semantics directly interpretable, we employ GPT-5.6 Sol to assign semantic labels by contrasting factor-generated samples with their DDIM counterparts. Fig. 6 (a) and (b) present representative factors across datasets. These results reveal recurring visual structures within the minority distribution that are obscured by conventional minority generation methods. All labels are assigned post hoc and are never used for factor learning; further details are provided in the Appendix.

4.3 Distribution-Level Evaluation

Beyond the factor-level capabilities demonstrated above, we further examine whether the learned factors improve modeling of the overall target minority distribution. Since our primary objective is to approximate the predefined minority distribution, we regard cFID and sFID as the principal metrics, as they directly measure the discrepancy between generated and reference minority distributions. Tab. 1 shows that LMF consistently achieves the lowest cFID and sFID across both datasets. In particular, LMF reduces cFID over the strongest baseline by 28.9% and 30.2% on ImageNet-1K and Foods, respectively. LMF also achieves the highest Recall on both datasets, indicating broader coverage of the target minority distribution. Although BnS Um et al. (2025) attains higher Precision, this is accompanied by substantially lower Recall, suggesting that its generations concentrate on a narrower subset of the minority distribution. In contrast, LMF provides broader coverage while maintaining competitive Precision. Eventually, CLIPScore and PickScore vary marginally across methods, suggesting that text alignment and perceptual preference are relatively saturated.

To complement the quantitative evaluation, we visualize the feature distributions of real and generated minority samples from ImageNet-1K and Foods using t-SNE in Fig. 6 (c) and (d). Compared with BnS Um et al. (2025), whose samples are concentrated in relatively localized regions, LMF spans a broader portion of the real minority distribution. This qualitative observation is consistent

Table 2: Ablation studies of the minority factor pool and training objectives on ImageNet-10.

(a) Minority Factor Pool

Factor Pool	cFID ↓	sFID ↓	Den ↑	Cov ↑	Pre ↑	Rec ↑
None	44.91	81.67	0.3698	0.5487	0.4880	0.0860
Static	45.47	77.38	0.4086	0.5943	0.5663	0.1087
Decomposable	**26.35**	**76.72**	**0.5231**	**0.8787**	**0.5909**	**0.1903**

(b) Training Objectives

$\mathcal{L}_{\text{MSE}}$	$\mathcal{L}_{\text{Anti}}$	$\mathcal{L}_{\text{SFD}}$	cFID ↓	sFID ↓	Den ↑	Cov ↑	Pre ↑	Rec ↑
✓			35.75	76.35	0.4144	0.7117	0.5484	0.1853
✓	✓		27.40	**75.92**	0.4852	0.8740	0.5775	**0.1940**
✓		✓	31.70	77.26	0.4012	0.7257	0.5103	0.1740
✓	✓	✓	**26.35**	76.72	**0.5231**	**0.8787**	**0.5909**	0.1903

with the substantially higher Recall and lower cFID, further supporting the effectiveness of LMF in modeling diverse minority modes.

4.4 Ablation Study and Analysis

Effect of minority factor pool. This experiment aims to delineate the contribution of the pool design. We compare our decomposable factor pool with two controls: a single LoRA (*None*) and a static pool of 64 jointly activated LoRAs (*Static*). As reported in Tab. 2 (a), simply increasing the number of LoRAs yields only marginal gains, indicating that parameter capacity alone is insufficient. By contrast, the decomposable factor pool substantially improves all metrics, highlighting the importance of sample-wise factor decomposition over indiscriminate joint activation.

Effect of loss function. In Tab. 2 (b), we ablate the loss functions used in our model. The use of either $\mathcal{L}_{\text{Anti}}$ or $\mathcal{L}_{\text{SFD}}$ improves cFID over the MSE-only baseline, while jointly applying both yields broad improvements across the metrics. Moreover, $\mathcal{L}_{\text{Anti}}$ produces a more pronounced improvement than $\mathcal{L}_{\text{SFD}}$ when applied in isolation, indicating that broad factor participation plays a more critical role than factor disentanglement.

More Ablation Studies. Additional ablations on common-mode modeling, the detail-aware modulator, Siamese factor disentanglement, and hyperparameters are provided in the Appendix.

4.5 Application of Minority Samples

Table 3: Comparison of pretrained visual representations on visuo-motor control tasks.

Method	Kitchen	Block	Push-T	LIBERO
VC-1	2.63	0.05	0.38	0.91
MVP	2.31	0.00	0.20	0.88
ImageNet	2.77	0.12	0.39	0.91
+ SGMS	2.57	0.14	0.36	0.90
+ BnS	3.07	0.14	0.40	0.89
+ MP	2.62	0.14	0.38	0.89
+ JEPA	2.93	0.17	0.38	0.88
+ LMF (Ours)	**3.15**	**0.20**	**0.42**	**0.94**

Beyond minority generation itself, we further explore the broader value of generated minority samples for downstream intelligent systems. We consider visuo-motor control Cui et al. (2024) as a representative testbed, where policies predict robot actions from visual observations and rely heavily on the representational capacity of the visual encoder. Specifically, we augment ImageNet with the generated minority samples from Sec. 4.3 to train a ResNet-18 encoder, which is then frozen for VQ-BeT Lee et al. (2024a) on Franka Kitchen Gupta et al. (2020), Block Pushing Florence et al. (2021), Push-T Chi et al. (2025), and LIBERO Goal Liu et al. (2023). As shown in Tab. 3, LMF augmentation consistently outperforms both ImageNet pretraining and equal-sized prior minority synthetic augmentation across all tasks, while surpassing some conventional visuo-motor representation methods Majumdar et al. (2023); Xiao et al. (2022). Under an equal augmentation budget, we attribute these gains to broader coverage of the minority distribution, highlighting the utility of our minority samples for robust representation learning and downstream embodied AI. More in Appendix.

5 Conclusion

In this work, we introduce decomposable minority generation, a new paradigm grounded in a factor-level formulation of minority formation. We instantiate this paradigm with LMF (Learning Minority Factors), which discovers minority factors from a target distribution to support pattern-specified generation and provide the interpretation of minority characteristics. Experiments show that the learned factors exhibit coherent semantics, cross-class reusability, continuous controllability, and compositionality, while LMF also substantially improves minority distribution matching. Beyond generation, we further demonstrate that the resulting minority samples benefit downstream visuo-motor representations, highlighting the broader utility of minority data synthesis.

AI USE STATEMENT

We used LLMs to improve the clarity and readability of the manuscript, to assist in identifying relevant literature, and as a tool for post hoc interpretation of the learned minority factors based on their generated examples. All AI-assisted content and retrieved references were reviewed and verified by the authors. In addition, synthetic images generated by our proposed model were used to train a visual encoder for downstream evaluation, as described in the paper. The authors take full responsibility for the final content of this work.

REFERENCES

Omri Avrahami, Kfir Aberman, Ohad Fried, Daniel Cohen-Or, and Dani Lischinski. Break-a-scene: Extracting multiple concepts from a single image. In *SIGGRAPH Asia 2023 Conference Papers*, pp. 1–12, 2023.

Sana Ayromlou, Vahid Reza Khazaie, Fereshteh Forghani, and Arash Afkanpour. Can generative models improve self-supervised representation learning? In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1800–1808, 2025.

Varun Belagali, Srikar Yellapragada, Alexandros Graikos, Saarthak Kapse, Zilinghan Li, Tarak Nath Nandi, Ravi K Madduri, Prateek Prasanna, Joel Saltz, and Dimitris Samaras. Gen-sis: Generative self-augmentation improves self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2886–2896, 2026.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pp. 446–461, 2014.

Fernando Julio Cendra and Kai Han. Ice: Intrinsic concept extraction from a single image via diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23734–23743, 2025.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Zichen J Cui, Hengkai Pan, Aadhithya Iyer, Siddhant Haldar, and Lerrel Pinto. Dynamo: In-domain dynamics pretraining for visuo-motor control. *Advances in Neural Information Processing Systems*, 37:33933–33961, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 248–255, 2009.

Pete Florence, Corey Lynch, Andy Zeng, Oscar A Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. Implicit behavioral cloning. In *5th Annual Conference on Robot Learning*, 2021.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations*, 2023.

Shelly Golan, Yotam Nitzan, Zongze Wu, Or Patashnik, et al. Vlm-guided adaptive negative prompting for creative generation. In *International Conference on Learning Representations*, volume 2026, pp. 4692–4714, 2026.

Abhishek Gupta, Vikash Kumar, Corey Lynch, Sergey Levine, and Karol Hausman. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. In *Proceedings of the Conference on Robot Learning*, pp. 1025–1037, 2020.

Jiyeon Han, Dahee Kwon, Gayoung Lee, Junho Kim, and Jaesik Choi. Enhancing creative generation on stable diffusion-based models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28609–28618, 2025.

Shaozhe Hao, Kai Han, Zhengyao Lv, Shihao Zhao, and Kwan-Yee K Wong. Conceptexpress: Harnessing diffusion models for single-image unsupervised concept extraction. In *European Conference on Computer Vision*, pp. 215–233, 2024.

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*, pp. 7514–7528, 2021.

Jang-Hyun Kim, Jinuk Kim, Seong Joon Oh, Sangdoo Yun, Hwanjun Song, Joonhyun Jeong, Jung-Woo Ha, and Hyun Oh Song. Dataset condensation via efficient synthetic-data parameterization. In *International Conference on Machine Learning*, pp. 11102–11118, 2022.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1931–1941, 2023.

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.

Seungjae Lee, Yibin Wang, Haritheja Etukuru, H. Jin Kim, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Behavior generation with latent actions. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 26991–27008, 2024a.

Sharon Lee, Yunzhi Zhang, Shangzhe Wu, and Jiajun Wu. Language-informed visual concept learning. In *International Conference on Learning Representations*, pp. 26267–26294, 2024b.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems*, pp. 44776–44791, 2023.

Arjun Majumdar, Karmesh Yadav, Sergio Arnaud, Jason Ma, Claire Chen, Sneha Silwal, Aryan Jain, Vincent-Pierre Berges, Tingfan Wu, Jay Vakil, et al. Where are we in the search for an artificial visual cortex for embodied intelligence? In *Advances in Neural Information Processing Systems*, pp. 655–677, 2023.

Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *International conference on machine learning*, pp. 7176–7185, 2020.

Charlie Nash, Jacob Menick, Sander Dieleman, and Peter Battaglia. Generating images with sparse representations. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 7958–7968, 2021.

Dongmin Park, Sebin Kim, Taehong Moon, Minkyu Kim, Kangwook Lee, and Jaewoong Cho. Rare-to-frequent: Unlocking compositional generation power of diffusion models on rare concepts with llm guidance. In *International conference on learning representations*, volume 2025, pp. 14650–14676, 2025.

Sol Park and Soobin Um. Beyond generative priors: Minority sampling with JEPA-guided diffusion. In *Forty-third International Conference on Machine Learning*, 2026.

Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in gan evaluation. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11400–11410, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685, 2022.

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 *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22500–22510, 2023.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in neural information processing systems*, pp. 36479–36494, 2022.

Vikash Sehwag, Caner Hazirbas, Albert Gordo, Firat Ozgenel, and Cristian Canton. Generating high fidelity data from low-density regions using diffusion models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pp. 11492–11501, 2022.

Hanyu Shi, Hong Tao, Guoheng Huang, Jianbin Jiang, Xuhang Chen, Chi-Man Pun, Shanhu Wang, and Pan Pan. Intrinsic concept extraction based on compositional interpretability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 38969–38978, 2026.

Yiren Song, Xiaokang Liu, and Mike Zheng Shou. Diffsim: Taming diffusion models for evaluating visual similarity. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16904–16915. IEEE, 2025.

Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *Advances in neural information processing systems*, 2023.

Soobin Um and Jong Chul Ye. Self-guided generation of minority samples using diffusion models. In *European Conference on Computer Vision*, pp. 414–430, 2024.

Soobin Um and Jong Chul Ye. Minority-focused text-to-image generation via prompt optimization. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20926–20936, 2025.

Soobin Um, Suhyeon Lee, and Jong Chul Ye. Don't play favorites: Minority guidance for diffusion models. In *International Conference on Learning Representations*, pp. 21632–21670, 2024.

Soobin Um, Beomsu Kim, and Jong Chul Ye. Boost-and-skip: A simple guidance-free diffusion for minority generation. In *Forty-second International Conference on Machine Learning*, 2025.

Yael Vinker, Andrey Voynov, Daniel Cohen-Or, and Ariel Shamir. Concept decomposition for visual exploration and inspiration. *ACM Transactions on Graphics (TOG)*, 42(6):1–13, 2023.

Zerun Wang, Jiafeng Mao, Xueting Wang, and Toshihiko Yamasaki. Difficulty controlled diffusion model for synthesizing effective training data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 10367–10375, 2026.

Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15897–15907, 2023.

Tete Xiao, Ilija Radosavovic, Trevor Darrell, and Jitendra Malik. Masked visual pre-training for motor control. *arXiv preprint arXiv:2203.06173*, 2022.

Zhi Xu, Shaozhe Hao, and Kai Han. Cusconcept: Customized visual concept decomposition with diffusion models. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3678–3687. IEEE, 2025.

Ning Yu, Ke Li, Peng Zhou, Jitendra Malik, Larry Davis, and Mario Fritz. Inclusive gan: Improving data and minority coverage in generative models. In *European Conference on Computer Vision*, pp. 377–393, 2020.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

Yanbing Zhang, Mengping Yang, Qin Zhou, and Zhe Wang. Attention calibration for disentangled text-to-image personalization. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4764–4774, 2024.

A Additional Implementation Details

A.1 Training Set Partition.

The training set is partitioned based on the reconstruction-based minority score defined in Eq. 3. Specifically, we use Stable Diffusion v1.5 as the T2I model for minority scoring. For each image, we add noise at a fixed timestep $t = 600$ according to Eq. 1, and then reconstruct the image using the Tweedie estimate. The denoiser is conditioned on the prompt `a photo of [class name]` throughout minority scoring. Classifier-free guidance is disabled during scoring. We repeat this procedure with $R = 5$ independently sampled Gaussian noises, as defined in Eq. 3, and use the average VGG-LPIPS distance between the original image and the decoded Tweedie estimate as the final minority score. Images are then ranked according to this score: the bottom 50% are used to learn the common-mode knowledge $\Delta\mathbf{W}^c$, while the top 50% are used as the training set for minority factor discovery and learning the minority factor pool.

A.2 Training Details of $\Delta\mathbf{W}^c$.

The common-mode knowledge $\Delta\mathbf{W}^c$ is parameterized as a standard LoRA update applied to the SD1.5 UNet. It is trained on the bottom 50% subset defined by the minority score and remains fixed during minority factor learning. We insert LoRA layers into the UNet attention projections `to_q`, `to_k`, `to_v`, and `to_out.0`, with both the LoRA rank and alpha set to 8. During training, the VAE, text encoder, and pretrained UNet parameters are frozen, and only the LoRA parameters are optimized. We use AdamW with a learning rate of 10^{-4}, betas $(0.9, 0.999)$, weight decay 10^{-2}, and epsilon 10^{-8}. Gradient clipping is applied with a maximum norm of 1.0. We adopt a constant learning-rate schedule with 5 warmup steps. The per-device batch size is 64 with a gradient accumulation step of 1. Training is performed with bfloat16 mixed precision, TF32 matrix multiplication, and gradient checkpointing. The common-mode LoRA is trained for 10 epochs on ImageNet-1K and 100 epochs on ImageNet-10 and Foods.

A.3 Details of the Minority Factor Pool

The minority factor pool contains 64 learnable factor entries. Each entry is implemented as an independent LoRA update $\Delta W_i = B_i A_i$ on every target projection. We use the same target projections as the common-mode LoRA: `to_q`, `to_k`, `to_v`, and `to_out.0`. The rank and alpha of each factor are both 8. In SD1.5 this places the factor pool on 128 linear projection layers. During minority-factor optimization, the pretrained SD1.5 parameters and ΔW_c remain frozen; only the factor pool and the routing module are optimized.

A.4 Details of the Detail-aware Modulator

The detail-aware representation is extracted from the frozen denoiser. To prevent the routing feature from depending on the randomly sampled diffusion timestep used by the denoising objective, we construct a separate routing latent using a fixed Gaussian noise tensor and a fixed timestep $t_{\text{fix}} = 100$. The fixed noise tensor is generated with seed 42 and shared across all samples. Following prior work Tang et al. (2023), we feed this routing latent into the denoiser with the text condition `a photo of [class name]` and extract the activation from layer index 16, corresponding to `mid_block.attentions.0`. The resulting activation is globally averaged over the spatial dimensions to obtain a 1280-dimensional routing feature.

A.5 Details of the Factor Routing Module

The factor routing module is implemented as a two-layer MLP:

$$\text{Linear}(1280, 1024) \rightarrow \text{SiLU} \rightarrow \text{Linear}(1024, 64) \rightarrow \text{Sigmoid}.$$

For each image, we select the Top-3 factor entries according to the 64 sigmoid outputs. The corresponding factor updates are then combined using their sigmoid routing weights. During training, the discrete Top-3 indices are detached from gradient propagation, while the selected routing weights remain differentiable. This allows the denoising objective to optimize the routing module through the activated factor entries while preserving the discrete factor selection.

B Detailed Evaluation Protocol

B.1 Reference-following Evaluation.

All distribution-level comparisons are conducted against the minority reference set of each dataset, constructed by ranking training images according to the minority score described above. For ImageNet-1K and Foods, the reference set contains 30K images, while ImageNet-10 uses 3K images. For all datasets, each method generates 30K samples whose class distribution follows that of the corresponding reference set rather than a uniform distribution. Specifically, each generated sample inherits the class prompt of a reference-set entry, ensuring that all methods are evaluated under the same class composition and sampling budget.

B.2 Evaluation Metrics

Image-text quality metrics. We report CLIPScore Hessel et al. (2021) and PickScore Kirstain et al. (2023) as reference-free image-text alignment metrics. CLIPScore is computed with OpenAI CLIP ViT-B/32 following the standard CLIPScore formulation,

$$\mathrm{CLIPScore}(x, c) = 2.5 \cdot \max\left(\cos\left(f_{\mathrm{img}}(x), f_{\mathrm{text}}(c)\right), 0\right),$$

where the text input is prefixed with `A photo depicts`. In tables, we report $100\times$ the mean CLIPScore over all generated images. PickScore is computed with `yuvalkirstain/PickScore_v1`. We use the raw PickScore matching logit between each image and its prompt, averaged over generated images.

Clean FID. Clean FID (cFID) Parmar et al. (2022) is computed between the reference image set and the generated image set using the `clean-fid` implementation in clean mode with the InceptionV3 feature extractor. The generated images are compared directly against the reference-set directory, not against the full training set. This metric measures whether the generated distribution matches the minority reference distribution.

Spatial FID. Spatial FID (sFID) Nash et al. (2021) is computed with the official OpenAI guided-diffusion evaluation code. Images are resized to 256×256 with bicubic interpolation before feature extraction. Unlike cFID, which uses pooled Inception features, sFID uses spatial Inception activations and computes a Frechet distance between the spatial feature statistics of the generated set and the reference set. We use the TensorFlow guided-diffusion evaluator for compatibility with the original sFID protocol.

Precision, Recall, Density, and Coverage. While cFID and sFID serve as our primary distribution-level metrics, we additionally report Precision and Recall Kynkäänniemi et al. (2019), as well as Density and Coverage Naeem et al. (2020), to provide complementary measurements of sample fidelity and distribution coverage. All metrics are computed using their official implementations.

Let $\mathcal{R}$ and $\mathcal{G}$ denote the reference and generated feature sets, respectively. Let $\rho_{\mathcal{R}}(r)$ be the distance from a reference feature $r \in \mathcal{R}$ to its k-th nearest neighbor in $\mathcal{R}$, and define $\rho_{\mathcal{G}}(g)$ analogously for generated features. Precision measures the fraction of generated samples that fall inside the reference manifold:

$$\mathrm{Precision} = \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \mathbf{1}\left[\exists r \in \mathcal{R} : \|g - r\|_2 \le \rho_{\mathcal{R}}(r)\right].$$

Recall measures the fraction of reference samples covered by the generated manifold:

$$\mathrm{Recall} = \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \mathbf{1}\left[\exists g \in \mathcal{G} : \|r - g\|_2 \le \rho_{\mathcal{G}}(g)\right].$$

Density measures how many reference neighborhoods contain each generated sample, normalized by k:

$$\mathrm{Density} = \frac{1}{k|\mathcal{G}|} \sum_{g \in \mathcal{G}} \sum_{r \in \mathcal{R}} \mathbf{1}\left[\|g - r\|_2 \le \rho_{\mathcal{R}}(r)\right].$$

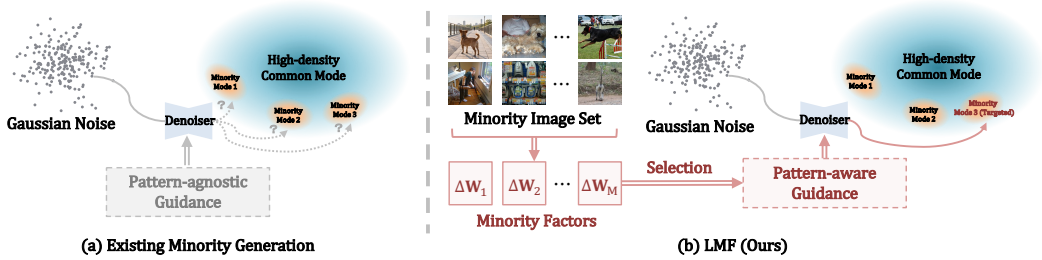


Figure 7: Comparison between existing minority generation methods and LMF. Prior methods steer generation toward minority regions through pattern-agnostic guidance without explicitly modeling what constitutes each minority sample. In contrast, LMF discovers reusable minority factors from the minority image set and selectively composes them to provide pattern-aware guidance, enabling targeted minority generation.

Coverage measures the fraction of reference samples whose neighborhoods contain at least one generated sample:

$$\text{Coverage} = \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \mathbf{1}\left[\min_{g \in \mathcal{G}} \|r - g\|_2 \leq \rho_{\mathcal{R}}(r)\right].$$

B.3 Repeated runs and reporting.

For stochastic generation baselines, we evaluate three independent runs with seeds 42, 0, and 1024. The reported mean and standard deviation are computed across these runs.

C Baseline Implementation Details

We compare LMF with SGMS Um & Ye (2024), BnS Um et al. (2025), MP Um & Ye (2025), and JEPA Park & Um (2026). For all baselines, we use the official code repositories released by the authors and retain their original method-specific hyperparameter settings whenever applicable. Since SGMS Um & Ye (2024), BnS Um et al. (2025), and JEPA Park & Um (2026) were originally developed under conditional generative settings, we adapt their implementations to the T2I setting for a consistent comparison with LMF. All methods are implemented with Stable Diffusion v1.5 as the generative backbone. To reduce the domain gap and ensure a fair comparison, we first adapt SD v1.5 to each dataset using the same LoRA-based fine-tuning protocol, and use the corresponding dataset-adapted model as the backbone for all compared methods.

C.1 Dataset Adaptation.

For each dataset, Stable Diffusion v1.5 is adapted through LoRA fine-tuning of the UNet. The VAE, text encoder, and original UNet parameters are frozen, while LoRA modules are inserted into the UNet attention projections `to_q`, `to_k`, `to_v`, and `to_out.0`. Both the LoRA rank and alpha are set to 8. We use the text template `a photo of [class name]` for all training samples. Training follows the standard latent diffusion noise-prediction objective with randomly sampled timesteps and Min-SNR loss weighting with $\gamma = 5$. Images are resized to 512×512, center-cropped, randomly horizontally flipped, and normalized to $[-1, 1]$. We optimize the LoRA parameters using AdamW with a learning rate of 10^{-4}, cosine learning-rate decay, and 500 warmup steps. Training uses bfloat16 mixed precision, and the LoRA scale is set to 1.0 during inference.

C.2 Baseline Configuration.

After dataset adaptation, each baseline follows the hyperparameter configuration provided in its original implementation, while all generation settings shared with LMF are kept identical to ensure a controlled comparison. The only exception is BnS Um et al. (2025). When directly applying its original sampling configuration to the adapted T2I model, we observe that the generated outputs are dominated by noise and fail to form natural images. We therefore use a milder perturbation by scaling the initial Gaussian latent with $\gamma = 1.05$ and skipping the first $\Delta t = 3$ highest-noise DDIM sampling steps. All other BnS settings remain unchanged from its original implementation.

C.3 Further Discussion on Baselines and LMF.

The key difference between existing minority generation methods and LMF lies in whether the minority structure is explicitly modeled. As illustrated in Fig. 7, prior methods Um & Ye (2024); Um et al. (2025); Um & Ye (2025); Park & Um (2026) provide pattern-agnostic guidance that steers generation away from the high-density common mode toward minority regions. However, since they do not explicitly identify the constituent factors of minority samples, the generated sample cannot be controlled toward a specific minority pattern. In contrast, LMF first discovers a set of minority factors from the minority image set and represents them as learnable factor updates. By selecting and composing the desired factors during generation, LMF converts the original pattern-agnostic guidance into pattern-aware guidance, enabling pattern-specified minority generation toward a targeted minority mode.

D Visuo-Motor Control Details

D.1 Purpose of the downstream evaluation.

We use visuo-motor control as a downstream testbed to examine whether minority samples improve visual representations beyond image generation metrics. The evaluation follows the DynaMo Cui et al. (2024) control protocol. A visual encoder is first pretrained and then frozen as the observation encoder for downstream policy learning. During policy training, the visual encoder remains fixed, while the policy learner is optimized on the corresponding control task. Across all experiments, the policy architecture, task data, and optimization settings are kept identical, and only the pretrained visual encoder is varied.

D.2 Visual Encoder Construction.

We use a ResNet-18 visual encoder with a 512-dimensional penultimate representation. For ImageNet in Tab. 3, we directly use the torchvision ImageNet-1K pretrained ResNet-18 encoder. For all +X variants, we further fine-tune the pretrained encoder as a 1000-way ImageNet classifier using the original ImageNet training set augmented with synthetic ImageNet samples generated by the corresponding method. All variants use the same number of synthetic samples and follow an identical classifier fine-tuning protocol, providing a controlled comparison of the utility of synthetic data generated by different methods.

The ResNet-18 fine-tuning follows standard ImageNet preprocessing. Training images are transformed using random resized cropping to 224×224, random horizontal flipping, and ImageNet normalization. Validation images are resized to 256, center-cropped to 224×224, and normalized using the ImageNet statistics. We fine-tune the classifier for 50 epochs using SGD with a learning rate of 10^{-4}, momentum 0.9, and weight decay 10^{-4}. The learning rate is decayed by a factor of 0.1 at epoch 30. We use no label smoothing and enable mixed-precision training. The checkpoint with the highest ImageNet validation Top-1 accuracy is selected. Finally, the classification head is discarded, and the resulting ResNet-18 backbone is frozen and used as the visual encoder for downstream control.

D.3 Tasks and Evaluation Metrics

We evaluate the visual representations on four visuo-motor control tasks: Franka Kitchen Gupta et al. (2020), Block Push Florence et al. (2021), Push-T Chi et al. (2025), and LIBERO Goal Liu et al. (2023), following the benchmark setup of DynaMo Cui et al. (2024). These tasks cover different manipulation behaviors and use task-specific evaluation metrics. For Franka Kitchen, we report the number of completed subtasks. For Block Push, performance is measured by whether the manipulated object enters the target zone. For Push-T, we report the final coverage between the pushed object and the target region. For LIBERO Goal, we use the task success rate. Higher values indicate better performance for all metrics.

D.4 POLICY LEARNER.

The downstream policy is VQ-BeT Lee et al. (2024a). For each control task, visual observations are encoded frame-wise by the frozen ResNet encoder. For multi-view environments, each view is encoded independently, and the resulting view embeddings are concatenated before being fed into the policy. The visual encoder remains frozen throughout policy training, so the downstream performance differences mainly reflect the quality of the pretrained visual representations rather than additional visual adaptation on robot data. All policy training and evaluation procedures follow DynaMo Cui et al. (2024).

E MECHANISM OF ANTI-COLLAPSE REGULARIZATION

This section analyzes the anti-collapse regularization defined in Eq. 10 and Eq. 11, explaining how it provides corrective signals to suppressed routing outputs while preserving sample-specific factor selection.

Soft estimation of batch-level factor usage. For a mini-batch of B samples and a pool of M factors, let $w_{b,i} \in (0, 1)$ denote the routing weight of factor i for sample b. As defined in Eq. 10, the routing weights are normalized within each sample and then averaged across the mini-batch:

$$p_{b,i} = \frac{w_{b,i}}{\sum_{j=1}^{M} w_{b,j}}, \qquad q_i = \frac{1}{B} \sum_{b=1}^{B} p_{b,i}.$$

The resulting distribution $\mathbf{q} = [q_1, \ldots, q_M]$ is a soft estimate of batch-level factor usage, computed before Top-K selection, rather than the empirical frequency of discrete selections. Following Eq. 11, we match this distribution to the uniform reference $\mathbf{u} = [1/M, \ldots, 1/M]$:

$$\mathcal{L}_{\text{Anti}} = D_{\text{KL}}\left(\mathbf{q} \,\|\, \mathbf{u}\right) = \sum_{i=1}^{M} q_i \log \frac{q_i}{1/M}.$$

The normalization above is used to estimate factor usage for regularization; the selected routing weights $w_{b,i}$ are still used for parameter composition.

Gradient signals for suppressed routing outputs. With the discrete Top-K indices held fixed during backpropagation, an unselected routing output does not directly affect the denoising prediction and therefore receives no direct gradient from $\mathcal{L}_{\text{MSE}}$ for that sample. Although updates to shared router parameters can still change this output, the sparse denoising objective does not directly optimize it. In contrast, $\mathcal{L}_{\text{Anti}}$ depends on all routing outputs before selection. Differentiating Eq. 11 gives:

$$\begin{aligned} \frac{\partial \mathcal{L}_{\text{Anti}}}{\partial w_{b,i}} &= \sum_{j=1}^{M} \left(\log \frac{q_j}{1/M} + 1 \right) \frac{\partial q_j}{\partial w_{b,i}} \\ &= \frac{1}{B \sum_{\ell=1}^{M} w_{b,\ell}} \left[\log \frac{q_i}{1/M} - \sum_{j=1}^{M} p_{b,j} \log \frac{q_j}{1/M} \right]. \end{aligned} \tag{15}$$

Here, the constant terms cancel because $\sum_{j=1}^{M} p_{b,j} = 1$. When the log-ratio of factor i is below the $p_{b,j}$-weighted average of the log-ratios, Eq. 15 is negative, providing a gradient direction that increases $w_{b,i}$. In particular, a factor with the smallest q_i receives such a direction whenever $\mathbf{q}$ is nonuniform.

Thus, a factor need not already belong to the Top-K set to receive a corrective routing signal. By counteracting persistently low relative routing weights, the regularizer can make a previously suppressed factor more competitive for selection on subsequent samples.

Batch-level balancing preserves sample-specific routing. The regularizer acts on the batch-averaged distribution $\mathbf{q}$, rather than requiring the normalized routing weights of every sample to

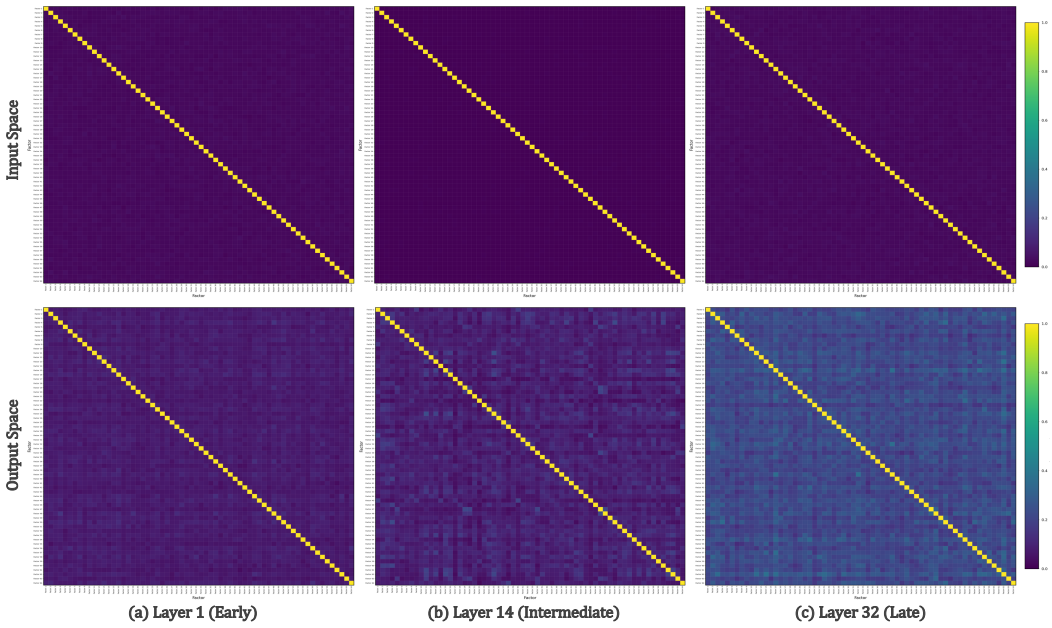


Figure 8: Pairwise subspace similarity of the learned minority-factor LoRA updates across representative network depths. The top and bottom rows show input- and output-subspace similarities, respectively, for the `to_q` projections at layers 1, 14, and 32. Low off-diagonal similarities indicate that the learned factors occupy largely distinct parameter subspaces, supporting effective factor disentanglement.

be uniform. Different samples can therefore assign larger weights to different factors while maintaining a balanced batch-level usage estimate. A factor may remain weakly associated with some samples and receive larger routing weights on others. Accordingly, the regularizer discourages aggregate suppression without requiring every factor to participate in every sample. Sample-dependent routing and task-driven optimization remain responsible for determining which factors are relevant to each image.

Relation to Top-K selection and early training. Because Top-K selection depends on relative rankings, a balanced soft usage distribution does not necessarily imply balanced discrete selection frequencies. The purpose of $\mathcal{L}_{\text{Anti}}$ is therefore to discourage premature suppression of routing outputs and preserve opportunities for broader factor participation, rather than to guarantee uniform Top-K assignments. Consistent with the main text, this regularizer is applied only during early training and is subsequently removed. This schedule encourages broader participation while the routing structure is being established, and then permits data-dependent factor specialization without imposing a uniform final usage distribution.

F Visualization of Factor Disentanglement

We visualize the degree of coupling between learned minority factors by measuring the overlap of their LoRA update subspaces. Specifically, we examine both the input and output subspaces of the query projections (`to_q`) at layers 1, 14, and 32, revealing how different factors interact at different network depths.

SVD-based subspace analysis. For each factor i, we first reconstruct its effective parameter update $\Delta W_i = \frac{\alpha}{r} B_i A_i$ at the corresponding layer and projection, where α is the LoRA scaling coefficient and r is the LoRA rank. We then perform singular value decomposition:

$$\Delta W_i = \widehat{U}_i S_i \widehat{V}_i^{\top}. \tag{16}$$

Let U_i and V_i contain the first r left and right singular vectors, respectively, where r is the LoRA rank. The columns of U_i span the output update subspace, while those of V_i span the input update subspace.

Table 4: Distribution-level evaluation on ImageNet-10. LMF achieves the lowest cFID and sFID among all baselines, demonstrating superior minority distribution modeling.

Method	cFID ↓	sFID ↓	Recall ↑	Precision ↑	CLIPScore ↑	PickScore ↑
SGMS	40.87±0.074	90.65±0.104	0.0967±0.0030	0.3705±0.0038	80.82±0.006	20.34±0.002
BnS	34.74±0.023	85.13±0.148	0.1166±0.0068	**0.6158±0.0016**	**80.87±0.010**	20.31±0.001
MP	43.10±0.209	80.45±0.084	0.1668±0.0107	0.4740±0.0035	80.16±0.015	20.35±0.003
JEPA	44.90±0.101	87.36±0.014	0.1741±0.0187	0.2694±0.0013	79.90±0.004	**20.39±0.001**
LMF (Ours)	**26.35±0.110**	**76.72±0.074**	**0.1903±0.0038**	0.5909±0.0005	79.69±0.023	20.14±0.000



Figure 9: Example grid used for post-hoc factor interpretation. For each class, images are generated using the same learned factor, random seed, and text prompt, while varying only the LoRA scaling weight. The class label is shown below each group, and the corresponding LoRA scaling weights are shown above the generated images. The resulting grid is provided to GPT-5.6 Sol for semantic interpretation.

Reconstructing ΔW_i allows us to analyze the effective update rather than its non-unique LoRA factorization. SVD further separates the update directions from their singular-value magnitudes and provides orthonormal bases for the two subspaces. Consequently, we can measure whether different factors operate along overlapping directions, independently of differences in update strength.

For any two factors i and j at the same layer and projection, we compute

$$\mathrm{Sim}_U(i,j) = \frac{\left\|U_i^\top U_j\right\|_F^2}{r}, \qquad \mathrm{Sim}_V(i,j) = \frac{\left\|V_i^\top V_j\right\|_F^2}{r}, \tag{17}$$

which quantify output- and input-subspace overlap, respectively. Both similarities lie in $[0,1]$, with larger values indicating stronger overlap. A value of 1 denotes identical subspaces, whereas 0 denotes orthogonal subspaces. These measures compare entire subspaces rather than individual singular vectors and are invariant to the choice of orthonormal basis within each subspace.

Visualization and results. Fig. 8 presents the pairwise input- and output-subspace similarity matrices for all 64 factors at the `to_q` projections of representative layers 1, 14, and 32. Each row and column corresponds to one factor, and the off-diagonal entries measure the degree of coupling between different factors. The input-subspace similarities remain consistently low across the three layers, indicating that different factors operate on largely distinct input directions. The output-subspace similarities are also low to moderate, with a more visible shared structure in the late decoder layer. This suggests that some factors may produce partially aligned output updates at deeper layers, while still being driven by distinct input subspaces. Overall, the low off-diagonal similarities indicate that the learned factors do not collapse to identical LoRA update directions, supporting the parameter-space disentanglement of the minority factor pool.

G Main Distribution-Level Evaluation on ImageNet-10

We further report the distribution-level evaluation on ImageNet-10, following the same evaluation protocol as Sec. 4.3. As shown in Tab. 4, LMF achieves the best cFID, sFID, and Recall among all compared methods. The lower cFID and sFID indicate that the generated samples better match the target minority distribution. Meanwhile, the higher Recall suggests that LMF covers a broader portion of the target minority distribution rather than concentrating on a limited set of minority modes. These results are consistent with the observations in Sec. 4.3, further confirming the effectiveness of LMF in modeling diverse minority regions.

Table 5: Ablation study of common-mode modeling on ImageNet-10.

Strategy	cFID ↓	sFID ↓	Den ↑	Cov ↑	Pre ↑	Rec ↑
Joint Common-Minority	29.55	77.99	0.4767	0.8080	0.5867	0.2107
Direct Minority	28.45	**75.63**	0.4079	0.7813	0.5412	**0.2467**
Residual Isolation	**26.35**	76.72	**0.5231**	**0.8787**	**0.5909**	0.1903

Table 6: Ablation study of the detail-aware modulator on ImageNet-10.

Visual Rep.	cFID ↓	sFID ↓	Den ↑	Cov ↑	Pre ↑	Rec ↑	CS ↑	PS ↑
CLIP-B/32	28.57	**75.98**	0.4382	0.7797	0.5315	0.2033	79.30	20.11
DINOv2-B/14	27.30	76.34	0.4272	0.8470	0.5154	0.2217	78.90	20.02
ResNet-50	27.75	76.90	0.4057	0.8100	0.5114	**0.2613**	79.15	20.12
ViT-B/16	28.50	77.32	0.4236	0.8410	0.5308	0.2390	79.18	20.09
Ours	**26.35**	76.72	**0.5231**	**0.8787**	**0.5909**	0.1903	**79.69**	**20.14**

H Post-hoc Factor Interpretation

To facilitate the semantic interpretation of the learned minority factors, we use GPT-5.6 Sol to perform post-hoc analysis of the visual effects induced by each learned LoRA factor. Importantly, GPT is used only for interpretation after model training and does not participate in factor discovery, factor routing, or generation. Specifically, we activate one learned factor at a time while keeping the random seed and text prompt fixed. For each factor, we vary its LoRA scaling weight and generate the corresponding images under otherwise identical sampling conditions. The generated images are then arranged into a grid according to the LoRA weight, allowing the visual changes induced by increasing the strength of a single factor to be directly compared. An illustration of such an interpretation grid is shown in Fig. 9.

We provide the grid to GPT-5.6 Sol and ask it to identify the visual characteristics that change consistently with the LoRA weight and summarize the dominant visual semantics using a concise phrase. The returned descriptions are subsequently used as the post-hoc interpretations of the corresponding learned factors. Therefore, the semantic factor names reported in the paper are descriptive labels assigned after training based on generated visual evidence, rather than semantic annotations used to supervise factor learning. The prompt used for factor interpretation is provided below:

Prompt for Factor Interpretation

You are given an image grid generated using the same learned LoRA factor across multiple object classes. For each class, the LoRA scaling weight increases from left to right, while the random seed, text prompt, and all other generation settings are fixed.
Your task is to identify the visual characteristic introduced by this factor.
Please:

1. Compare how the images change as the LoRA weight increases.
2. Identify the visual change that is consistent across different classes.
3. Focus on interpretable visual attributes.
4. Summarize the dominant shared effect using a concise semantic phrase of 1–4 words.

Output format:
Factor interpretation: [short semantic phrase]

I More Ablation Studies

I.1 Effect of Common-mode Modeling.

As defined in Sec. 3.2, we formulate minority factors as residual information relative to the common mode. To validate this design, we compare three training strategies: joint common-minority training,

Table 7: Ablation study of Siamese Factor Disentanglement on ImageNet-10.

Siamese Pair	cFID ↓	sFID ↓	Pre ↑	Rec ↑	Den ↑	Cov ↑	CS ↑	PS ↑
None	27.40	**75.92**	0.5775	0.1940	0.4852	0.8740	79.62	20.13
Random	28.90	79.65	0.5373	**0.2097**	0.4039	0.8030	79.44	**20.20**
Inter-sample	26.94	76.20	0.5408	0.1897	0.5071	0.8753	79.49	20.06
Intra-sample	**26.35**	76.72	**0.5909**	0.1903	**0.5231**	**0.8787**	**79.69**	20.14

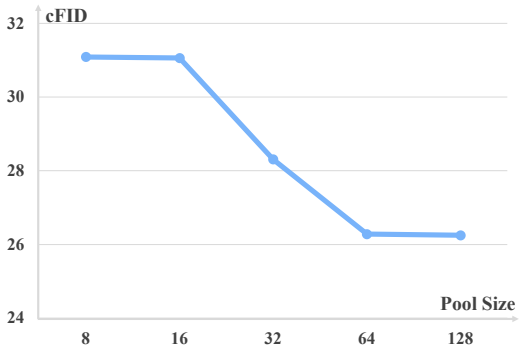


Figure 10: Effect of minority factor pool size on ImageNet-10. Increasing the pool size improves cFID substantially before saturating around $M = 64$.

minority-only training without $\Delta\mathbf{W}^c$, and our explicit common-mode modeling with a fixed $\Delta\mathbf{W}^c$. As reported in Tab. 5, explicit common-mode modeling achieves the best cFID, Density, Coverage, and Precision. Without an explicit common-mode reference, the factor entries must simultaneously model shared category-level information and minority-specific variations, which can entangle the learned factors with common visual patterns. By separating common information through $\Delta\mathbf{W}^c$, the factor pool can focus on modeling the residual variations that distinguish minority samples, supporting our formulation of minority factors as deviations from the common mode.

I.2 Effect of Detail-aware Modulator.

We examine the impact of visual representations on factor routing. We compare our detail-aware modulator with representative visual encoders, including CLIP-B/32, DINOv2-B/14, ResNet-50, and ViT-B/16, covering vision-language, self-supervised, convolutional, and transformer-based representations. As reported in Tab. 6, our detail-aware modulator achieves the best cFID, Precision, Density, Coverage, CLIPScore, and PickScore among all representation choices. These generic visual encoders are primarily optimized for semantic alignment or recognition, and therefore tend to suppress appearance variations that are irrelevant to category semantics. However, such fine-grained variations, including texture, presentation, local composition, and contextual details, are precisely important for identifying minority factors. In contrast, diffusion features retain richer image-level details required for denoising, making them better suited for minority-factor routing.

I.3 Effect of Siamese Factor Disentanglement.

We investigate different Siamese pair construction strategies to determine an effective and efficient design for SFD. We compare no SFD, random pairing, inter-sample pairing, and the proposed intra-sample pairing. For inter-sample pairing, we select a negative sample within the same mini-batch and use its Top-K routed factors to construct the Siamese counterpart, whereas intra-sample pairing constructs the counterpart from the same sample with different factor compositions. As shown in Tab. 7, intra-sample pairing achieves the best results on the largest number of metrics, including Precision, Density, Coverage, and CLIPScore. Although its FID performance is comparable to inter-sample pairing, intra-sample pairing avoids processing an additional negative sample and therefore introduces lower computational overhead. We consequently adopt intra-sample pairing for SFD.

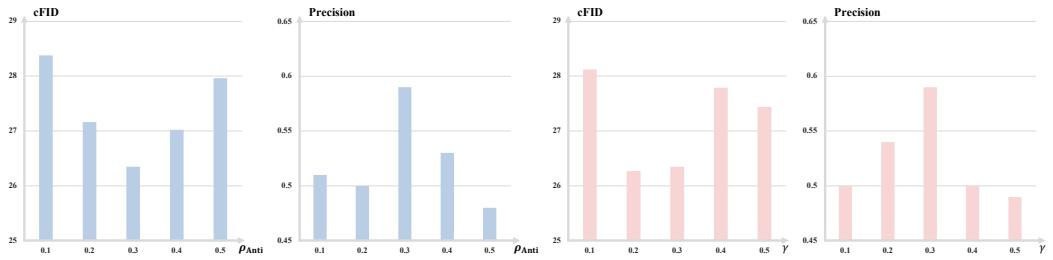


Figure 11: Sensitivity analysis of the anti-collapse duration ρ_{Anti} and SFD margin γ on ImageNet-10. We evaluate their effects on cFID and Precision.

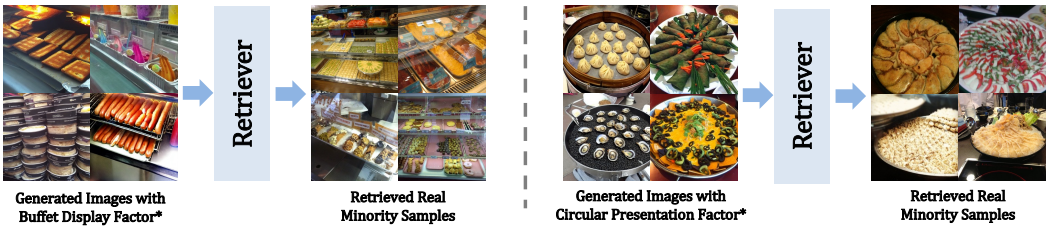


Figure 12: Training-set evidence for learned minority factors. For each learned factor, we construct a factor prototype from its generated samples and retrieve the nearest real samples from the minority set in DINOv2 feature space. The retrieved samples exhibit visual patterns consistent with the corresponding factor semantics, such as *Buffet Display* and *Circular Presentation*.

I.4 Effect of Hyperparameters

We first examine the effect of minority factor pool size M. As shown in Fig. 10, increasing M from 8 to 64 substantially reduces cFID, indicating that a larger pool provides greater capacity to capture diverse minority factors. The improvement saturates beyond $M = 64$, with little additional gain at $M = 128$. We therefore set $M = 64$ as the default pool size.

We also study the sensitivity of two hyperparameters introduced in our training objectives. The first is ρ_{Anti} in Eq. 14, which controls the portion of training during which the anti-collapse regularization is applied. The second is the margin γ in the SFD loss (Eq. 13), which determines the separation enforced between factor representations. As shown in Fig. 11, both hyperparameters achieve their best overall performance at moderate values. For ρ_{Anti}, applying the anti-collapse loss for an intermediate portion of training provides the best balance between cFID and Precision, while either a shorter or longer duration degrades performance. Similarly, a moderate γ yields a better trade-off between distribution fidelity and Precision. These results indicate that the selected hyperparameter settings provide a stable operating point for LMF.

J Training-Set Evidence for Learned Factors

To examine whether the learned minority factors reflect visual patterns present in the training data, we conduct a feature-space retrieval analysis on Foods. For each factor, we construct a factor prototype from its generated samples and retrieve its nearest real samples from the minority set.

For a given factor k, we independently activate the factor with weight 1.0 and generate samples using all 101 Foods class prompts in the form "a photo of {class name}". We generate three samples per class with seeds 0, 42, and 1024, resulting in $M_{\text{gen}} = 303$ samples for each factor. We extract DINOv2 features from all generated samples, ℓ_2-normalize them, and aggregate them into a factor prototype:

$$\mathbf{p}_k = \frac{\sum_{m=1}^{M_{\text{gen}}} \frac{\phi(\mathbf{x}_m^{(k)})}{\|\phi(\mathbf{x}_m^{(k)})\|_2}}{\left\| \sum_{m=1}^{M_{\text{gen}}} \frac{\phi(\mathbf{x}_m^{(k)})}{\|\phi(\mathbf{x}_m^{(k)})\|_2} \right\|_2}, \tag{18}$$

Table 8: Classification performance across ImageNet and distribution-shifted benchmarks.

Method	IN	Sketch	IN-V2	IN-R	IN-A
SGMS Um & Ye (2024)	70.08	**20.72**	57.57	<u>20.10</u>	<u>0.16</u>
BnS Um et al. (2025)	70.05	<u>20.54</u>	<u>57.62</u>	20.05	0.13
MP Um & Ye (2025)	<u>70.09</u>	20.45	57.53	20.05	0.15
JEPA Park & Um (2026)	70.08	<u>20.54</u>	57.42	20.05	<u>0.16</u>
LMF (Ours)	**70.37**	20.53	**57.70**	**20.15**	**0.20**

where $\phi(\cdot)$ denotes the DINOv2 feature extractor and $\mathbf{x}_m^{(k)}$ denotes a sample generated using factor k. The resulting prototype represents the feature of factor k. We then retrieve the nearest real samples from the minority set. For each real image $\mathbf{x}_n$, we extract and normalize its DINOv2 feature and rank all samples according to cosine similarity, $s_{k,n} = \mathbf{p}_k^\top \phi(\mathbf{x}_n)/\|\phi(\mathbf{x}_n)\|_2$. The highest-ranked samples are used as the nearest real neighbors of each factor prototype.

Fig. 12 shows representative retrieval results. For the factor interpreted as *Buffet Display*, the retrieved real minority samples consistently exhibit buffet-style food displays. Similarly, the factor interpreted as *Circular Presentation* retrieves real samples characterized by circular or radial presentation patterns. The consistent visual patterns between factor-generated samples and retrieved real minority samples provide evidence that the learned factors capture recurring patterns in the target minority distribution, indicating that these factors represent minority-specific visual characteristics.

K Application on Image Classification

Beyond the visuo-motor control evaluation in Sec. 4.5, we further examine the utility of generated minority samples for image classification under both in-distribution and out-of-distribution settings. Specifically, we use the minority samples generated in Sec. 4.3 as additional training data to fine-tune a ResNet-18 classifier, and evaluate it on ImageNet and several distribution-shifted benchmarks.

As shown in Tab. 8, LMF achieves the best accuracy on ImageNet, ImageNet-V2, ImageNet-R, and ImageNet-A, demonstrating that the generated minority samples provide useful complementary supervision for downstream recognition. The overall improvements are relatively modest, which may be attributed to the relatively small amount of generated minority data used for fine-tuning. With more synthetic minority samples, the benefit of minority-data augmentation may become more pronounced. Nevertheless, the consistent gains across multiple benchmarks support the effectiveness of LMF-generated minority samples for downstream image classification.

L More Results of Factor-Level Evaluation

In the main paper, we show that the learned minority factors exhibit coherent semantics, cross-class reusability, continuous controllability, and compositionality. Here, we provide additional qualitative results.

Fig. 13 presents additional results of single-factor-specified minority generation. Each column after the DDIM baseline corresponds to an independently activated factor. Across different food classes, the same factor consistently induces its corresponding visual characteristic, further demonstrating coherent factor semantics and cross-class reusability.

Fig. 14 further evaluates continuous factor control. We activate the learned factor and progressively increase its weight during generation. As the factor strength increases, the corresponding visual characteristic gradually becomes more pronounced, producing a smooth transition toward a closer-view presentation. These results further demonstrate the continuous controllability of the learned factors.

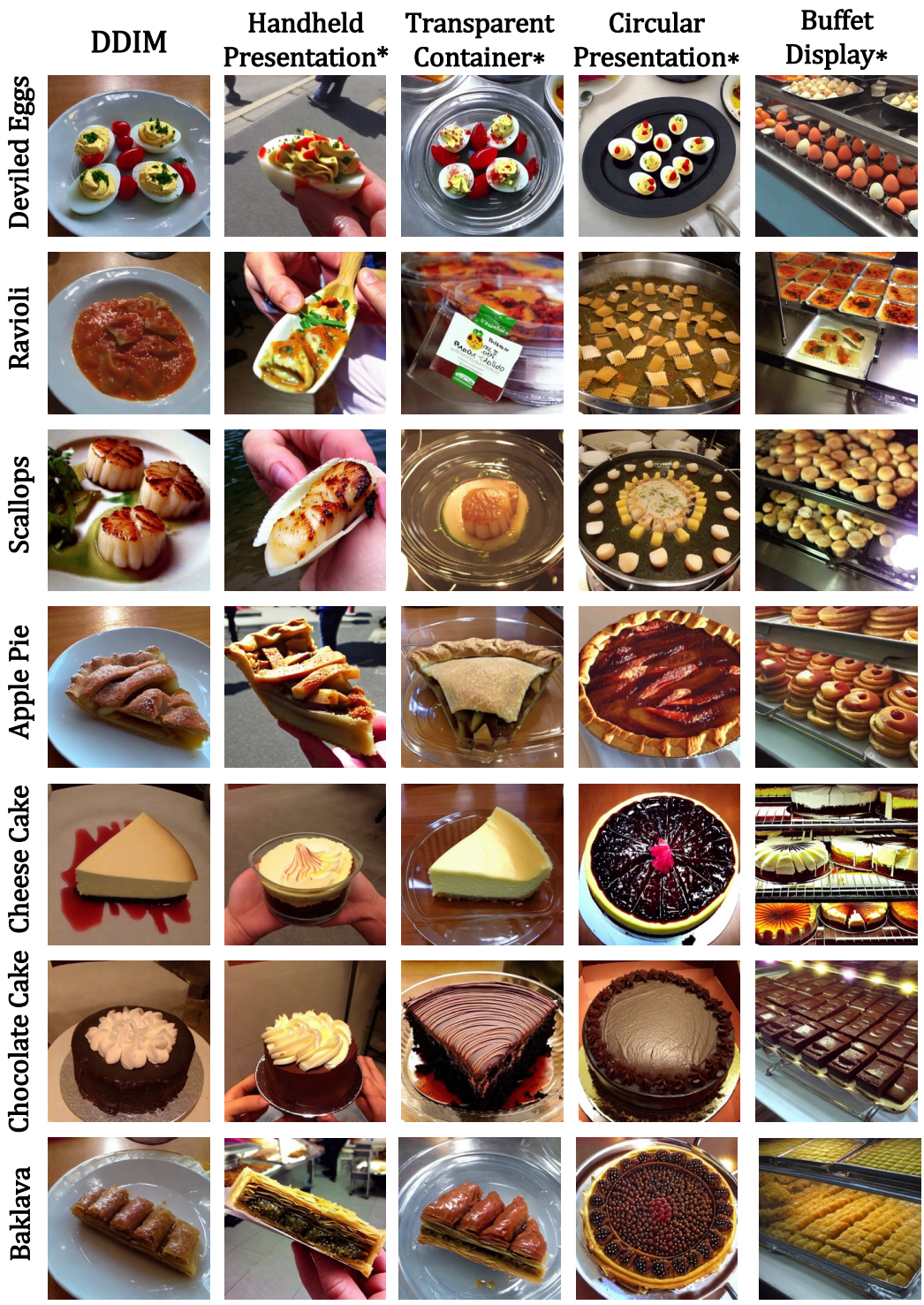


Figure 13: Additional results of single-factor-specified minority generation. Each column after the DDIM corresponds to an independently activated factor, demonstrating coherent semantics and cross-class reusability across different food classes.

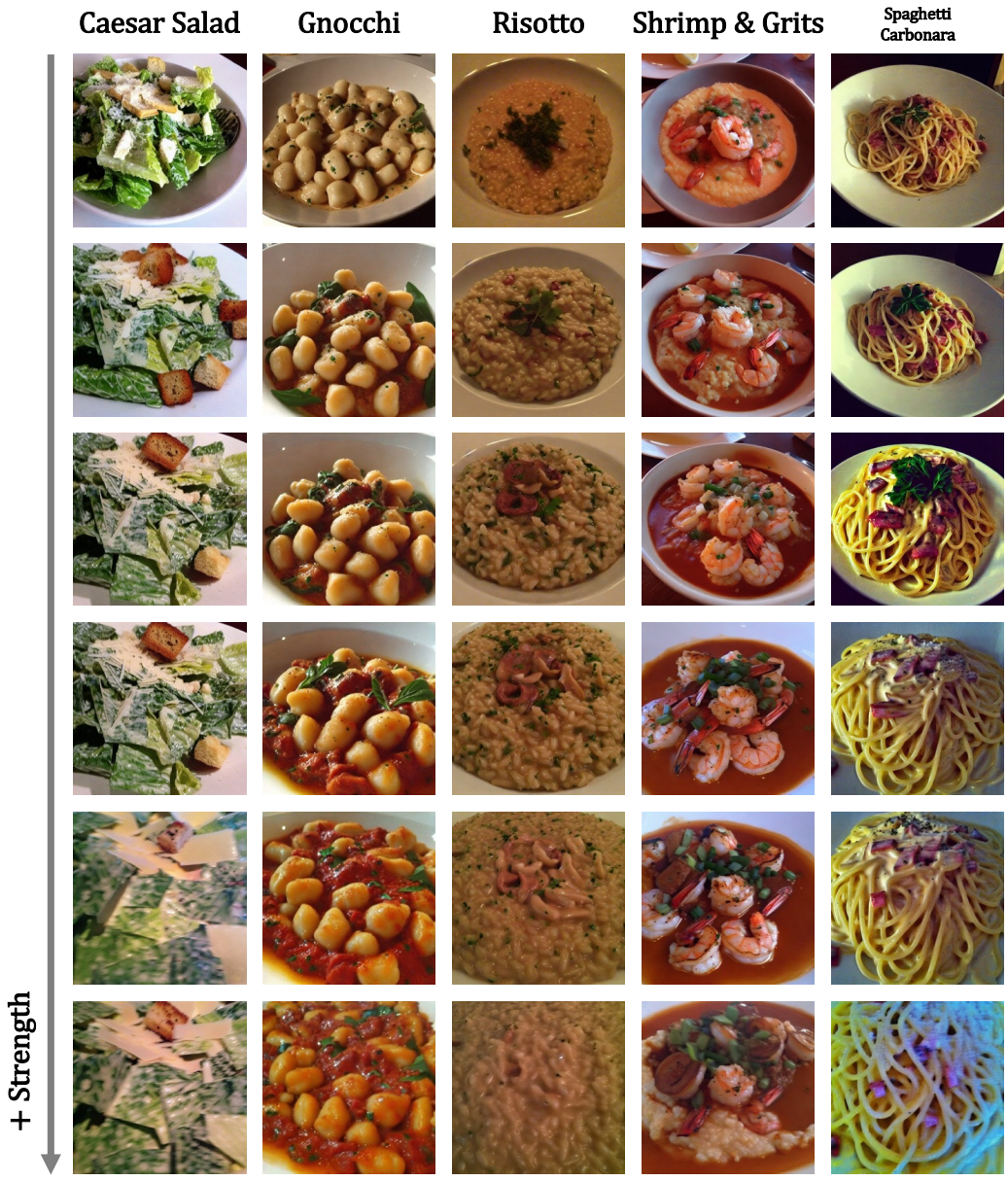


Figure 14: Additional results of continuous factor control. Progressively increasing the weight of the learned factor smoothly strengthens its corresponding closer-view characteristic.