Metric Entropy-Driven Explicit Manifold Modeling for Diffusion Image Generation

Duoduo Xue, *Member, IEEE,* Zhiyu Zhu, Junhui Hou *Senior Member, IEEE*

***Abstract*—Image generative models aim to sample data points from the underlying data manifold, a task that requires learning and decoding a dense, low-dimensional, and compact parameterization space. Existing methods require a large number of parameters to approximate the complex implicit data manifold. In this paper, we propose the Data Manifold-aware Image diffusioN moDel (MIND), a novel image generative framework that explicitly models manifold geometry by integrating discrete patch tokenization into the score function of a continuous diffusion model, which theoretically guarantees learning the score network in a parameterized space with lower metric entropy compared with traditional continuous diffusion models without explicit data manifold. MIND successfully leverages both the structural quantization capabilities of discrete tokens and the parallel generation flexibility of continuous diffusion. Moreover, we enable end-to-end differentiable training via a novel soft top-k aggregation mechanism and introduce dual-branch high-frequency feature embedding layers to alleviate the spectral bias of transformer backbones on low-dimensional inputs. For inference, we design a multi-stage transition sampling scheme that dynamically adjusts the sampling process based on the timestep. Furthermore, we propose a one-step differentiable distillation with the straight-through estimator, which successfully leverages the FD loss to accelerate the discrete image generation. For image generation on ImageNet 256×256 with guidance, the proposed MIND-B-G, with only 130M parameters, achieves an FID of 2.06, surpassing LlamaGen-3B with 3.1B parameters. Furthermore, MIND-XL-G with 715M parameters reduces the FID to 1.84 and one-step distillation using FD loss on the MIND sets a new SOTA FID of 0.9 among discrete image generation methods. Evaluated on the comprehensive FDr^6 metric, our MIND-XL-G achieves a score of 4.87, marking a 42.3% reduction compared to the continuous baseline SiT. Our MIND introduces a fresh perspective on diffusion-based image generation, paving the way for future research and innovation in this community. The code and project page are publicly available at https://github.com/xddgit/MIND and https://xddgit.github.io/MIND/, respectively.**

***Index Terms*—Image generation, Discrete diffusion model, Data manifold, Image tokenizer, Metric entropy**

I. Introduction

DEEP generative models, encompassing generative adversarial networks [1], [2], variational autoencoders [3], and normalizing flows [4], have fundamentally revolutionized image synthesis [4]–[6]. In recent years, continuous diffusion models and score-based generative models [7]–[13], particularly latent diffusion models [14]–[16] and Diffusion Transformers (DiT) [17], [18], have become the standard for high-fidelity image generation. Despite their impressive performance, these models typically operate within unbounded Euclidean spaces with infinite continuous states. While theoretical analyses heavily rely on the low-dimensional manifold hypothesis—which posits that high-dimensional natural images reside on a low-dimensional topological structure [19]–[25], existing continuous diffusion algorithms rarely model this geometric prior explicitly, aside from a few specialized Riemannian diffusion formulations [26], [27]. Consequently, image generation with standard continuous diffusion models is efficient but struggles to leverage the data manifold geometry optimally. Parallel to continuous diffusion models, discrete generation paradigms, e.g., next-token prediction [28], [29] or masked diffusion models [30], [31], compress images into discrete tokens via vector quantization [32]–[36]. Both of the methods (intentionally or unintentionally) set a decoding order onto the image generation process, e.g., the 1D generation order for the next-token prediction or random order for the discrete diffusion, which is constructed based on the assumption of causality among different patches. However, unlike language data, the success of vision diffusion models [8], [17], [30], [37] has demonstrated that high generation fidelity relies on progressively refining each token in parallel, which also applies to consistency-like models [38]. Furthermore, image generation models under the next-token prediction scheme cannot utilize global bidirectional information during the early sampling stages due to the inherent sequential generation scheme [36].

Explicitly constraining continuous diffusion processes onto a low-dimensional geometric manifold is highly appealing: it not only aligns existing theory with the low-dimensional manifold hypothesis, but also restricts score function learning to a compact space with lower metric entropy (as we formally prove in Section IV), thereby fundamentally reducing generative complexity. However, achieving this in continuous diffusion is fundamentally challenging. As demonstrated by representation autoencoders (RAE) [37], low-dimensional latent spaces restrict information capacity, such that continuous diffusion models struggle to operate effectively within compact parameterized spaces and demand high-dimensional representations to achieve high-fidelity generation with a low FID inherently. As demonstrated by numerical experiments, projecting continuous latent features directly onto a strict topological bottleneck (e.g., a low-dimensional hypersphere surface) leads to catastrophic representation collapse. Because

D. Xue and J. Hou are with the Department of Computer Science, City University of Hong Kong, Hong Kong SAR, China (email: duoduxue@cityu.edu.hk;jh.hou@cityu.edu.hk).

Z. Zhu is with City University of Hong Kong (Dongguan), Dongguan, China, and also with the Department of Computer Science, City University of Hong Kong, Hong Kong SAR, China (email: zhiyu.zhu@cityu-dg.edu.cn).

This work was supported in part by the National Natural Science Foundation of China under Grant 62422118, and in part by the Hong Kong Research Grants Council under Grants 11219324, N_CityU1114/25, and R1018-25F

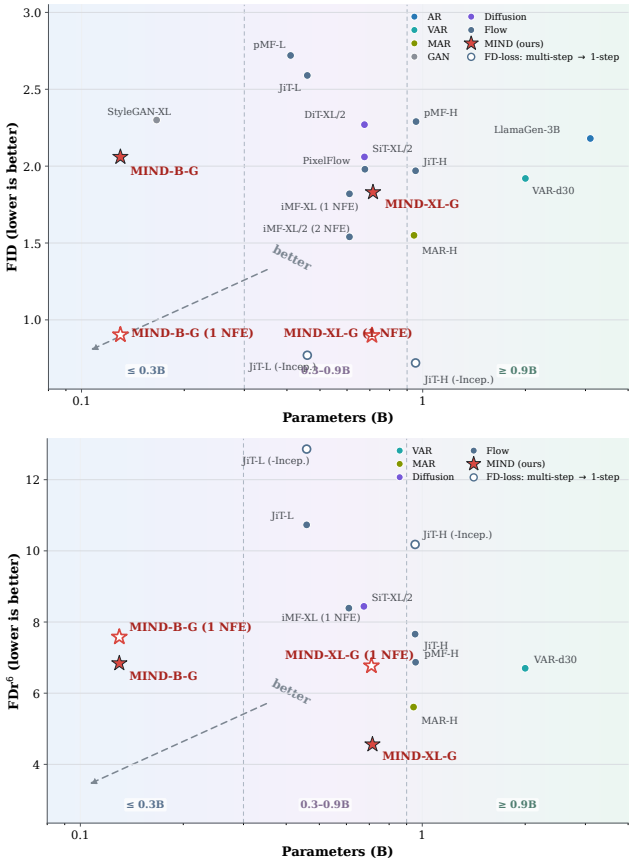


Fig. 1: MIND establishes a state-of-the-art FID of 0.9 among discrete generative models. Compared to the baseline DiT, MIND-B-G achieves better generation quality using $\sim 4\times$ fewer parameters. When evaluated on FD_{r^6}, MIND outperforms standard continuous diffusion and flow-matching baselines, as well as discrete VAR with the same codebook size.

continuous autoencoders rely heavily on vector magnitudes to encode structural information, the strict ℓ_2-normalization inherent to hyperspherical manifolds destroys this information, resulting in severe perceptual distortion. In contrast, discrete token indices encode information categorically. When mapped to a low-dimensional manifold, discrete tokens rely entirely on angular separation rather than magnitude, demonstrating an innate immunity to bottleneck degradation. See more analysis in Section III-A.

A. Contributions

In this paper, we propose a metric-entropy driven generative framework named data Manifold-aware Image diffusioN moDel (MIND), which combines the representational robustness of discrete tokenization with the sampling flexibility of continuous diffusion. MIND models the diffusion process on an explicitly defined data manifold and is theoretically guaranteed to possess lower metric entropy than standard continuous diffusion models. Experimental results demonstrate that the proposed framework achieves state-of-the-art performance in image generation. In summary, our main contributions are threefold:

- Motivated by reducing score function complexity, we introduce a data manifold-aware image diffusion model that explicitly models a hypersphere manifold and theoretically guarantees lower metric entropy than standard continuous diffusion. To enable this explicit modeling and mitigate the representation collapse when projecting continuous variables onto low-dimensional manifolds, we demonstrate the fundamental necessity and superiority of discrete tokenization, establishing a theoretically grounded and robust generative baseline.
- The proposed MIND features a differentiable soft top-k bridge for training and multi-stage transition sampling for inference. Furthermore, we propose a one-step differentiable distillation via a straight-through estimator based on the FD loss.
- Extensive experiments demonstrate that MIND significantly improves generation quality. As shown in Fig. 1, the proposed MIND-B-G with only 130M parameters achieves an FID of 2.06, surpassing LlamaGen with 3B parameters and DiT-XL/2 with 675M parameters. MIND-XL with 715M parameters further reduces the FID to 1.84. The proposed one-step generation method achieves the SOTA FID of 0.9 among discrete image generation methods. Evaluated on the FDr^6 metric, our MIND-XL-G achieves a score of 4.87, marking a 42.3% reduction compared to the baseline SiT.

This manuscript presents substantial extensions and improvements over our earlier conference version [39]. The key additions are as follows:

- We theoretically establish that learning the score network within the formulated hypersphere parameterized space guarantees a lower metric entropy, providing a solid mathematical foundation for reducing the complexity of the score function.
- We accelerate MIND to one-step generation by formulating the first discrete-continuous distillation framework that optimizes the FD loss via a straight-through estimator, yielding superior frequency-domain fidelity compared to continuous one-step baselines optimized under the same objective.
- We extend MIND to text-to-image generation, where it achieves highly competitive performance using only 1.28M training images and 130M parameters. In addition, we evaluate our framework on a more comprehensive metric, FDr^6, to rigorously assess generation quality.

B. Organization

The remainder of the paper is organized as follows. Section II reviews existing works most related to this work. In Section III, we introduce the overall architecture of our MIND, a novel and general image generation method that explicitly models the data manifold to achieve theoretically lower metric entropy as shown in Section IV. In Sections V and VI, we validate the effectiveness of our proposed method on class and text conditioned image generation, respectively. Finally, Section VII concludes this paper.

TABLE I: Summary of variables and their descriptions

Variable	Description
$\boldsymbol{k}$	Discrete tokens
$\boldsymbol{c}$	Condition information
t	Time step in range [0,1]
$\boldsymbol{I}$	Image sampled from data distribution
p_{data}	Data distribution
$\mathcal{P}_{\boldsymbol{\theta}}(\cdot)$	Projection operator
$\boldsymbol{s}_{\boldsymbol{\phi}}(\cdot, t)$	Denoising network
c_1, c_2	Noise and signal coefficients in the forward diffusion process
$\boldsymbol{w}$	Gaussian noise sampled from $\mathcal{N}(\mathbf{0}, \mathbf{I})$
$\mathcal{N}(\mathcal{Q}, \epsilon)$	The covering number of a set $\mathcal{Q}$
$\log \mathcal{N}(\mathcal{Q}, \epsilon)$	The metric entropy of $\mathcal{Q}$
$\mathcal{C}^s(U)$	Hölder space of functions on s-order differentiable manifold U
$\mathcal{C}^s_M(U)$	$\mathcal{C}^s(U)$ with bounded functional norm M
V	Vocabulary size of the tokenizer
L	Dimension of the continuous latent
$\mathcal{K}_\delta$	A compact domain s.t. $\boldsymbol{x}_t \in \mathcal{K}_\delta$ with probability at least $1-\delta$

C. Notations

We use normal symbols for scalars and bold symbols for vectors. $\|\cdot\|_\infty$ refers to the ℓ_∞-norm. Table I summarizes the frequently-used notations.

II. Related Work

A. Diffusion Models

Diffusion models have beaten GAN and achieved great success in image generation recently [7], [11]. Start from image generation in pixel space, such as DDPM [8] and DDIM [9], latent diffusion models train diffusion models in latent space with much lower dimension by encoding images with an autoencoder to get continuous latent and enable high-resolution generation, the neural backbone is implemented with a time-dependent UNet in [14] and a transformer in DiT [17]. A series of works is built on the pioneering work of DiT. Scalable Interpolant Transformers (SiT) [18] improves DiT with an interpolant framework. REPresentation Alignment (REPA) [40] aligns the diffusion model representation with the pretrained visual representation. REPA-E [41] proposes end-to-end training of the variational autoencoder (VAE) and diffusion model based on representation-alignment loss. Representation autoencoders [37] propose to replace the VAE in DiT with the well-developed and pretrained representation encoders to obtain a semantically rich latent space. Riemannian flow matching with Jacobi regularization [27] proposes performing diffusion in the feature space of representation learning and constraining the generative process to manifold geodesics. Deco [42] proposes to decouple the generation by generating high-frequency details and low-frequency semantics in pixel and latent spaces, respectively. To improve the efficiency of diffusion models, several works explore one-step or few-step generation under the frameworks of consistency model [38], flow-matching [43]–[45], and distillation [46].

Conditional diffusion models on class labels are crucial to further improve the generation quality. Class information is incorporated into adaptive group normalization layers together with the time step in [11]. Dhariwal *et al.* [11] proposed to train a classifier on noisy images and then use the gradient of the trained classifier to guide the generation model. The widely used classifier-free guidance [47] is proposed to avoid an extra classifier, which trains a single network to parameterize both the unconditional and conditional models and samples with the linear combination of the conditional and unconditional score estimations. Rombach *et al.* [14] proposed flexible conditional image generation by mapping the embedded class label to the intermediate layers of UNet with a cross-attention layer. Karras *et al.* proposed autoguidance that guides the model with an inferior version of itself [48] rather than an unconditional model as in [47], which performs better but needs to train an additional guiding model. However, the diffusion models operator in continuous space with infinite states and cannot incorporate the prior information of the data manifold.

B. Image Generation with Language Models

In parallel with diffusion models, popular language models, such as autoregressive language models, have been extended to image generation since the great success of large language models in natural language generation. Images are first tokenized into discrete tokens $\boldsymbol{k}$ using a well-developed visual tokenizer [49]. The autoregressive language models [28] predict the next token $\boldsymbol{k}_i$ using previous tokens $\boldsymbol{k}_1, \boldsymbol{k}_2, \cdots, \boldsymbol{k}_{i-1}$ and conditional information $\boldsymbol{c}$. The training stage learns the categorical distribution $p_{\boldsymbol{\theta}}(\boldsymbol{k}_i|\boldsymbol{k}_{i-1}, \boldsymbol{c})$ by minimizing the negative log-likelihood $\mathcal{L}(\boldsymbol{\theta}) = -\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{k}_i|\boldsymbol{k}_1, \boldsymbol{k}_2, \cdots, \boldsymbol{k}_{i-1}; \boldsymbol{c})$, where $\boldsymbol{\theta}$ is the model parameters. The inference stage generates discrete tokens sequentially based on next-token prediction. The masked language models [30] learns the distribution $p_{\boldsymbol{\theta}}(\boldsymbol{k}_i|\boldsymbol{k}_j, \boldsymbol{c})$ in the training stage, where mask $\boldsymbol{m} \in \{0,1\}, m_j = 1, \forall j, m_i = 0, \forall i$. Starts with a fully masked sequence, the inference stage alternatively samples the whole sequence with the given non-masked tokens and remasks the tokens with the lowest probability in decreasing mask ratio. eMIGM (effective and efficient Masked Image Generation Models) [31] unifies the masked diffusion models [50], [51] and masked image generation [30], which systematically explores the design space of masked image generation models. More recently, the diffusion language models have combined the advantages of diffusion models and language generation, which can be generalized to image generation directly [52], [53] by tokenizing images into discrete tokens. However, these language models restrict the whole generation process to a finite discrete space and limit the generation ability.

Several works try to combine the advantages of diffusion models and language models. The Riemannian diffusion language model [54] proposes a continuous diffusion model for language modeling and performs the diffusion process on the hypersphere surface, which is extended to image generation in the feature space of representation learning [27]. However, these methods cannot leverage the mature generation tools in Euclidean space and require complex forward and reverse diffusion processes. Note that our MIND is *fundamentally different* from RDLM [54] and RJF [27]: ***Motivation***: RJF aids convergence in high-dimensional representation spaces, whereas we embed a low-dimensional manifold prior. ***Input space***: RJF is purely continuous; we process both discrete tokens and continuous latents. ***Trajectory***: RDLM and RJF require *complex, approximated formulations* to strictly constrain

trajectories to the hypersphere, while we allow *off-sphere and exact* trajectories, only requiring data points on the surface.

C. Image Tokenization

Image tokenization is the first step for image generation with autoregressive and masked language models. Images can be represented as discrete variables at the pixel level, but this is redundant and has a quadratically increasing computational cost. Image tokenizers compress images into discrete tokens. The general pipeline maps the image to a latent feature map and then quantizes the continuous features as discrete codes. A representative work is VQ-GAN [32] that constructs a learned codebook and finds the nearest codebook entry for each spatial feature vector. VQ-GAN is built on the grounding study named VQ-VAE [33] and introduces discriminator loss into the training objective. A series of works have been proposed to improve the initialization rate and size of the codebook. VQGAN-LC [34] initializes the codebook entry with the feature of the pretrained vision encoder, then keeps the static codebook and optimizes a projector. The Index Backpropagation Quantization (IBQ) updates all codebook embeddings leveraging a soft one-hot categorical distribution [35]. The scale-based tokenizer named VAR proposes a multi-scale VQ autoencoder that quantizes residuals of feature maps recursively [55]. Recently, visual language models have also been incorporated into image tokenizers to extract semantically rich features [56]. Above discrete tokenizers relying on learnable codebooks suffer from constrained codebook capacity due to the severe computational and memory overhead of nearest-neighbor lookups. To break this capacity bottleneck, recent literature has explored lookup-free quantizers, such as finite scale quantization [57], lookup free quantization [49], and binary scale quantization [58]. By projecting continuous latent features directly onto hyper-dimensional bounded grids or binary hypercube vertices, these methods bypass explicit codebook learning and have higher codebook capacity. A complementary survey of image tokenizers can be referred to [36].

III. Proposed Method

The low-dimensional manifold hypothesis is widely adopted in the theoretical analysis of image generation with diffusion models [19], [21]–[25], but has not been explicitly utilized in modern generative models. In this paper, we leverage image tokenization as the quantification measurement of image patch manifold structure and anchor those discrete points onto a simple and tackleable geometry, as illustrated in Fig. 2.

A. Effectively Parameterizing the Image Manifold Geometry

Mapping images or their continuous latent representations onto a compact topological manifold presents a fundamental challenge, as capturing diverse visual information requires highly complex features. Furthermore, facilitating an effective generation process necessitates that real data samples densely populate the parameterization space. This strict high-density requirement severely exacerbates the difficulty of the learning objective.

To mitigate these challenges, we employ discrete tokenization to quantize the manifold structure, an approach that significantly reduces the reparameterization burden associated with continuous latent spaces. To validate this advantage, we empirically contrast two distinct methodologies: a *continuous projection* that maps patchified VAE latents directly onto an L-dimensional hypersphere (where the exceptionally low dimensionality enforces a high-density space), and a *discrete approach* that projects VQ token indices into the identical manifold.

For a rigorous and fair comparison, both methods are restricted to linear/MLP layers with the same parameter level and trained on the same randomly selected image, with carefully controlled optimization and architectural conditions (detailed in the *Supplementary material*). By decoding the reconstructed bottleneck features back into the RGB domain, we measure the Learned Perceptual Image Patch Similarity (LPIPS) [59], as illustrated in Fig. 3. We experiment with our actual operating range in this paper, our results indicate that the continuous latent formulation suffers from representation collapse due to its heavy reliance on feature magnitude to encode structural information. In contrast, the discrete token path encodes information categorically and relies entirely on angular separation. This grants it an innate robustness against the magnitude-destructive bottleneck, consistently yielding significantly lower perceptual distortion and demonstrating the inherent superiority of discrete tokens for manifold parameterization.

B. Diffusion with Explicit Modeling of Data Manifold

1) Formulation of Diffusion Process: Given an image $\boldsymbol{I} \in \mathbb{R}^{C\times H\times W}$ sampled from data distribution p_{data}, it is tokenized as discrete token $\boldsymbol{k} \in \mathbb{R}^N$ using a tokenizer with vocabulary size V. The token $\boldsymbol{k} \in \mathbb{R}^N$ is then mapped to the continuous latent feature by a projection operator $\mathcal{P}_{\boldsymbol{\theta}}(\cdot)$ parameterized by $\boldsymbol{\theta}$ to get a continuous representation of dimension L $\boldsymbol{x}_0 \in \mathbb{R}^{N\times L}$ on the hypersphere surface satisfying $\sum_{l=0}^{L-1} \boldsymbol{x}_0^2(n,l) = R^2$ for all $n = 0, \cdots, N-1$.

The forward and reverse diffusion processes are performed based on the hypersphere surface with radius R. The forward diffusion process starts from the continuous representation $\boldsymbol{x}_0 \in \mathbb{R}^{N\times L}$ and perturbs it to get noisy latent,

$$\boldsymbol{x}_t = c_1\sqrt{t}\cdot\boldsymbol{w} + c_2\sqrt{1-t}\cdot\boldsymbol{x}_0, \tag{1}$$

where $\boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $t \in [0,1]$, and the end of the forward diffusion process is the Gaussian noise $\boldsymbol{x}_1 = c_1\boldsymbol{w}$. The reverse diffusion process starts from sampled Gaussian noise $\boldsymbol{x}_1 \sim \mathcal{N}(\mathbf{0}, c_1^2\mathbf{I})$. The network $\boldsymbol{s}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \boldsymbol{c})$ takes the noisy latent $\boldsymbol{x}_t$ as input and predicts the denoised logits $\tilde{\boldsymbol{x}}_{0t} = \boldsymbol{s}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \boldsymbol{c}) \in \mathbb{R}^{N\times V}$, where $\boldsymbol{c}$ denotes an optional conditioning signal (e.g., a class label or text prompt). The denoised latent $\dot{\boldsymbol{x}}_{0t}$ is predicted by sampling discrete tokens from $\tilde{\boldsymbol{x}}_{0t}$ and then mapping the discrete tokens to continuous latent.

2) Training Pipeline: During the training stage, to maintain the differentiability of the discrete token sampling process while ensuring the output strictly resides on the target hypersphere manifold, we introduce a soft sampling mechanism.

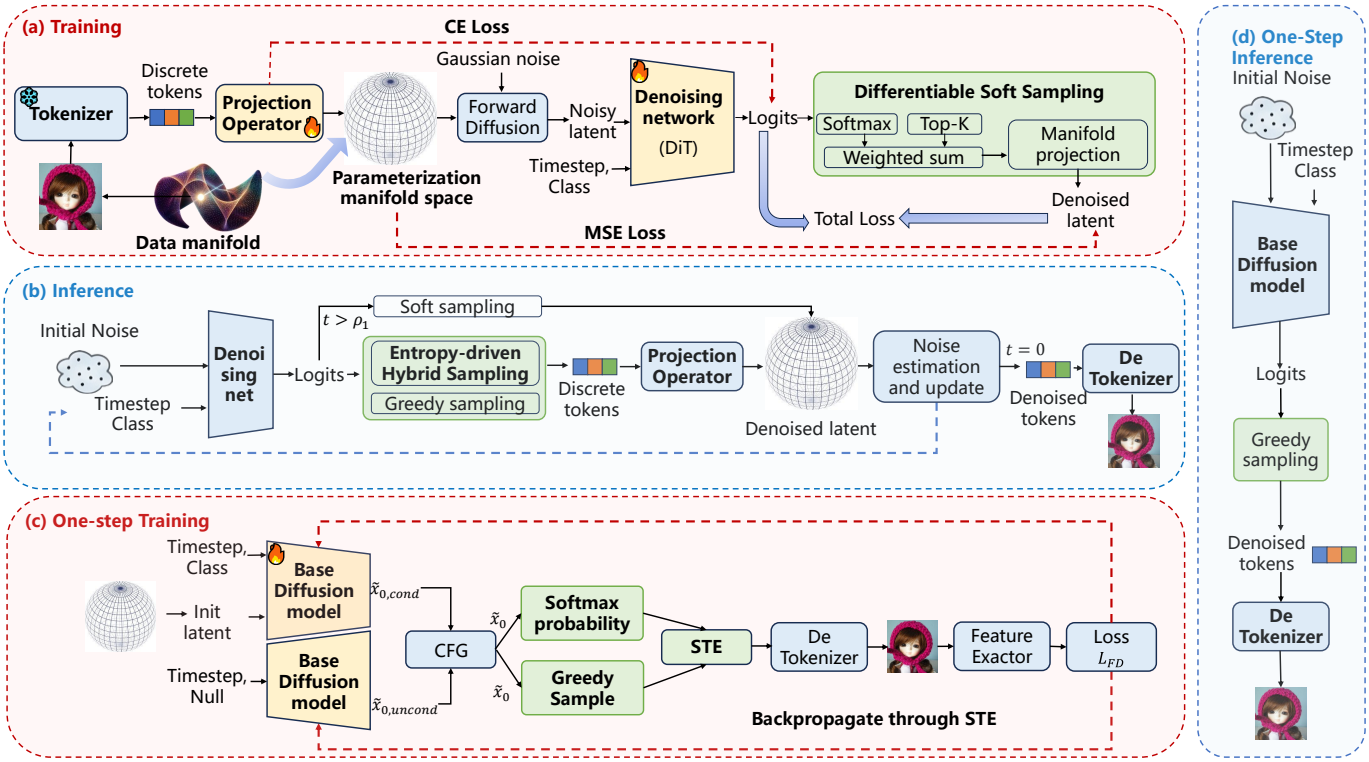
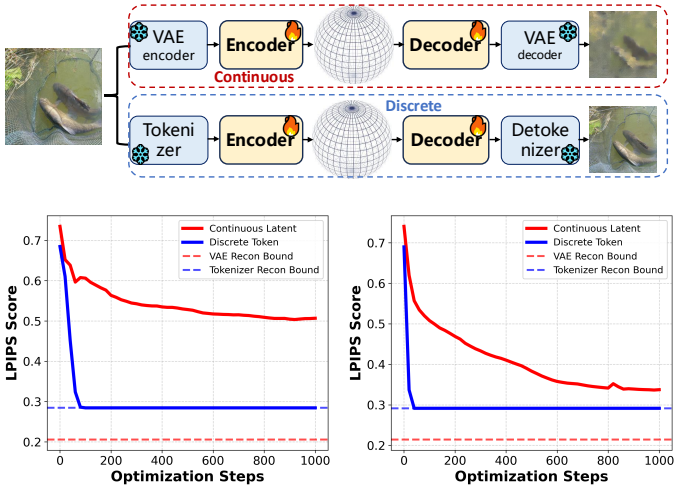


Fig. 2: **Overview of the proposed MIND framework. (a)–(b)** The training and inference pipelines of MIND, featuring explicit modeling of data manifold geometry. **(c)–(d)** The training and inference pipelines for one-step generation, enabled by differentiable distillation via a straight-through estimator (STE).



(a) Dim=8 (MIND-S) (b) Dim=16 (MIND-B)

Fig. 3: Illustration of continuous and discrete projection. The plot shows the LPIPS distance between the reconstructed and original images during optimization. Solid lines represent the models with manifold constraints, whereas dashed lines indicate the unconstrained reconstruction.

Let $\tilde{\boldsymbol{x}}_{0t}(n) \in \mathbb{R}^V$ denote the predicted logits of the n-th token over the vocabulary of size V. We first extract the set of indices corresponding to the top-k logit values, denoted as $\mathcal{I}_k$. The normalized soft weights α_i are computed via the softmax function over these selected elements:

$$\alpha_i = \exp(x_i) / \sum_{j \in \mathcal{I}_k} \exp(x_j), \forall i \in \mathcal{I}_k, \tag{2}$$

where x_i is the i-th element of $\tilde{\boldsymbol{x}}_{0t}(n)$. Subsequently, we aggregate the corresponding continuous latent using the weights α_i. Since the convex combination of points on a hypersphere falls into the interior of the sphere, we explicitly project the aggregated feature onto the hypersphere surface via ℓ_2-normalization:

$$\dot{\boldsymbol{x}}_{0t}(n) = \frac{\sum_{i \in \mathcal{I}_k} \alpha_i \mathcal{P}_{\boldsymbol{\theta}}(i)}{\left\| \sum_{i \in \mathcal{I}_k} \alpha_i \mathcal{P}_{\boldsymbol{\theta}}(i) \right\|_2}, \tag{3}$$

where $\dot{\boldsymbol{x}}_{0t} \in \mathbb{R}^{N \times L}$ is the denoised continuous latent at timestep t, seamlessly bridging the discrete token space and the continuous diffusion manifold.

Based on the diffusion process formulated with the explicitly modeled hypersphere manifold prior, the neural network $\boldsymbol{s}_{\boldsymbol{\phi}}$ is optimized by minimizing the cross-entropy (CE) loss between denoised logits $\tilde{\boldsymbol{x}}_{0t}$ and the discrete tokens $\boldsymbol{k}$ and the mean squared error (MSE) loss between $\boldsymbol{x}_0$ and the denoised latent $\dot{\boldsymbol{x}}_{0t}$:

$$\mathcal{L}(\boldsymbol{s}_{\boldsymbol{\phi}}, \mathcal{P}_{\boldsymbol{\theta}}) = \mathbb{E}_{\boldsymbol{k} \sim p_{data}, t \in [0,1]} \{ \mathrm{CE}(\tilde{\boldsymbol{x}}_{0t}, \boldsymbol{k}) + \lambda \cdot \mathrm{MSE}(\dot{\boldsymbol{x}}_{0t}, \boldsymbol{x}_0) \}, \tag{4}$$

where $\tilde{\boldsymbol{x}}_{0t} = \boldsymbol{s}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t, \boldsymbol{c})$, $\boldsymbol{x}_t = c_1 \sqrt{t} \cdot \boldsymbol{w} + c_2 \sqrt{1-t} \cdot \boldsymbol{x}_0$, $\boldsymbol{x}_0 = \mathcal{P}_{\boldsymbol{\theta}}(\boldsymbol{k})$, λ controls the strength of MSE loss. To support Classifier-Free Guidance (CFG) during inference, the condition $\boldsymbol{c}$ is randomly replaced with a null condition $\emptyset$ with

Algorithm 1 Training Process of MIND

Input: Dataset p_{data}, pre-trained Tokenizer, initialized networks $\boldsymbol{s_\phi}$ and $\mathcal{P}_{\boldsymbol{\theta}}$, noise schedules c_1, c_2, timestep range $[t_{min}, t_{max}]$, condition $\boldsymbol{c}$.
Output: Trained networks $\boldsymbol{s_\phi}$ and $\mathcal{P}_{\boldsymbol{\theta}}$.

repeat
  Sample image $\boldsymbol{I} \sim p_{data}$;
  Extract discrete tokens: $\boldsymbol{k} = \text{Tokenizer}(\boldsymbol{I})$;
  Project to hypersphere manifold: $\boldsymbol{x}_0 = \mathcal{P}_{\boldsymbol{\theta}}(\boldsymbol{k})$;
  Sample timestep $t \sim \mathcal{U}[t_{min}, t_{max}]$ and Gaussian noise $\boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
  Compute noisy latent: $\boldsymbol{x}_t = c_1\sqrt{t} \cdot \boldsymbol{w} + c_2\sqrt{1-t} \cdot \boldsymbol{x}_0$;
  Predict unnormalized logits: $\tilde{\boldsymbol{x}}_{0t} = \boldsymbol{s_\phi}(\boldsymbol{x}_t, t, \boldsymbol{c})$;
  Obtain denoised latent $\dot{\boldsymbol{x}}_{0t}$ via soft sampling over $\tilde{\boldsymbol{x}}_{0t}$;
  Compute $\mathcal{L}(\boldsymbol{s_\phi}, \mathcal{P}_{\boldsymbol{\theta}}) = \text{CE}(\tilde{\boldsymbol{x}}_{0t}, \boldsymbol{k}) + \lambda \cdot \text{MSE}(\dot{\boldsymbol{x}}_{0t}, \boldsymbol{x}_0)$;
  Take gradient descent step on $\nabla_{\boldsymbol{\theta},\boldsymbol{\phi}}\mathcal{L}$ and update $\boldsymbol{\theta}, \boldsymbol{\phi}$;
until network converges

Algorithm 2 Inference Process of MIND

Input: Trained networks $\boldsymbol{s_\phi}$,$\mathcal{P}_{\boldsymbol{\theta}}$, parameters c_1, c_2, variance coefficient $\eta \in [0,1]$, thresholds ρ_1,ρ_2, CFG scale cfg, condition $\boldsymbol{c}$, sampling timestep schedule.
Output: Generated image DeTokenizer($\hat{\boldsymbol{k}}_0$).

Sample initial noise: $\boldsymbol{x}_1 \sim \mathcal{N}(\mathbf{0}, c_1^2\mathbf{I})$;
for $t = 1$ **to** 0 **do**
  if $cfg = 1$ **then**
    Predict logits: $\tilde{\boldsymbol{x}}_{0t} = \boldsymbol{s_\phi}(\boldsymbol{x}_t, t, \boldsymbol{c})$;
  else
    Predict unconditional logits: $\tilde{\boldsymbol{x}}_{0t} = \boldsymbol{s_\phi}(\boldsymbol{x}_t, t, \emptyset)$
    Predict conditional logits: $\tilde{\boldsymbol{x}}^c_{0t} = \boldsymbol{s_\phi}(\boldsymbol{x}_t, t, \boldsymbol{c})$;
    $\tilde{\boldsymbol{x}}_{0t} = \tilde{\boldsymbol{x}}_{0t} + cfg * (\tilde{\boldsymbol{x}}^c_{0t} - \tilde{\boldsymbol{x}}_{0t})$
  end if
  if $t > \rho_1$ **then**
    $\dot{\boldsymbol{x}}_{0t} \leftarrow$ Soft sampling via Eq. (3);
  else
    Sample $\hat{\boldsymbol{k}}_t = \text{S}(\tilde{\boldsymbol{x}}_{0t})$ with Algorithm 3;
    Project back to hypersphere manifold: $\dot{\boldsymbol{x}}_{0t} = \mathcal{P}_{\boldsymbol{\theta}}(\hat{\boldsymbol{k}}_t)$;
  end if
  Estimate noise $\boldsymbol{w}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t) = \frac{\boldsymbol{x}_t - c_2\sqrt{1-t}\cdot\dot{\boldsymbol{x}}_{0t}}{c_1\sqrt{t}}$;
  Sample stochastic noise $\boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
  Compute latent for the next timestep via Eq. (5).
end for
Return DeTokenizer($\hat{\boldsymbol{k}}_0$).

probability p_{uncond} during training. Algorithm 1 demonstrates the training pipeline of the proposed method.

3) Inference Pipeline: During the inference stage, the denoised logits are sampled to obtain the denoised latent $\dot{\boldsymbol{x}}_{0t} = \mathcal{P}_{\boldsymbol{\theta}}(\text{S}(\tilde{\boldsymbol{x}}_{0t}))$. To ensure that the generated token sequences remain in the high-probability manifold of the codebook space while preserving diversity in the early stage of the generation, we propose a parameterized multi-stage transition sampling strategy with thresholds ρ_1 and ρ_2. In the initial phase ($t > \rho_1$), the operator $\mathcal{P}(\text{S}(\cdot))$ inherits the soft sampling mechanism from the training stage in Eq. (3). During the middle phase ($\rho_2 < t \leq \rho_1$), the operator $\text{S}(\cdot)$ is implemented by an

Algorithm 3 Multi-Stage Transition Sampling Operator $\text{S}(\cdot)$

Input: Logits $\tilde{\boldsymbol{x}}_{0t}$, $t \in [0, 1]$, thresholds ρ_1, ρ_2, params p, k, base temperature τ, use_entropy $\in$ {True,False}.
Output: Sampled token $\hat{\boldsymbol{k}}_t$.
if $\rho_2 < t \leq \rho_1$ **then**
  $\boldsymbol{p} \leftarrow \text{Softmax}(\tilde{\boldsymbol{x}}_{0t})$;
  if use_entropy **then**
    {**Entropy-Driven Adaptive Temperature**}
    $H \leftarrow -\sum \boldsymbol{p} \log \boldsymbol{p}$;
    $\tau_{adj} \leftarrow \tau \cdot (2.5 \cdot e^{-H/3} + 0.6)$;
  else
    {**Standard Base Temperature**}
    $\tau_{adj} \leftarrow \tau$;
  end if
  $\mathcal{V} \leftarrow \text{TopK}(\tilde{\boldsymbol{x}}_{0t}/\tau_{adj}, k) \cap \text{Nucleus}(\boldsymbol{p}, p)$;
  $\hat{\boldsymbol{k}}_t \sim \text{Multinomial}\left(\text{Softmax}(\tilde{\boldsymbol{x}}_{0t}/\tau_{adj}|\mathcal{V})\right)$;
else if $t \leq \rho_2$ **then**
  Greedy sampling: $\hat{\boldsymbol{k}}_t \leftarrow \arg\max(\tilde{\boldsymbol{x}}_{0t})$;
end if
return $\hat{\boldsymbol{k}}_t$.

entropy-aware discrete hybrid-filtering scheme. For the given distribution $\boldsymbol{p} = \text{Softmax}(\tilde{\boldsymbol{x}}_{0t})$, we compute the Shannon entropy H. The sampling temperature is scaled adaptively: $\tau_{adj} = \tau \cdot (2.5e^{-H/3} + 0.6)$ following [60], forcing the model to perform confident sampling when the predictive entropy is high. The entropy-driven adaptive temperature is optional in practice. We then apply a hybrid filter combining Top-k and Nucleus (Top-p) constraints, drawing the discrete token from the renormalized distribution. In the terminal phase ($t \leq \rho_2$), the operator $\text{S}(\cdot)$ collapses to greedy sampling to eliminate stochasticity and ensure maximal local consistency. The detailed sampling scheme is summarized in Algorithm 3. The denoised latent $\dot{\boldsymbol{x}}_{0t} \in \mathbb{R}^{N\times L}$ is perturbed to get the input of the next timestep,

$$\begin{aligned}\boldsymbol{x}_{t-\Delta t} = c_2\sqrt{1-(t-\Delta t)} \cdot \dot{\boldsymbol{x}}_{0t} + \\ c_1\sqrt{t-\Delta t} \cdot (\eta \boldsymbol{w}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t) + (1-\eta)\boldsymbol{w}),\end{aligned} \tag{5}$$

where $\boldsymbol{w}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t) = (\boldsymbol{x}_t - c_2\sqrt{1-t} \cdot \dot{\boldsymbol{x}}_{0t})/(c_1\sqrt{t})$, $\eta \in [0, 1]$. Algorithm 2 demonstrates the inference pipeline of the proposed method.

4) Network Architecture: The projector operator $\mathcal{P}_{\boldsymbol{\theta}}$ extracts the continuous latent $\boldsymbol{x}_0 \in \mathbb{R}^{N\times L}$ via an embedding layer, where each L-dimensional vector is factored into d_{sub}-dimensional hyperspherical subspaces through sub-vector normalization. The geometrically constrained $\boldsymbol{x}_0$ is then perturbed to yield the noisy latent $\boldsymbol{x}_t$. The denoising network $\boldsymbol{s_\phi}$ is implemented based on the DiT. Before feeding into the DiT backbone, the continuous latent is processed through two branches named the base high-frequency mapper and the residual sinusoidal projection module. The high-frequency mapper maps the input $\boldsymbol{x}_t$ using a learnable projection matrix $\boldsymbol{B} \in \mathbb{R}^{L\times d_{map}}$, which is initialized from a Gaussian distribution $\mathcal{N}(0, \sigma^2)$ with a large variance to amplify high-frequency signals. The projected features are then transformed by sine

and cosine functions,

$$\boldsymbol{f}_{base} = [\sin(2\pi \boldsymbol{x}_t \boldsymbol{B}), \cos(2\pi \boldsymbol{x}_t \boldsymbol{B})]\,. \tag{6}$$

The concatenated features $\boldsymbol{f}_{base} \in \mathbb{R}^{2d_{map}}$ are subsequently passed through a shallow Multi-Layer Perceptron (MLP) with Layer Normalization (LN) and GELU activations to yield the base hidden states $\boldsymbol{h}_{base} \in \mathbb{R}^{d_{hidden}}$, matching the hidden dimension of the DiT backbone. The residual sinusoidal projection module expands the continuous latent $\boldsymbol{x}_t$ using a deterministic sinusoidal positional encoding to get a high-dimensional vector $\boldsymbol{f}_{freq}$, which is passed through an MLP equipped with LN and SiLU activations. Crucially, to ensure stable gradient flow and prevent catastrophic divergence at initialization, the weight matrix of the final linear layer in this residual MLP is strictly initialized to zero. Finally, the outputs from both branches are aggregated via a residual addition $\boldsymbol{h}_{input} = \boldsymbol{h}_{base} + \boldsymbol{h}_{res}$ before interacting with the timestep and class conditions. The backbone network utilizes the transformer-based diffusion architecture in DiT, which scales across multiple model capacities. Conditioning is handled via an adaptive LayerNorm that injects combined timestep and class label representations into each Transformer block, while spatial dependencies are modeled using rotary positional embeddings.

C. One-Step Distillation via Straight-Through Estimator

This section proposes a one-step differentiable distillation framework for our MIND based on the FD loss [61]. Unlike iterative sampling, where the model can progressively refine latent states through stochastic exploration, one-step generation demands an immediate and deterministic mapping from the initial prior to the target data distribution. To this end, we introduce a continuous-to-discrete optimization pipeline utilizing the straight-through estimator.

Specifically, given an initial state $\boldsymbol{x}_1$, the model executes a single forward pass incorporating classifier-free guidance. Let $\tilde{\boldsymbol{x}}_0 \in \mathbb{R}^V$ denote the predicted token logits. To bridge the gap between the continuous gradient flow required for backpropagation and the discrete token selection required for accurate pixel-level decoding, we perform continuous-to-discrete optimization directly in the probability space rather than truncating the embeddings. Given the predicted logits $\tilde{\boldsymbol{x}}_0 \in \mathbb{R}^V$, where V is the vocabulary size, we first compute the soft probability distribution over the vocabulary

$$\boldsymbol{P}^{soft} = \mathrm{Softmax}(\tilde{\boldsymbol{x}}_0/\tau), \tag{7}$$

where τ is a scaling temperature. To ensure the spatial decoder receives the exact hard-quantized tokens (mimicking inference behavior precisely) while preserving the gradient flow, we enforce a differentiable argmax via the straight-through estimator (STE) applied directly to the one-hot representations,

$$\boldsymbol{P}^{ste} = \mathrm{OneHot}(\mathrm{argmax}(\tilde{\boldsymbol{x}}_0)) + \boldsymbol{P}^{soft} - \mathcal{SG}(\boldsymbol{P}^{soft}), \tag{8}$$

where $\mathcal{SG}(\cdot)$ is the stop-gradient operator. The continuous latent vectors required by the spatial decoder are then obtained by projecting the STE-modified probabilities through the frozen codebook matrix $\boldsymbol{W}_{codebook}$ of the tokenizer,

$$\boldsymbol{z}_q = \boldsymbol{P}^{ste} \boldsymbol{W}_{codebook}. \tag{9}$$

This formulation allows the perceptual gradients from the Fréchet Distance (FD) loss [61] to flow backward seamlessly from the reconstructed pixels, through the codebook mapping $\boldsymbol{z}_q$, and back to the continuous soft probabilities $\boldsymbol{P}^{soft}$ to update the backbone weights $\boldsymbol{\phi}$. To ensure stable estimation of the generated data distribution without incurring prohibitive memory overhead, we adopt the memory-efficient feature queue proposed in [61]. Specifically, a First-In-First-Out buffer is maintained to store detached historical features. During each forward pass, the statistics $\boldsymbol{\mu}_{gen}$ and $\boldsymbol{\Sigma}_{gen}$ are dynamically approximated by online accumulating the detached historical buffer with the current synthesized batch. This strategy ensures continuous gradient flow through the current batch while leveraging a sufficiently large sample size for stable Fréchet Distance evaluation. Once the differentiable statistics $\boldsymbol{\mu}_{gen}$ and $\boldsymbol{\Sigma}_{gen}$ are approximated, the raw Fréchet Distance d^2_{FD} against the reference distribution $(\boldsymbol{\mu}_{ref}, \boldsymbol{\Sigma}_{ref})$ is computed as:

$$d^2_{FD} = \|\boldsymbol{\mu}_{ref} - \boldsymbol{\mu}_{gen}\|_2^2 + \mathrm{Tr}\left(\boldsymbol{\Sigma}_{\text{ref}} + \boldsymbol{\Sigma}_{gen} - 2\left(\boldsymbol{\Sigma}_{ref}\boldsymbol{\Sigma}_{gen}\right)^{1/2}\right), \tag{10}$$

where $\mathrm{Tr}(\cdot)$ denotes the matrix trace, and $\|\cdot\|_2^2$ represents the ℓ_2-norm. To prevent numerical instability and gradient explosion during optimization, the raw distance is dynamically normalized and clamped via a straight-through estimator:

$$\mathcal{L}_{FD} = \frac{\max(0, d^2_{FD})}{\mathcal{SG}(\max(\epsilon_{floor}, d^2_{FD})) + \epsilon_{norm}}, \tag{11}$$

where $\epsilon_{floor}, \epsilon_{norm}$ are small constants to ensure denominator stability. The complete one-step optimization process is summarized in Algorithm 4.

Algorithm 4 One-Step Differentiable Distillation with STE

Input: Pretrained base diffusion model $\boldsymbol{s}_{\boldsymbol{\phi}}$, frozen $\mathcal{P}_{\boldsymbol{\theta}}$, de-tokenizer $\mathcal{D}$, image feature extractor Φ, reference FD statistics μ_{ref}, Σ_{ref}, CFG scale cfg, temperature τ, $t_\epsilon \to 1$.

Output: Distilled model parameters $\boldsymbol{\phi}^*$

while not converged **do**
  Sample class labels $\boldsymbol{c}$ and initialize latent $\boldsymbol{x}_1$.
  $\tilde{\boldsymbol{x}}_{0,cond} \leftarrow \boldsymbol{s}_{\boldsymbol{\phi}}(\boldsymbol{x}_0, t_\epsilon, \boldsymbol{c})$
  $\tilde{\boldsymbol{x}}_{0,uncond} \leftarrow \boldsymbol{s}_{\boldsymbol{\phi}}(\boldsymbol{x}_0, t_\epsilon, \emptyset)$
  $\tilde{\boldsymbol{x}}_0 \leftarrow \tilde{\boldsymbol{x}}_{0,uncond} + cfg \cdot (\tilde{\boldsymbol{x}}_{0,cond} - \tilde{\boldsymbol{x}}_{0,uncond})$
  STE based on (8) and compute $\boldsymbol{z}_q$ based on Eq. (9).
  Differentiable decoder: $I_{gen} \leftarrow \mathcal{D}(\boldsymbol{z}_q)$
  Extract perceptual features $f_{gen} \leftarrow \Phi(I_{gen})$
  Update feature queue and compute μ_{gen}, Σ_{gen}
  Compute $\mathcal{L}_{FD}$ via Eq. (11).
  Backpropagate: $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} - \eta \nabla_{\boldsymbol{\phi}} \mathcal{L}_{FD}$.
end while

IV. Theoretical Analysis

In this section, we theoretically compare the functional complexities of the proposed MIND and continuous diffusion models (CDM), specifically isolating the effect of incorporating explicit data manifold geometry. We prove that the metric entropy of the posterior mean hypothesis classes in MIND

scales linearly with the vocabulary size V of the tokenizer and dimension L of the hypersphere, while that of CDM scales exponentially with the data dimension. This reveals that the functional complexity of the proposed MIND is smaller than that of CDM and MIND can achieve better precision with the same amount of parameters. Theorems 1 and 2 establish the metric entropy for tokens at the single and sequence-level, respectively.

Assumption 1 (Data Prior of CDM): The continuous latent distribution p_0^{cont} of CDM is supported on a compact smooth manifold $\mathcal{M} \subset \mathbb{R}^{L_c}$ and belongs to the Hölder space $\mathcal{C}^s_{M_1}(\mathcal{M})$ for a finite smoothness index $s \leq s_0 < \infty$.

For the data with a single token, Theorem 1 shows that the metric entropy of the posterior mean for the hypothesis class in MIND is smaller than CDM.

Theorem 1 (Functional Complexity of the Posterior Mean): Let $\varepsilon > 0$ be the approximation error bound in the ℓ_∞-norm. Let $\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t]$ represent the target posterior mean, denoted by $f^*_{\text{dist}}(\boldsymbol{x}_t)$ and $f^*_{\text{cont}}(\boldsymbol{x}_t)$ for MIND and CDM, respectively. For any $\delta > 0$, with probability at least $1-\delta$, we have

(1) MIND: Denote $\mathcal{F}_{\text{disc}}$ as the hypothesis class of $f^*_{\text{dist}}(\boldsymbol{x}_t)$. The metric entropy of $\mathcal{F}_{\text{disc}}$ satisfies

$$\log \mathcal{N}(\mathcal{F}_{\text{disc}}, \varepsilon, \|\cdot\|_\infty) = \mathcal{O}\left(VL \log \varepsilon^{-1}\right), \tag{12}$$

where V is the vocabulary size and L is the dimension of the continuous latent in MIND.

(2) CDM: Let $\mathcal{F}_{\text{cont}}$ be the hypothesis class of $f^*_{\text{cont}}(\boldsymbol{x}_t)$, under Assumption 1, the metric entropy of $\mathcal{F}_{\text{cont}}$ is lower-bounded by

$$\log \mathcal{N}(\mathcal{F}_{\text{cont}}, \varepsilon, \|\cdot\|_\infty) = \Omega\left((\varepsilon)^{-L_c/s}\right), \tag{13}$$

where L_c is the dimension of the continuous latent in CDM.

Proof: Please refer to S3-B of the *Supplementary material*. ∎

Next, we analyze the function complexity with sequence-level data containing N tokens. Let $\boldsymbol{\pi}_n(\boldsymbol{x}_t) \in \mathbb{R}^V$ denote the routing probability vector for the n-th token.

Assumption 2: For any fixed input sequence $\boldsymbol{x}_t \in \mathcal{K}_\delta$, where $\mathcal{K}_\delta$ is a compact domain s.t. $\boldsymbol{x}_t \in \mathcal{K}_\delta$ with probability at least $1-\delta$, there exists a finite constant $C_\pi < \infty$ such that the gradients satisfy

$$\sum_{v=1}^{V} \left\| \nabla_{\boldsymbol{a}_j} [\boldsymbol{\pi}_n(\boldsymbol{x}_t)]_v \right\|_2 \leq C_\pi, \tag{14}$$

for all $n \in \{1, \ldots, N\}$ and $j, v \in \{1, \ldots, V\}$, where $\boldsymbol{a}_j$ represents the j-th vocabulary anchor.

To see why this assumption naturally holds, consider a representative instantiation where the routing probabilities are derived from inner-product-based logits:

$$[\boldsymbol{\pi}_n(\boldsymbol{x}_t)]_v = \frac{\exp(\mathbf{h}_n^\top \boldsymbol{a}_v/\tau)}{\sum_{k=1}^{V} \exp(\mathbf{h}_n^\top \boldsymbol{a}_k/\tau)}, \tag{15}$$

where $\mathbf{h}_n$ denotes the network's hidden state, and $\tau > 0$ is a temperature parameter. Since hidden states $\mathbf{h}_n$ are typically bounded by normalization layers (e.g., LayerNorm) and anchors reside in a compact domain, the composite partial derivatives with respect to the anchors remain structurally bounded due to the smoothness of the softmax operator, thereby precluding gradient explosions.

Theorem 2: Let N denote the sequence length and $\varepsilon > 0$ denote the approximation error bound in the ℓ_∞-norm. Given the continuous noisy sequence $\boldsymbol{x}_t$, consider the exact sequence-level posterior mean operator $\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t]$ mapping denoted by $\boldsymbol{F}^*_{\text{disc}}(\boldsymbol{x}_t)$ and $\boldsymbol{F}^*_{\text{cont}}(\boldsymbol{x}_t)$, respectively. For any $\delta > 0$, with probability at least $1-\delta$, we have

(1) MIND. Let $\mathcal{F}^{\text{seq}}_{\text{disc}}$ be the hypothesis class of the posterior mean $\boldsymbol{F}^*_{\text{disc}}(\boldsymbol{x}_t)$, where $\boldsymbol{x}_t \in \mathbb{R}^{N \times L}$. Under Assumption 2, the metric entropy satisfies

$$\log \mathcal{N}(\mathcal{F}^{\text{seq}}_{\text{disc}}, \varepsilon, \|\cdot\|_\infty) = \mathcal{O}\left(V(L+N) \log \varepsilon^{-1}\right). \tag{16}$$

(2) CDM. Let $\mathcal{F}^{\text{seq}}_{\text{cont}}$ be the hypothesis class of the continuous sequence posterior mean $\boldsymbol{F}^*_{\text{cont}}(\boldsymbol{x}_t)$, where $\boldsymbol{x}_t \in \mathbb{R}^{N \times L_c}$. Assuming the sequence data prior is globally $\mathcal{C}^s$-smooth and Assumption 1 holds on, the metric entropy satisfies

$$\log \mathcal{N}(\mathcal{F}^{\text{seq}}_{\text{cont}}, \varepsilon, \|\cdot\|_\infty) = \Omega\left(\varepsilon^{-NL_c/s}\right). \tag{17}$$

Proof: Please refer to S3-C of the *Supplementary material*. ∎

Remark 1: Theorems 1 and 2 rigorously disentangle the representational difficulties of the two paradigms. For MIND, the target posterior mean remains within the finite-dimensional manifold. Therefore, the metric entropy scales only logarithmically with respect to the precision requirement ($\log \varepsilon^{-1}$). In contrast, CDM operates in an unconstrained continuous space where the optimal posterior mean is defined as an integral ratio. Since the continuous latents are merely regularized toward a Gaussian prior, the smoothness index s is naturally bounded from above. Consequently, its metric entropy suffers from a severe polynomial explosion ($\varepsilon^{-NL_c/s}$). Therefore, MIND effectively bypasses this high-dimensional bottleneck, achieving superior representational efficiency and robustness against stringent precision demands.

V. Experiments

A. Experimental Settings

To evaluate our method, we followed the setups of DiT [17] and SiT [18] to perform experiments on the standard ImageNet dataset [62]. Each image was processed to the resolution of 256×256 and encoded into discrete token sequences of length 256 using the pre-trained tokenizer Index Backpropagation Quantization (IBQ) [35] and GigaTok [63] with a vocabulary size of V. These discrete tokens are mapped to continuous latent features on the hypersphere surface. The model was trained within the continuous feature space using the forward noising formulation in Eq. (1) and the hybrid loss combining cross-entropy and MSE defined in Eq. (4). To enable classifier-free guidance during inference, we randomly replaced class labels with an unconditional token with a probability of $p_{drop} = 0.1$ during training. All models were optimized using the AdamW optimizer, utilizing chunked gradient checkpointing to optimize memory consumption. Detailed training hyperparameters, including specific learning rates and batch sizes for each experimental setting, are comprehensively summarized in Table S-3 of the *Supplementary material*. During the inference

TABLE II: **Class-conditional image generation without guidance on Imagenet 256× 256.** Bold values indicate the best performance. "N/A" denotes that the metric was not reported in the original publication. All models are trained for 80 epochs.

	#Params	FID ↓	IS ↑	Precision ↑	Recall ↑
DiT-S/2 [17]	33M	68.40	N/A	N/A	N/A
SiT-S/2 [18]	33M	57.64	24.78	0.41	0.60
MIND-S(Our)	∼35M	**40.72**	**31.51**	**0.48**	**0.61**
eMIGM-S [31]	97M	44.47	19.82	0.49	0.57
DiT-B/2 [17]	130M	43.47	N/A	N/A	N/A
SiT-B/2 [18]	130M	33.02	43.71	0.53	**0.63**
MIND-B(Our)	∼ 130M	**22.73**	**56.17**	**0.55**	0.58
DiT-L/2 [17]	458M	23.33	N/A	N/A	N/A

phase, we employed a custom SDE solver in Algorithm 2 with 250 denoising steps to iteratively map Gaussian noise back to the data manifold. At each timestep t, the backbone network s_ϕ processed the continuous latent state to predict the unnormalized logit distribution over the discrete token vocabulary. We applied classifier-free guidance with scale cfg for condition alignment.

B. Performance Evaluation

We report standard generative metrics: Fréchet Inception Distance (FID), Inception Score (IS), Precision, and Recall on 50,000 samples that are generated with class-balanced sampling following [37].

Image Generation without Guidance. We performed a comprehensive system-level evaluation comparing our proposed model against the vanilla DiT baseline. To ensure a fair comparison under a constrained computational budget, all models were evaluated after training for 80 epochs on the ImageNet 256×256 dataset without classifier-free guidance. Table S-3 of the *Supplementary material* summarizes the hyperparameters during inference. As shown in Table II, the proposed MIND significantly outperforms the vanilla DiT, demonstrating vastly superior performance. Specifically, at the small (S) scale, the proposed MIND-S achieves a remarkable FID of 40.72, yielding a massive absolute improvement of 27.68 over DiT-S/2 (68.40). This trend is further amplified at the base (B) scale, where the proposed MIND-B reaches an FID of 22.73, nearly halving the FID of DiT-B/2 (43.47). Furthermore, compared to more advanced frameworks such as the flow-based SiT and the discrete diffusion-based model eMIGM [31], the proposed MIND establishes state-of-the-art FID while maintaining highly competitive diversity. It should be noted that the proposed method using 130M parameters even outperforms DiT-L with 458M parameters.

Image Generation with Guidance. Table III presents the quantitative results for class-conditional image generation with guidance on ImageNet 256 × 256. In the absence of publicly available 80-epoch performance and checkpoints for DiT, SiT, and eMIGM, we report results by our training from scratch following the default hyperparameter configurations specified in the original works. We sampled with different classifier-free guidance scale cfg and default hyperparameters in the original works. eMIGM was sampled with the recommended higher cfg since it adopts the time interval guidance strategy by default. The proposed MIND demonstrates superior generative performance across different model scales and classifier-free guidance scales. Under the small-scale setting (∼35M parameters), the proposed MIND-S significantly outperforms the baseline models. Specifically, at $cfg = 1.5$, MIND-S improves the FID by 22.09 and 13.29 compared with DiT-S/2 and SiT-S/2, respectively. MIND-S surpasses eMIGM-S with only one-third of the parameters. Scaling up to the Base configuration (∼130M parameters), the proposed MIND-B consistently exhibits strong capabilities. At $cfg = 1.5$, MIND-B reaches the best FID of 12.15 and the highest IS of 100.16. When the guidance scale is increased to 2.0, our model reduces FID by 2.98 and 1.1 compared with DiT-B/2 and SiT-B/2, respectively.

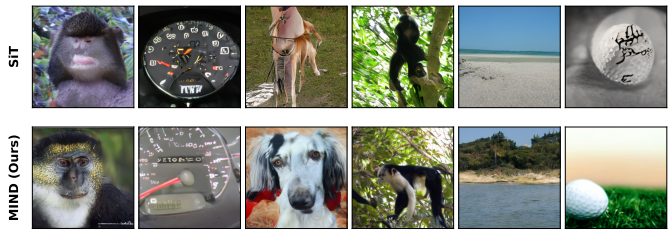


Fig. 4: Visual comparison between our MIND-B and SiT-B/2 with cfg=2.0.

TABLE III: **Class-conditional image generation with guidance on ImageNet 256×256.** Bold values indicate the best performance. All models were trained for 80 epochs. Our method achieves a superior FID of 7.97, outperforming the baseline DiT (10.95) and flow-based SiT (9.07).

	#Params	cfg	FID ↓	IS ↑	Precision ↑	Recall ↑
DiT-S/2 [17]	33M	1.5	44.42	34.57	0.47	0.56
SiT-S/2 [18]	33M	1.5	35.62	43.00	0.52	**0.56**
MIND-S(Our)	∼35M	1.5	**22.33**	**60.32**	**0.63**	0.52
DiT-S/2 [17]	33M	2.0	29.29	55.37	0.57	0.51
SiT-S/2 [18]	33M	2.0	22.96	67.81	0.62	**0.51**
MIND-S(Our)	∼35M	2.0	**14.93**	**92.65**	**0.74**	0.45
eMIGM-S [31]	97M	1.3	41.18	21.23	0.50	0.56
eMIGM-S [31]	97M	4.0	23.52	30.83	0.61	0.51
DiT-B/2 [17]	130M	1.5	20.01	73.00	0.65	0.56
SiT-B/2 [18]	130M	1.5	16.88	84.26	0.66	**0.56**
MIND-B(Our)	∼ 130M	1.5	**12.15**	**100.16**	**0.71**	0.51
DiT-B/2 [17]	130M	2.0	10.95	119.12	0.76	0.47
SiT-B/2 [18]	130M	2.0	9.07	137.07	0.77	**0.47**
MIND-B(Our)	∼ 130M	2.0	**7.97**	**154.20**	**0.81**	0.43

Fig. 4 illustrates a qualitative comparison between SiT-B/2 and the proposed MIND-B, using the same class labels with $cfg = 2.0$. Our method synthesizes significantly more realistic images with precise semantic alignment, effectively avoiding the structural distortions observed in SiT-B/2. These visual results align with our superior quantitative scores in FID, IS, and Precision, which demonstrate that prioritizing sample fidelity and precision over diversity (Recall) yields more visually compelling and structurally accurate generative outcomes.

C. System-level Comparisons with Prior Work

In this section, we present the ultimate performance evaluation of our model against existing state-of-the-art methods. For inference, the classifier-free guidance scale is applied in a limited interval without entropy-driven adaptive temperature as proposed in [64]. The training and inference parameters remain consistent with Table S-3 of the *Supplementary material*, except those summarized in Table S-4. MIND-B-G and MIND-XL are sampled with NPU-compatible code. We report five generative metrics: FDr^6, FID, IS, Precision, and Recall. The FDr^6, defined as a normalized FD ratio averaged over 6 feature spaces [61] (including Inception-v3, ConvNeXtv2, DINOv2, MAE, SigLIP2, and CLIP), is reported to reduce the blind spots of any single representation.

As shown in Table IV, quantitative evaluations on ImageNet 256×256 demonstrate the exceptional parameter efficiency of the proposed method. Our MIND with about 130M parameters achieves a highly competitive FID of 2.06 (IS: 268.03, Precision: 0.78), successfully outperforming mainstream baselines that are substantially larger across multiple generative paradigms. Specifically, MIND-B-G surpasses billion-parameter discrete models like LlamaGen-3B (3.1B, FID 2.18), RQTran. [65] (3.8B, FID 3.80), IBQ-XXL (2.1B, FID 2.05) [35], VAR [55] (1.0B, FID2.09).

Furthermore, our method maintains a distinct advantage over continuous architectures. Specifically, compared with its baseline DiT-XL/2 [17], MIND improves the generation ability from FID=2.27 to 1.84. The proposed MIND-XL-G outperforms the 2B-parameter SimDiff [66] and SiT-XL/2 [18]. Compared with recent masked generative models that rely on continuous tokens, such as MAR-B [67], the proposed MIND-XL-G reduces the FDr^6 from 5.61 to 4.59. Without relying on FD-loss post-training, the proposed MIND achieves superior FDr^6 among existing baselines. Ultimately, our approach challenges the conventional reliance on massive parameter scaling, demonstrating that a compact $\sim$130M model can effectively match or exceed the performance of state-of-the-art architectures ranging from 300M to 3B parameters. Fig. 5 provides some visual results from our MIND-B-G. Note that, as a foundational framework, MIND is fully compatible with advanced orthogonal enhancements (e.g., codebook scaling [57], token generation ordering [68], and variable-length coding [69]). These techniques can be seamlessly integrated into our backbone to further elevate empirical performance in future exploration.

D. Evaluating One-Step Image Generation

We evaluate the one-step generation capability of the proposed differentiable distillation equipped with a straight-through estimator. To preclude metric-specific overfitting to FDr^6 and ensure the impartiality of empirical evaluations, feature extraction across all one-step generation frameworks is restricted strictly to the standard Inception network during the optimization of the FD loss. Starting from the pretrained MIND-B-G model, we fine-tune the network for 35 epochs with a batch size of 64 and $cfg = 3.0$. Scaling up, the pretrained MIND-XL-G model is fine-tuned for 50 epochs with a batch size of 256 and $cfg = 1.5$.

Within the multi-step to one-step distillation track, the proposed MIND framework establishes new state-of-the-art results. MIND-B-G achieves an FDr^6 of 7.58 and an FID of 0.9034 using merely 130M parameters, already surpassing the much larger JiT-H (953M, FDr^6 of 10.18). When scaled to MIND-XL-G, our model achieves an FDr^6 of 6.77, representing a 33.5% reduction in overall FDr^6 compared to JiT-H with 25% fewer parameters. Compared with one-step to one-step distillation frameworks (e.g., iMF-XL and pMF-H), our distilled one-step models achieve comparable performance. However, our original MIND-XL-G achieves an FDr^6 of 4.56, substantially outperforming all original one-step baselines (e.g., pMF-H at 6.87). This demonstrates that MIND not only compresses into a competitive one-step generator, but also provides a superior performance ceiling when higher visual quality is prioritized. Fig. 6 shows some visual results from the one-step MIND-XL-G.

To further validate the structural advantages of extending FD loss to the discrete generative paradigm, we evaluate the frequency-domain fidelity of various one-step distilled models via 1D radial power spectrum analysis. Specifically, generated images are first converted to grayscale luminance Y using the standard formulation $Y = 0.2126R + 0.7152G + 0.0722B$. After removing the spatial mean, we apply a 2D Fast Fourier Transform (FFT) and compute the squared magnitude. To obtain an isotropic energy profile, the 2D spectrum is azimuthally averaged into discrete radial frequency bins $r \in [1, 128]$. Spectral distortion is rigorously quantified by the log-scale Root Mean Square Error (dB-RMSE) over 50K generated samples per method, referenced against the ImageNet validation set. As illustrated in Fig. 7, prior continuous baselines suffer from inherent frequency decay: JiT-H and pMF-H exhibit noticeable energy attenuation in the high-frequency regime, while iMF-XL deteriorates in the middle-frequency band. In contrast, by successfully integrating FD loss with explicitly modeled discrete manifolds, our MIND-XL-G rigorously preserves spectral integrity across all frequency bands. It strictly aligns with the reference distribution and achieves a substantially lower dB-RMSE, demonstrating that ultra-fast one-step generation and pristine frequency-domain fidelity can be simultaneously achieved in the proposed framework.

E. Ablation Study

1) Hypersphere justification and ablation: The hypersphere constraint *controls the magnitude of data components, which is essential to regulate the smooth signal-to-noise ratio variations*. We train MIND-B with 32 epochs and sample 50K images without guidance using the same parameters (linear timestep schedule, $\tau = 0.99$, Top-k=100, Top-p=0.8, $\eta = 0.99$, $\rho_1 = 0.9$, $\rho_2 = 0.1$, sampling steps=250, without entropy). As listed in Table VI, (B) without the ℓ_2-norm, the training loss fails to converge due to severe norm shrinkage (the average norm of the continuous latents $\|\mathcal{P}_\theta\|_2$ drops to 0.04). (C) Re-initializing of $\mathcal{P}_\theta$ to force convergence drops loss to near zero but *collapses generative performance* (FID 194.61), as the

TABLE IV: Quantitative comparison of image generation models on ImageNet 256×256. Models are categorized by their generation paradigm and sorted by parameter size ascendingly within each group. With only $\sim$130M parameters, our method achieves performance exceeding discrete models with billions of parameters and continuous models with hundreds of millions of parameters. Models with the suffix "-re" used rejection sampling, *: taken from VAR [55], -G: GigaTok, **: models with semantic distillation.

Method	Family	#Params	FDr[6]	FID ↓	IS ↑	Prec. ↑	Rec. ↑
Continuous Models							
BigGAN [70]	GAN	112M	N/A	6.95	224.5	0.89	0.38
GigaGAN [63]	GAN	569M	N/A	3.45	225.5	0.84	0.61
StyleGan-XL [71]	GAN	166M	N/A	2.30	265.1	0.78	0.53
ADM-G [11]	Diff.	554M	N/A	4.59	186.70	0.82	0.52
ADM-G, ADM-U [11]	Diff.	554M	N/A	3.94	215.84	0.83	0.53
LDM-4-G [14]	Diff.	400M	N/A	3.60	247.7	0.87	0.48
DiT-L/2* [17]	Diff.	458M	N/A	5.02	167.2	0.75	0.57
DiT-XL/2 [17]	Diff.	675M	N/A	2.27	278.2	0.83	0.57
SimDiff [66]	Diff.	2B	N/A	2.77	211.8	N/A	N/A
SiT-XL/2 [18]	Diff.	675M	8.44	2.06	270.3	0.82	0.59
eMIGM-XS [31]	Masked	69M	N/A	3.62	224.91	0.80	0.51
eMIGM-S [31]	Masked	97M	N/A	2.87	254.48	0.80	0.54
eMIGM-B [31]	Masked	208M	N/A	2.32	278.97	0.81	0.57
eMIGM-L [31]	Masked	478M	N/A	1.72	304.16	0.80	0.60
eMIGM-H [31]	Masked	942M	N/A	1.57	305.99	0.80	0.63
MAR-B [67]	MAR	208M	N/A	2.31	281.7	0.82	0.57
MAR-L [67]	MAR	479M	6.68	1.78	296.0	0.81	0.60
MAR-H [67]	MAR	943M	5.61	1.55	303.7	0.81	0.62
RJF [27]	Diff.	131M	N/A	3.37	180.26	0.80	0.56
RJF [27]	Diff.	677M	N/A	2.81	201.22	0.82	0.56
REPA** [40]	Diff.	675M	N/A	1.29	306.3	0.79	0.64
REPA-E** [41]	Diff.	675M	3.04	1.12	302.9	0.79	0.66
DDT** [72]	Diff.	675M	3.04	1.26	310.6	0.79	0.65
RAE** [37]	Diff.	839M	3.26	1.13	262.6	0.78	0.67
PixelFlow [73]	Flow	677M	N/A	1.98	282.1	0.81	0.60
MeanFlow-XL/2 [43]	Flow	676M	N/A	2.20	N/A	N/A	N/A
IMF-XL/2 (1NFE) [44]	Flow	610M	8.39	1.72	N/A	N/A	N/A
IMF-XL/2 (2NFE) [44]	Flow	610M	7.48	1.54	N/A	N/A	N/A
JiT-L [74]	Flow	459M	10.73	2.59	288.5	0.79	0.59
JiT-H [74]	Flow	953M	7.66	1.97	296.0	0.78	0.63
Discrete Models							
VQGAN [32]	AR	1.4B	N/A	15.78	74.3	N/A	N/A
VQGAN-re [32]	AR	1.4B	N/A	5.20	280.3	N/A	N/A
ViTVQ [75]	AR	1.7B	N/A	4.17	175.1	N/A	N/A
ViTVQ-re [75]	AR	1.7B	N/A	3.04	227.4	N/A	N/A
RQTran. [65]	AR	3.8B	N/A	7.55	134.0	N/A	N/A
RQTran.-re [65]	AR	3.8B	N/A	3.80	323.7	N/A	N/A
LlamaGen-B [28]	AR	111M	N/A	5.46	193.61	0.83	0.45
LlamaGen-L [28]	AR	343M	N/A	3.07	256.06	0.83	0.52
LlamaGen-XL [28]	AR	775M	N/A	2.62	244.08	0.80	0.57
LlamaGen-XXL [28]	AR	1.4B	N/A	2.34	253.90	0.80	0.59
LlamaGen-3B [28]	AR	3.1B	N/A	2.18	263.33	0.81	0.58
VAR-d16 [55]	VAR	310M	N/A	3.30	274.4	0.84	0.51
VAR-d20 [55]	VAR	600M	N/A	2.57	302.6	0.83	0.56
VAR-d24 [55]	VAR	1.0B	N/A	2.09	312.9	0.82	0.59
VAR-d30 [55]	VAR	2.0B	6.70	1.92	323.1	0.82	0.59
IBQ-B [35]	AR	342M	N/A	2.88	254.73	0.84	0.51
IBQ-L [35]	AR	649M	N/A	2.45	267.48	0.83	0.52
IBQ-XL [35]	AR	1.1B	N/A	2.14	278.99	0.83	0.56
IBQ-XXL [35]	AR	2.1B	N/A	2.05	286.73	0.83	0.57
MaskGIT [30]	Masked	227M	N/A	6.18	182.1	0.80	0.51
MaskGIT-re [30]	Masked	227M	N/A	4.02	355.6	N/A	N/A
MIND-B(Ours)	**Diff.+Dis.**	**$\sim$130M**	**10.31**	**2.18**	**258.46**	**0.80**	**0.58**
MIND-B-G(Ours)	**Diff.+Dis.**	**$\sim$130M**	**6.84**	**2.06**	**268.03**	**0.78**	**0.62**
MIND-XL(Ours)	**Diff.+Dis.**	**$\sim$715M**	**7.31**	**1.97**	**297.93**	**0.78**	**0.61**
MIND-XL-G(Ours)	**Diff.+Dis.**	**$\sim$715M**	**4.56**	**1.83**	**317.50**	**0.78**	**0.65**

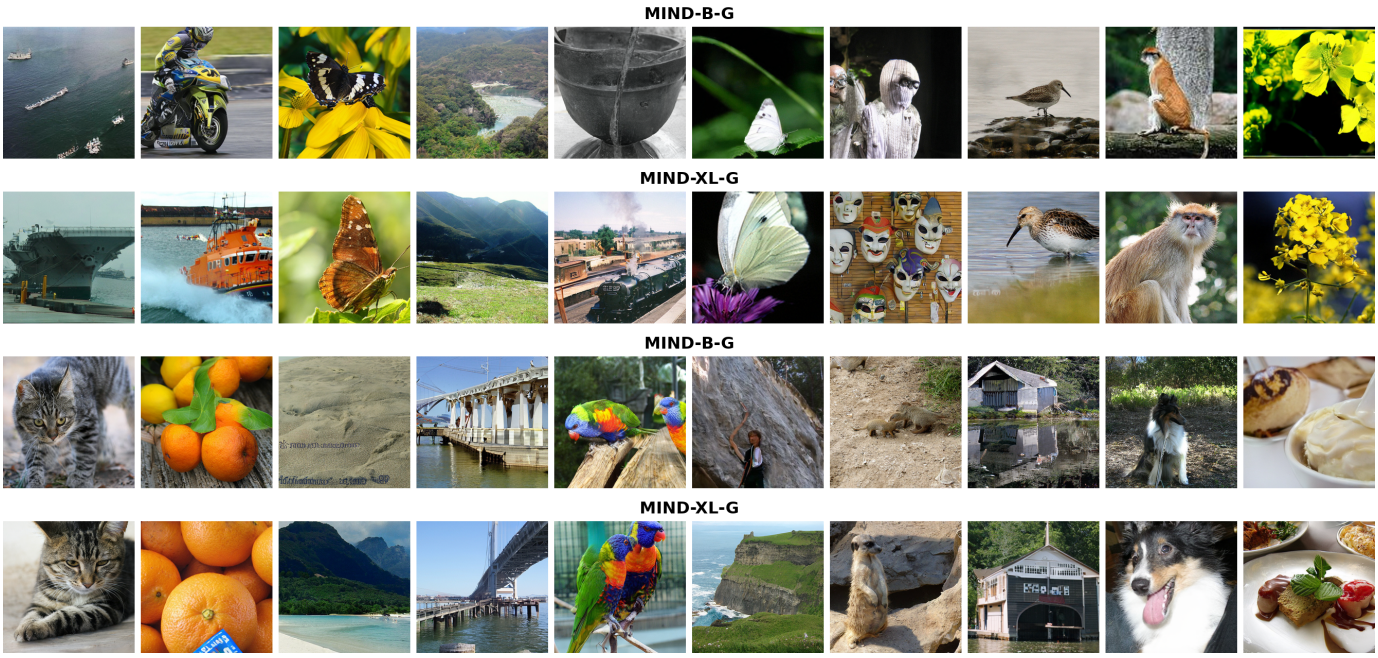


Fig. 5: Examples of visual results from MIND-B-G (the first and third rows) and MIND-XL-G (the second and fourth rows).

TABLE V: Quantitative comparison of one-step image generation models by distilling multi-step models with FD loss on ImageNet 256×256. We report per-representation FDr after distilling the model with the Inception network.

Method	Family	#Params	FDr^6↓	Incep.↓	ConvNeXt↓	DINOv2↓	MAE↓	SigLIP↓	FDr-CLIP↓	FID ↓
Original models										
JiT-L [74]	Flow	459M	10.73	1.54	3.49	6.10	8.07	19.37	25.82	2.59
JiT-H [74]	Flow	953M	7.66	1.18	2.52	4.28	5.65	11.91	20.40	1.97
iMF-XL(1NFE) [44]	Flow	610M	8.39	1.09	1.72	7.30	6.09	17.02	17.14	1.82
pMF-L [76]	Flow	410M	9.09	1.62	1.36	6.70	9.72	20.34	14.81	2.72
pMF-H [76]	Flow	956M	6.87	1.37	1.15	5.43	6.25	15.33	11.68	2.29
MIND-B-G (Ours)	**Diff.+Dis.**	**∼130M**	**6.84**	**1.22**	**1.54**	**5.89**	**7.12**	**13.48**	**11.79**	**2.06**
MIND-XL-G (Ours)	**Diff.+Dis.**	**∼710M**	**4.56**	**1.09**	**0.95**	**3.34**	**5.13**	**8.12**	**8.74**	**1.83**
Distill multi-step models to one-step models with FD loss										
JiT-L(-Incep.) [74]	Flow	459M	12.86	0.46	2.57	7.34	15.28	26.22	25.29	0.77
JiT-H(-Incep.) [74]	Flow	953M	10.18	0.43	1.92	5.39	13.21	20.70	19.44	0.72
MIND-B-G (Ours,1NFE)	**Diff.+Dis.**	**∼130M**	**7.58**	**0.54**	**1.34**	**6.36**	**6.98**	**16.59**	**13.67**	**0.9034**
MIND-XL-G (Ours,1NFE)	**Diff.+Dis.**	**∼710M**	**6.77**	**0.53**	**1.19**	**5.14**	**6.60**	**14.24**	**12.93**	**0.8985**
Distill one-step models to one-step models with FD loss										
iMF-XL(-Incep.) [44]	Flow	610M	6.01	0.43	1.04	4.54	4.25	12.17	13.64	0.72
pMF-L(-Incep.) [76]	Flow	410M	6.19	0.44	0.94	4.60	4.24	14.34	12.55	0.73
pMF-H(-Incep.) [76]	Flow	956M	4.86	0.43	0.62	3.58	3.42	10.81	10.27	0.72

network amplifies latent magnitude ($\|\mathcal{P}_\theta\|_2$ explodes to 9.68) to trivially overwhelm injected noise. (D) Even increasing the noise schedule ($c_1 = 1.0$) fails, with a trivial solution of a larger latent (13.66) for diffusion. Thus, the hypersphere is a vital structural regularizer that anchors the data scale.

2) Ablation of components in MIND: We experiment with MIND-B using the same settings as Table VI. As shown in Table VII, the hypersphere constraint is crucial for valid generation, with soft top-k and advanced tokenizer offering further refinement, and the multi-stage sampling strategy is critical for generation with guidance.

3) Embedding Dimension and Noise Scaling: To systematically analyze the impact of the continuous embedding space capacity and the diffusion noise schedule, we conducted a comprehensive ablation study focusing on the embedding dimension (L) and the diffusion coefficients (c_1 for noise, c_2 for signal). We perform experiments on MIND-B with 2K samples utilizing a fixed cfg=4.0. All models were trained for 12 epochs.

As shown in Table IX, varying the embedding dimension explicitly controls the representational bottleneck. A lower dimension overly compresses the topological structure, whereas an excessively high dimension introduces spatial sparsity and complicates the continuous diffusion matching process. Concurrently, adjusting the noise scaling factor c_1 and signal scaling factor c_2 directly modulates the signal-to-noise ratio schedule during the forward noising process. Specifically, setting the noise scale $c_1 = 0.6$ effectively constrains the

Fig. 6: Examples of visual results from our one-step MIND-XL-G.

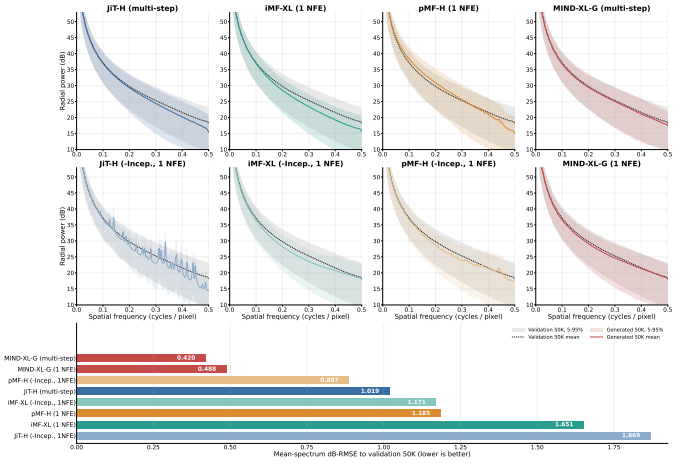


Fig. 7: Frequency-domain fidelity. Mean radial power spectra of generated and ImageNet validation images. Curves closer to the reference, and lower dB-RMSE, indicate better frequency-statistics alignment.

TABLE VI: Hypersphere ablation. $||\mathcal{P}_\theta||_2$: the average norm of the continuous latents.

Method	(A) MIND-B (Xavier init, $c_1 = 0.6$)	(B) no ℓ_2	(C) no ℓ_2, Normal init	(D) no ℓ_2, Normal init $c_1 = 1.0$
Train Loss	4.02	9.57	0.007	0.04
$||\mathcal{P}_\theta||_2$	2.00	0.04	9.68	13.66
FID ↓	**52.10**	280.78	194.61	124.40

TABLE VII: Component ablation.

Model Variant	**FID**↓ ($cfg = 1.0$ in default)
(A) MIND-B w/o Hypersphere	280.78
(B) MIND-B w/o Soft top-k	53.03
(C) MIND-B (32 epochs)	52.10
(D) + Longer Training (200 epochs)	19.32
(E) + Advanced Tokenizer (GigaTok)	11.38, 6.42 ($cfg = 4.0$)
(F) + Multi-stage Sampling Strategy	12.00, 2.64 ($cfg = 4.0$)

variance within the bounded ℓ_2-normalized embedding space and significantly reduces the overall FID.

4) Residual Connection: To investigate the impact of the proposed residual sinusoidal projection module, we conducted an ablation study across embedding dimensions $L \in \{16, 64\}$. We performed experiments on MIND-B with 2K samples utilizing a fixed cfg=4.0. All models were trained for 12 epochs. As shown in Table X, the residual connection generally enhances both the fidelity and diversity of the generated images. At $L = 16$, the residual pathway reduces the FID from 34.91 to 33.91 and improves the IS from 119.14 to 120.56. This improvement becomes even more pronounced at a higher dimension ($L = 64$), where the model with the residual connection achieves a substantial FID reduction of 2.46 and an IS increase of 8.43 compared to the baseline. The Recall metric consistently improves across all evaluated dimensions when the residual connection is applied. This consistent gain indicates that the residual mechanism effectively helps the model preserve high-frequency details and data distribution modes.

TABLE VIII: Inference latency comparison on a single NVIDIA-3090 GPU (Batch=1, 256×256).

Method	**Steps**	**Sampling (s)**	**Decoding (s)**	**Total (s)**
DiT-B/2	250	2.10	0.03	2.13
SiT-B/2	250	0.54	0.03	0.57
eMIGM	128	10.49	0.02	10.51
MIND-B	250	3.25	0.03	3.28

TABLE IX: Ablation study on the embedding dimension (L) and diffusion scaling coefficients (c_1, c_2).

L	d_{sub}	c_1 (Noise)	c_2 (Signal)	FID ↓	IS ↑	Precision ↑	Recall ↑
8	2	0.6	1.0	125.40	17.21	0.26	0.29
8	2	1.0	1.0	206.19	7.52	0.26	0.28
16	4	0.6	1.0	**34.91**	**119.14**	**0.81**	**0.56**
32	8	0.6	1.0	38.95	89.63	0.71	0.53
64	16	0.6	1.0	43.03	72.49	0.63	0.51
64	16	0.8	1.0	44.07	67.24	0.59	0.51
128	32	1.0	1.0	52.03	51.35	0.50	0.46

TABLE X: Ablation study on the effectiveness of residual connection.

	L	d_{sub}	c_1	c_2	FID ↓	IS ↑	Precision ↑	Recall ↑
w/o res	16	4	0.6	1.0	34.91	119.14	0.81	0.56
w/ res	16	4	0.6	1.0	**33.91**	**120.56**	**0.79**	**0.58**
w/o res	64	16	0.6	1.0	43.03	72.49	0.63	0.51
w/ res	64	16	0.6	1.0	40.57	80.92	0.64	0.54

Note that NPU-compatible code was used for the ablations on hypersphere justification, component analysis, and hyperparameter robustness, and GPU-compatible code for others. More ablation studies about sampling parameters can be referred to Table S-1 and Table S-2 of the *Supplementary material*.

F. Efficiency and Robustness Analysis

1) Inference efficiency and convergence speed.: We conducted rigorous wall-clock profiling on a single NVIDIA-3090 GPU (Batch=1). The tokenizer in MIND-B does not need extra training latency using pre-encoded tokens, with decoding costs on par with VAEs in DiT/SiT architectures. MIND-B, implemented with CFG intervals like eMIGM, incurs a sampling cost that matches DiT and is significantly lower than eMIGM. Moreover, we test the convergence speed using the experiment settings in Table III, MIND-B exhibits superior convergence speed as shown in Fig. 8.

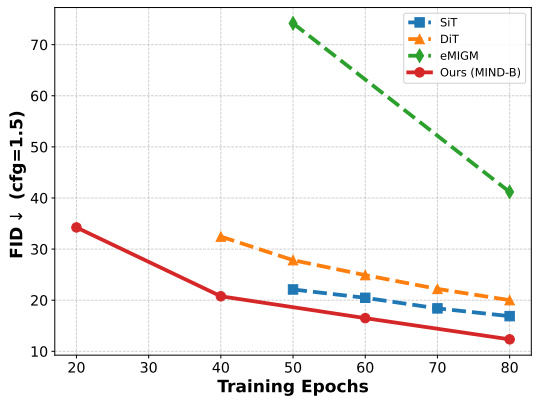


Fig. 8: Convergence speed

2) Hyperparameter robustness: As shown in Fig 9, extensive evaluations of MIND-B-G trained for 200 epochs with 768 sampling settings demonstrate strong hyperparameter robustness. Variations across most parameters (p, k, τ, ρ_2, cfg, and $[c_{min}, c_{max}]$) shift the average FID by $\leq 4.0\%$. While ρ_1 and η are more sensitive, fixing $\eta = 0.99$ and $\rho_1 = 0.1$ consistently yields excellent results across all our image generation with guidance. Consequently, adapting MIND to new datasets or resolutions circumvents the need for brittle hyperparameter tuning.

VI. Extension to Text-to-Image Generation

In this section, we extend the proposed framework to text-to-image generation. Note that our primary goal is to validate the generality of the proposed MIND architecture rather than competing for state-of-the-art performance. Therefore, we follow [77] to train the 130M scale MIND-B-G on the ImageNet

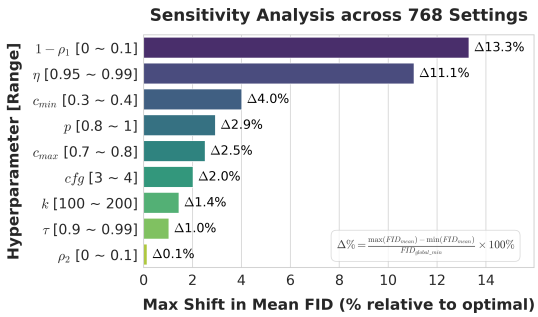


Fig. 9: Hyperparameter robustness.

dataset using text and image augmentations, circumventing the need for commercial-scale datasets and computational resources. To seamlessly integrate complex textual conditions into the visual generation process, we adopt a dual-track text injection mechanism including a global semantic modulation track via adaptive layer normalization (AdaLN) and a local alignment track via multi-head cross-attention (MHCA).

The text prompt is first encoded by the T5 text encoder [78]. To efficiently handle variable lengths, the resulting embeddings are padded or truncated to a fixed maximum sequence length of 256, accompanied by a binary attention mask to identify valid tokens. These text embeddings are then projected into the latent dimension via a multi-layer perceptron. To enable classifier-free guidance, we randomly drop the text conditions during training. Subsequently, the projected sequence is processed by a *Text Refiner*—a 2-layer transformer encoder—to facilitate dense token-to-token interactions, producing a refined contextual sequence. For the global semantic modulation track (Track A), we aggregate this refined sequence into a single global vector using masked mean pooling, which computes the average of all valid token embeddings based on the attention mask. This global text representation is then fused with the diffusion timestep embedding to form a unified conditioning vector that predicts the scale and shift parameters for AdaLN.

While AdaLN efficiently governs global attributes, it lacks the spatial granularity required for fine-grained text-to-image alignment. To address this, our local alignment track (Track B) introduces a multi-head cross-attention mechanism within each block. Specifically, the intermediate spatial image tokens generated during the diffusion process serve as the queries for the multi-head cross-attention, while the refined text contextual sequence serves as both the keys and values. The binary text attention mask is applied during this operation to prevent image tokens from attending to padding tokens. The output of the cross-attention is then injected back into the residual stream, scaled by a learnable gating parameter derived from the global AdaLN condition. By dual-tracking the textual information, the spatial image tokens autonomously learn to query precise local text details through Track B, while adhering to the overarching artistic style dictated by the global conditioning in Track A.

The diffusion network is initialized by the weights in class-to-image generation and trained with a batch size of 1024 for 440 epochs. We access the generation quality of in-distribution with ImageNet validation set, zero-shot MSCOCO-30K, and Genval. We report FID with Inception-v3 backbones, Precision, Recall, Density, Coverage calculated with Dino-v2 features, and CLIP score. As shown in Table XI, the proposed MIND-B-G with only 133M parameters reduces FID by 1.4 and 33.88 on ImageNet and MS-COCO, respectively. We further experiment with the Genval benchmark [87]. As shown in Table XII, we achieve the overall score of 0.61 using only 1.2M training data and 130M scale model, which surpass DiT-I [77] using the same amount of training data but larger model and even surpass than billion scale model SDXL [82]. Fig. 10 provides some visual results. Note that our model is good at generating images with rich text description since it

TABLE XI: Text-to-image performance on ImagNet validation set and MS-COCO validation set. All models are trained with text and image augmentations. FID is computed with Inception V3, and others are computed with DINOV2.

Dataset	Models	#Para.	FID Inc.↓	Prec.↑	Rec.↑	Den.↑	Cov.↑	CLIP Score↑
ImageNet val	DiT-I	400M	7.30	0.79	0.74	0.88	0.75	N/A
ImageNet val	**MIND-B-G**	**133M**	**5.90**	**0.78**	**0.82**	**0.87**	**0.74**	N/A
MSCOCO	DiT-I	400M	49.12	0.65	0.45	0.54	0.30	25.68
MSCOCO	**MIND-B-G**	**133M**	**15.24**	**0.60**	**0.55**	**0.52**	**0.43**	**25.73**

TABLE XII: Text-to-image performance on Genval benchmark. Images in DiT-I and MIND-B-G are generated under 256×256 resolution.

Models	# Para.	# Train Data	Overall↑	One obj.↑	Two obj.↑	Count.↑	Col.↑	Pos.↑	Col. attr.↑
SDv1.5 [79]	0.9B	5B+	0.43	0.97	0.38	0.35	0.76	0.04	0.06
PixArt-α [80]	0.6B	25M	0.48	0.98	0.50	0.44	0.80	0.08	0.07
PixArt-Σ [81]	0.6B	35M+	0.52	0.98	0.59	0.50	0.80	0.10	0.15
SDv2.1 [79]	0.9B	5B+	0.50	0.98	0.51	0.44	0.85	0.07	0.17
SDXL [82]	3.5B	5B+	0.55	0.98	0.74	0.39	0.85	0.15	0.23
SD3M [83]	2B	1B+	0.62	0.98	0.74	0.63	0.67	0.34	0.36
SANA-0.6 [84]	0.6B	N/A	0.64	0.99	0.71	0.63	0.91	0.16	0.42
Sana-1.6 [84]	1.6B	N/A	0.66	0.99	0.79	0.63	0.88	0.18	0.47
FLUX-dev [85]	12B	N/A	0.67	0.99	0.81	0.79	0.74	0.20	0.47
LUMINA-Next [86]	2.0B	20M	0.46	0.92	0.46	0.48	0.70	0.09	0.13
DiT-I (256) [77]	400M	1.2M	0.58	0.95	0.67	0.43	0.80	0.30	0.35
MIND-B-G	**133M**	**1.2M**	**0.61**	**0.93**	**0.63**	**0.49**	**0.71**	**0.39**	**0.53**

is trained with text augmentations. We generate without text augmentations for MSCOCO-30K for fair comparison with baselines.

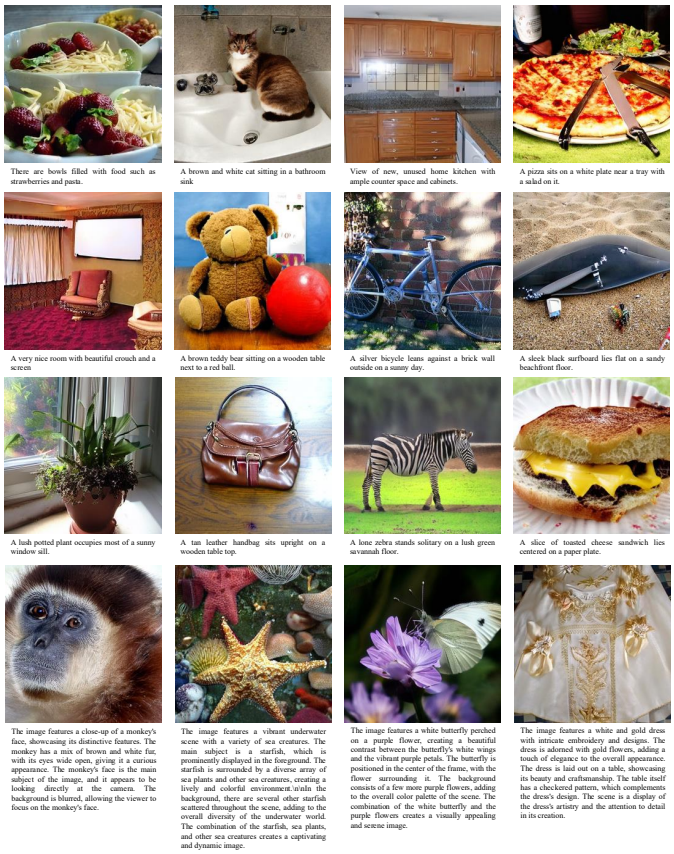


Fig. 10: Examples of visual results from our MIND-B-G for text-to-image generation.

VII. Conclusion and Discussion

In this paper, we introduced a novel generative framework that explicitly models the data manifold by bridging discrete image tokenization with continuous hyperspherical diffusion, which theoretically guarantees learning the score network in a parameterized space having lower metric entropy compared with traditional continuous diffusion models without an explicit data manifold. To enable this discrete-continuous hybrid system, we proposed a differentiable soft top-k aggregation mechanism for stable training and an entropy-driven hybrid sampling operator for robust inference. Furthermore, we propose a one-step distillation framework for our MIND based on the FD loss. Extensive experiments on ImageNet demonstrate that our approach significantly outperforms the DiT baseline, flow-based continuous models, and discrete image generation models.

It should be noted that we instantiated our method on the basic DiT architecture to highlight its fundamental superiority, our significant gains over DiT show its immense potential for image generation. To evolve MIND into a flexible generative foundation, we list several critical research frontiers below.

- **Scaling to ultra-large codebooks.** MIND can be extended by replacing the standard categorical logits prediction head with a compact bit prediction head, factorizing each discrete token index into an V_m-bit binary representation ($V = 2^{V_m}$) and explicitly modeling inter-bit correlations to preserve structural semantic relationships. Projecting these correlation-aware, high-capacity bit representations onto the continuous hyperspherical manifold enables MIND to achieve near-lossless discretization while maintaining lightweight memory consumption and linear scaling efficiency during generation.
- **End-to-end co-evolution of discrete and continuous representations.** The proposed MIND can be extended

to enable the joint optimization of discrete tokenizer and continuous diffusion models. Using differentiable quantization techniques allows continuous vector-field gradients to flow directly into discrete codebook centroids, thereby enabling dynamic representation co-evolution.

- **Extension to flow models.** MIND models the continuous generative process in parameterized space using standard diffusion paradigms, which can be extended to modern flow-based generative paradigms. By inherently leveraging optimal transport principles, flow-based modeling constructs straight, minimal-length velocity fields on the manifold.
- **Unified multimodal generation.** MIND naturally scales across a broad spectrum of complex domains, including text and code generation, video generation, 3D point cloud generation. These scenarios are inherently suitable for MIND because real-world physical and perceptual signals naturally possess a low-dimensional manifold. Crucially, this hybrid representation provides a principled foundation for unified multimodal generation. Discrete tokenization serves as a universal, modal-agnostic semantic bridge that compresses disparate heterogeneous inputs (text, audio, discrete visual priors) into a shared symbolic representation space. Concurrently, continuous manifold diffusion acts as a high-capacity geometric decoder, modeling fine-grained, domain-specific continuous variations within target manifolds. Explicitly co-modeling discrete semantics and continuous geometric latents establishes a versatile pipeline for next-generation, cross-modal foundation models.

We believe MIND introduces a general framework and fresh perspective on image generation, paving the way for future research and innovation in this community.

REFERENCES

[1] M. Kang, J. Shin, and J. Park, "Studiogan: A taxonomy and benchmark of gans for image synthesis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 12, pp. 15 725–15 742, 2023.

[2] A. Melnik, M. Miasayedzenkau, D. Makaravets, D. Pirshtuk, E. Akbulut, D. Holzmann, T. Renusch, G. Reichert, and H. Ritter, "Face generation and editing with stylegan: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 5, pp. 3557–3576, 2024.

[3] A. Vahdat and J. Kautz, "Nvae: A deep hierarchical variational autoencoder," in *Adv. Neural Inf. Process. Syst.*, vol. 33. Curran Associates, Inc., 2020, pp. 19 667–19 679.

[4] I. Kobyzev, S. J. Prince, and M. A. Brubaker, "Normalizing flows: An introduction and review of current methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 3964–3979, 2021.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, p. 139–144, 2020.

[6] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *2019 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 4396–4405.

[7] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," in *Adv. Neural Inf. Process. Syst.*, 2019, p. 11895–11907.

[8] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Adv. Neural Inf. Process. Syst.*, 2020, pp. 6840–6851.

[9] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *9th Int. Conf. Learn. Rep.*, 2021.

[10] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *Proc. 37th Int. Conf. Mach. Learn.*, 2021, pp. 8162–8171.

[11] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," in *Adv. Neural Inf. Process. Syst.*, 2021, pp. 8780–8794.

[12] T. Karras, M. Aittala, T. Aila, and S. Laine, "Elucidating the design space of diffusion-based generative models," in *Adv. Neural Inf. Process. Syst.*, 2022, pp. 26 565–26 577.

[13] F. Bao, C. Li, J. Zhu, and B. Zhang, "Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models," in *10th Int. Conf. Learn. Rep.*, 2022.

[14] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 10 684–10 695.

[15] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler, and K. Kreis, "Align your latents: High-resolution video synthesis with latent diffusion models," in *2023 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2023, pp. 22 563–22 575.

[16] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 9, pp. 10 850–10 869, 2023.

[17] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *2023 IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2023, pp. 4195–4205.

[18] N. Ma, M. Goldstein, M. S. Albergo, N. M. Boffi, E. Vanden-Eijnden, and S. Xie, "Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers," in *Proc. Europ. Conf. Comp. Visi.*, 2024, pp. 23–40.

[19] I. Azangulov, G. Deligiannidis, and J. Rousseau, "Convergence of diffusion models under the manifold hypothesis in high-dimensions," *arXiv preprint arXiv:2409.18804*, 2024.

[20] H. Cui, C. Pehlevan, and Y. M. Lu, "A solvable model of learning generative diffusion: theory and insights," in *Adv. Neural Inf. Process. Syst.*, 2025, pp. 5253–5296.

[21] G. Loaiza-Ganem, B. L. Ross, R. Hosseinzadeh, A. L. Caterini, and J. C. Cresswell, "Deep generative models through the lens of the manifold hypothesis: A survey and new connections," *Transactions on Machine Learning Research*, 2024.

[22] M. Chen, K. Huang, T. Zhao, and M. Wang, "Score approximation, estimation and distribution recovery of diffusion models on low-dimensional data," in *Proc. 39th Int. Conf. Mach. Learn.*, 2023, pp. 4672–4712.

[23] P. Pope, C. Zhu, A. Abdelkader, M. Goldblum, and T. Goldstein, "The intrinsic dimension of images and its impact on learning," in *9th Int. Conf. Learn. Rep.*, 2021.

[24] P. Wang, H. Zhang, Z. Zhang, S. Chen, Y. Ma, and Q. Qu, "Diffusion models learn low-dimensional distributions via subspace clustering," in *ICLR 2025 Workshop on Deep Generative Model in Machine Learning: Theory, Principle and Efficacy*, 2025.

[25] R. Tang and Y. Yang, "Adaptivity of diffusion models to manifold structures," in *Proc. Int. Conf. Artif. Intell. Stat.*, 2024, pp. 1648–1656.

[26] V. De Bortoli, E. Mathieu, M. Hutchinson, J. Thornton, Y. W. Teh, and A. Doucet, "Riemannian score-based generative modelling," in *Adv. Neural Inf. Process. Syst.*, 2022, pp. 2406–2422.

[27] A. Kumar and V. M. Patel, "Learning on the manifold: Unlocking standard diffusion transformers with representation encoders," *arXiv preprint arXiv:2602.10099*, 2026.

[28] P. Sun, Y. Jiang, S. Chen, S. Zhang, B. Peng, P. Luo, and Z. Yuan, "Autoregressive model beats diffusion: Llama for scalable image generation," *arXiv preprint arXiv:2406.06525*, 2024.

[29] Y. Liu, L. Qu, H. Zhang, X. Wang, Y. Jiang, Y. Gao, H. Ye, X. Li, S. Wang, D. K. Du *et al.*, "Detailflow: 1d coarse-to-fine autoregressive image generation via next-detail prediction," *arXiv preprint arXiv:2505.21473*, 2025.

[30] H. Chang, H. Zhang, L. Jiang, C. Liu, and W. T. Freeman, "Maskgit: Masked generative image transformer," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 11 305–11 315.

[31] Z. You, J. Ou, X. Zhang, J. Hu, J. ZHOU, and C. Li, "Effective and efficient masked image generation models," in *Proc. 42th Int. Conf. Mach. Learn.*, 2025, pp. 72 730–72 746.

[32] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 12 873–12 883.

[33] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *Adv. Neural Inf. Process. Syst.*, 2017, pp. 6309–6318.

[34] L. Zhu, F. Wei, Y. Lu, and D. Chen, "Scaling the codebook size of vqgan to 100,000 with a utilization rate of 99%," in *Adv. Neural Inf. Process. Syst.*, 2024, pp. 12 612–12 635.

[35] F. Shi, Z. Luo, Y. Ge, Y. Yang, Y. Shan, and L. Wang, "Scalable image tokenization with index backpropagation quantization," in *2025 IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2025, pp. 16 037–16 046.

[36] J. Xiong, G. Liu, L. Huang, C. Wu, T. Wu, Y. Mu, Y. Yao, H. Shen, Z. Wan, J. Huang *et al.*, "Autoregressive models in vision: A survey," *Transactions on Machine Learning Research*, 2025.

[37] B. Zheng, N. Ma, S. Tong, and S. Xie, "Diffusion transformers with representation autoencoders," in *14th Int. Conf. Learn. Rep.*, 2026.

[38] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever, "Consistency models," in *Proc. 40th Int. Conf. Mach. Learn.*, 2023, pp. 32 211–32 252.

[39] D. Xue, Z. Zhu, and J. Hou, "Diffusion image generation with explicit modeling of data manifold geometry," in *Proc. Europ. Conf. Comp. Visi.*, 2026.

[40] S. Yu, S. Kwak, H. Jang, J. Jeong, J. Huang, J. Shin, and S. Xie, "Representation alignment for generation: Training diffusion transformers is easier than you think," in *13th Int. Conf. Learn. Rep.*, 2025.

[41] X. Leng, J. Singh, Y. Hou, Z. Xing, S. Xie, and L. Zheng, "Repa-e: Unlocking vae for end-to-end tuning of latent diffusion transformers," in *2025 IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2025, pp. 18 262–18 272.

[42] Z. Ma, L. Wei, S. Wang, S. Zhang, and Q. Tian, "Deco: Frequency-decoupled pixel diffusion for end-to-end image generation," in *2026 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2026, pp. 43 600–43 610.

[43] Z. Geng, M. Deng, X. Bai, J. Z. Kolter, and K. He, "Mean flows for one-step generative modeling," in *Adv. Neural Inf. Process. Syst.*, 2025, pp. 75 460–75 482.

[44] Z. Geng, Y. Lu, Z. Wu, E. Shechtman, J. Z. Kolter, and K. He, "Improved mean flows: On the challenges of fastforward generative models," in *2026 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2026, pp. 30 467–30 476.

[45] Y. Huang, S.-H. Wang, A. L. Bertozzi, and B. Wang, "RMFlow: Refined mean flow by a noise-injection step for multimodal generation," in *14th Int. Conf. Learn. Rep.*, 2026.

[46] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," in *10th Int. Conf. Learn. Rep.*, 2022.

[47] J. Ho and T. Salimans, "Classifier-free diffusion guidance," in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[48] T. Karras, M. Aittala, T. Kynkäänniemi, J. Lehtinen, T. Aila, and S. Laine, "Guiding a diffusion model with a bad version of itself," in *Adv. Neural Inf. Process. Syst.*, 2024, pp. 52 996–53 021.

[49] L. Yu, J. Lezama, N. B. Gundavarapu, L. Versari, K. Sohn, D. Minnen, Y. Cheng, A. Gupta, X. Gu, A. G. Hauptmann, B. Gong, M.-H. Yang, I. Essa, D. A. Ross, and L. Jiang, "Language model beats diffusion - tokenizer is key to visual generation," in *12th Int. Conf. Learn. Rep.*, 2024.

[50] A. Lou, C. Meng, and S. Ermon, "Discrete diffusion modeling by estimating the ratios of the data distribution," in *Proc. 40th Int. Conf. Mach. Learn.*, 2024, pp. 32 819–32 848.

[51] J. Shi, K. Han, Z. Wang, A. Doucet, and M. K. Titsias, "Simplified and generalized masked diffusion for discrete data," in *Adv. Neural Inf. Process. Syst.*, 2024, pp. 103 131–103 167.

[52] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. van den Berg, "Structured denoising diffusion models in discrete state-spaces," in *Adv. Neural Inf. Process. Syst.*, 2021, pp. 17 981–17 993.

[53] S. Sahoo, M. Arriola, Y. Schiff, A. Gokaslan, E. Marroquin, J. Chiu, A. Rush, and V. Kuleshov, "Simple and effective masked diffusion language models," in *Adv. Neural Inf. Process. Syst.*, 2024, pp. 130 136–130 184.

[54] J. Jo and S. J. Hwang, "Continuous diffusion model for language modeling," in *Adv. Neural Inf. Process. Syst.*, 2025, pp. 97 394–97 430.

[55] K. Tian, Y. Jiang, Z. Yuan, B. Peng, and L. Wang, "Visual autoregressive modeling: scalable image generation via next-scale prediction," in *Adv. Neural Inf. Process. Syst.*, 2024, pp. 84 839–84 865.

[56] A. Zheng, X. Wen, X. Zhang, C. Ma, T. Wang, G. YU, X. Zhang, and X. QI, "Vision foundation models as effective visual tokenizers for autoregressive generation," in *Adv. Neural Inf. Process. Syst.*, 2025, pp. 62 656–62 675.

[57] F. Mentzer, D. Minnen, E. Agustsson, and M. Tschannen, "Finite scalar quantization: Vq-vae made simple," in *12th Int. Conf. Learn. Rep.*, vol. 2024, 2024, pp. 51 772–51 783.

[58] Y. Zhao, Y. Xiong, and P. Kraehenbuehl, "Image and video tokenization with binary spherical quantization," in *13th Int. Conf. Learn. Rep.*, 2025, pp. 90 844–90 868.

[59] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *2018 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 586–595.

[60] X. Ma, F. Zhao, P. Ling, H. Qiu, Z. Wei, H. Yu, J. Huang, Z. Zeng, and L. Ma, "Towards better & faster autoregressive image generation: From the perspective of entropy," in *Adv. Neural Inf. Process. Syst.*, 2025, pp. 31 466–31 497.

[61] J. Yang, Z. Geng, X. Ju, Y. Tian, and Y. Wang, "Representation fréchet loss for visual generation," *arXiv:2604.28190*, 2026. [Online]. Available: https://arxiv.org/abs/2604.28190

[62] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2009, pp. 248–255.

[63] M. Kang, J.-Y. Zhu, R. Zhang, J. Park, E. Shechtman, S. Paris, and T. Park, "Scaling up gans for text-to-image synthesis," in *2023 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2023, pp. 10 124–10 134.

[64] T. Kynkäänniemi, M. Aittala, T. Karras, S. Laine, T. Aila, and J. Lehtinen, "Applying guidance in a limited interval improves sample and distribution quality in diffusion models," in *Adv. Neural Inf. Process. Syst.*, 2024, pp. 122 458–122 483.

[65] D. Lee, C. Kim, S. Kim, M. Cho, and W.-S. Han, "Autoregressive image generation using residual quantization," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 11 523–11 532.

[66] E. Hoogeboom, J. Heek, and T. Salimans, "Simple diffusion: End-to-end diffusion for high resolution images," in *Proc. 40th Int. Conf. Mach. Learn.*, 2023, pp. 13 213–13 232.

[67] T. Li, Y. Tian, H. Li, M. Deng, and K. He, "Autoregressive image generation without vector quantization," in *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 56 424–56 445.

[68] Q. Yu, Q. Liu, J. He, X. Zhang, Y. Liu, L.-C. Chen, and X. Chen, "Autoregressive image generation with masked bit modeling," in *Porc. 43th Int. Conf. Mach. Learn.*, 2026.

[69] Z. Mao, M. Huang, Y. Lin, Q. Wang, L. Zhang, and Y. Zhang, "Toward accurate image generation via dynamic generative image transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 48, no. 5, pp. 5910–5927, 2026.

[70] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *7th Int. Conf. Learn. Rep.*, 2019.

[71] A. Sauer, K. Schwarz, and A. Geiger, "Stylegan-xl: Scaling stylegan to large diverse datasets," in *ACM SIGGRAPH 2022 Conference Proceedings*, 2022.

[72] S. Wang, Z. Tian, W. Huang, and L. Wang, "Ddt: Decoupled diffusion transformer," in *2026 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2026, pp. 40 633–40 642.

[73] S. Chen, C. Ge, S. Zhang, P. Sun, and P. Luo, "Pixelflow: Pixel-space generative models with flow," *arXiv preprint arXiv:2504.07963*, 2025.

[74] T. Li and K. He, "Back to basics: Let denoising generative models denoise," in *2026 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2026, pp. 36 115–36 125.

[75] J. Yu, X. Li, J. Y. Koh, H. Zhang, R. Pang, J. Qin, A. Ku, Y. Xu, J. Baldridge, and Y. Wu, "Vector-quantized image modeling with improved VQGAN," in *10th Int. Conf. Learn. Rep.*, 2022.

[76] Y. Lu, S. Lu, Q. Sun, H. Zhao, Z. Jiang, X. Wang, T. Li, Z. Geng, and K. He, "One-step latent-free image generation with pixel mean flows," *arXiv preprint arXiv:2601.22158*, 2026.

[77] L. Degeorge, A. Ghosh, N. Dufour, D. Picard, and V. Kalogeiton, "How far can we go with imagenet for text-to-image generation?" *arXiv*, 2025.

[78] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei, "Scaling instruction-finetuned language models," 2022. [Online]. Available: https://arxiv.org/abs/2210.11416

[79] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2022, pp. 10 684–10 695.

[80] J. Chen, J. YU, C. GE, L. Yao, E. Xie, Z. Wang, J. Kwok, P. Luo, H. Lu, and Z. Li, "PixArt-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis," in *12th Int. Conf. Learn. Rep.*, 2024, pp. 57 611–57 640.

[81] J. Chen, C. Ge, E. Xie, Y. Wu, L. Yao, X. Ren, Z. Wang, P. Luo, H. Lu, and Z. Li, "Pixart-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation," in *Proc. Europ. Conf. Comp. Visi.*, A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, and G. Varol, Eds., 2025, pp. 74–91.

[82] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, "Sdxl: Improving latent diffusion models for high-resolution image synthesis," in *12th Int. Conf. Learn. Rep.*, vol. 2024, 2024, pp. 1862–1874.
[83] P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, F. Boesel, D. Podell, T. Dockhorn, Z. English, and R. Rombach, "Scaling rectified flow transformers for high-resolution image synthesis," in *Porc. 41th Int. Conf. Mach. Learn.*, 2024.
[84] E. Xie, J. Chen, J. Chen, H. Cai, H. Tang, Y. Lin, Z. Zhang, M. Li, L. Zhu, Y. Lu, and S. Han, "Sana: Efficient high-resolution text-to-image synthesis with linear diffusion transformers," in *13th Int. Conf. Learn. Rep.*, 2025, pp. 20 383–20 407.
[85] B. F. Labs, S. Batifol, A. Blattmann, F. Boesel, S. Consul, C. Diagne, T. Dockhorn, J. English, Z. English, P. Esser, S. Kulal, K. Lacey, Y. Levi, C. Li, D. Lorenz, J. Müller, D. Podell, R. Rombach, H. Saini, A. Sauer, and L. Smith, "Flux.1 kontext: Flow matching for in-context image generation and editing in latent space," 2025.
[86] L. Zhuo, R. Du, H. Xiao, Y. Li, D. Liu, R. Huang, W. Liu, X. Zhu, F.-Y. Wang, Z. Ma, X. Luo, Z. Wang, K. Zhang, L. Zhao, S. Liu, X. Yue, W. Ouyang, Y. Qiao, H. Li, and P. Gao, "Lumina-next : Making lumina-t2x stronger and faster with next-dit," in *Adv. Neural Inf. Process. Syst.*, 2024. [Online]. Available: https://openreview.net/forum?id=ieYdf9TZ2u
[87] D. Ghosh, H. Hajishirzi, and L. Schmidt, "Geneval: an object-focused framework for evaluating text-to-image alignment," in *Adv. Neural Inf. Process. Syst.*, Red Hook, NY, USA, 2023.
[88] R. Vershynin, *High-dimensional probability: An introduction with applications in data science*. Cambridge university press, 2018, vol. 47.
[89] B. Efron, "Tweedie's formula and selection bias," *Journal of the American Statistical Association*, vol. 106, no. 496, pp. 1602–1614, 2011.
[90] M. J. Wainwright, *High-dimensional statistics: A non-asymptotic viewpoint*. Cambridge University Press, 2019, vol. 48.
[91] J. Park and K. Muandet, "Towards empirical process theory for vector-valued functions: Metric entropy of smooth function classes," in *International Conference on Algorithmic Learning Theory*. PMLR, 2023, pp. 1216–1260.
[92] R. M. Dudley, *Uniform Central Limit Theorems*, 2nd ed., ser. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2014.

Duoduo Xue received the BS degree from Xidian University, Xi'an, China, in 2020, and the PhD degree from Shanghai Jiao Tong University, Shanghai, China, in 2025. She is currently a postdoctoral researcher at City University of Hong Kong. Her research interests include signal processing and machine learning.

Zhiyu Zhu is an assistant professor in the department of computer science at City University of Hong Kong (DongGuan). He received the B.E. and M.E. degrees in Mechatronic Engineering, both from Harbin Institute of Technology, in 2017 and 2019, respectively. He has also received a Ph.D. degree in computer science from the City University of Hong Kong in 2023, where he currently holds a postdoctoral position. His research interests include generative models and computer vision.

Junhui Hou is a Professor with the Department of Computer Science, City University of Hong Kong. His research interests include multidimensional visual computing, such as light field, hyperspectral, geometry, and event data. He received the Early Career Award from the Hong Kong Research Grants Council in 2018, the Excellent Young Scientists Fund from NSFC in 2024, and the IEEE TIP Best Paper Award in 2025. He is serving as a Senior Area Editor for IEEE TIP and an Associate Editor for IEEE TVCG and TMM. He served as an Associate Editor for IEEE TIP and TCSVT.

Metric Entropy-Driven Explicit Manifold Modeling for Diffusion Image Generation (Supplementary Material)

Duoduo Xue, *Member, IEEE,* Zhiyu Zhu, Junhui Hou *Senior Member, IEEE*

S1. Ablation Study on Sampling Parameters

Table S-1 reports the performance of different sampling parameters by generating 2K and 50K images without classifier-free guidance. We perform inference with MIND-B model trained for 80 epochs. We fixed $\rho_1 = 1.0$ and investigated the impact of different sampling schedules, sampling thresholds (Top-p/Top-k), and ρ_2. We find that the Linear schedule generally achieves superior FID and IS compared to cosine-based variants. Crucially, Top-p and Top-k are important parameters for performance. Expanding the active window (e.g., from Top-100 to Top-200) significantly impacts the FID and IS. This sensitivity confirms that our model maintains a healthy, broad utilization of the vocabulary without gradient starvation. Additionally, the 50K-image evaluation confirms that applying greedy sampling during the last 10% of the generative steps (ρ_2=0.1) effectively refines the final image details, improving FID from 26.06 to 25.75.

Next, we generate 50K samples with MIND-S with different ρ_1, which shows that the threshold ρ_1 serves as a balance of FID and recall. Disabling the soft phase ($\rho_1 = 1.0$) yields optimal standard FID and Inception scores, while enabling it ($\rho_1 = 0.7$) drastically improves precision and recall by preserving diversity in early steps. This mechanism empowers users to flexibly control the trade-off between local sharpness and spatial coherence. Moreover, as shown in Table S-2, our ablation studies across different embedding dimensions and diffusion scaling coefficients demonstrate that incorporating guidance at the logit level (*i.e.*, the output of the network $\boldsymbol{s}_\phi$) yields superior generative performance compared to applying it at the intermediate feature level (*i.e.*, the features extracted before the final DiT block). Since the "Feature" level inherently retains all continuous architectural benefits (including our dual-branch high-frequency embeddings), the strict superiority of "Logits" guidance confirms that explicitly modeling the discrete data manifold is the genuine driver of our performance breakthrough.

S2. Experiment Settings

A. Training Configurations of MIND

Table S-3 shows the network architecture, training, and inference configurations of MIND. For inference, we consistently adopt the same 250 sampling steps as continuous baselines like DiT and SiT to ensure a strictly fair comparison of generative capability. Table S-4 shows the training configurations of MIND for system-level comparison. The diffusion and noise settings, embedding and vocabulary settings, and inference parameters of MIND-B(-G) and MIND-XL(-G) keep the same as MIND-B in Table S-3 without specification. Moreover, MIND-XL-G is first trained for 1,350 epochs following Table S-4. In the subsequent training phase, the timestep range is constrained to $[0.05, 0.55]$, with the learning rate set to 5×10^{-5} for the first 100 epochs and 1×10^{-5} for the remaining epochs.

B. Experiment Settings of Section III-A

To ensure a rigorous and fair comparison, we strictly align the trainable parameters of both projection modules to approximately 196K. For the continuous projection, we employ a two-layer MLP with a hidden dimension of 4468 to map 16-dimensional VAE patch features to the 6-dimensional latent. The discrete path utilizes an embedding layer and a linear decoder operating over a vocabulary size of 16384. To enforce the hypersphere constraint, the 6-dimensional latent space is partitioned into three coordinate pairs, with ℓ_2-normalization applied independently within each 2D subspace. The optimization is conducted on a single ImageNet validation image, we apply random resized cropping (256×256) and horizontal flipping to generate a batch of 8 augmented views to prevent trivial memorization. Both models are trained for 1000 iterations using the AdamW optimizer with a learning rate of 0.01. The continuous projection minimizes the Mean Squared Error (MSE) against the target VAE latent, while the discrete projection is supervised via Cross-Entropy loss against the target token indices.

S3. Proof of Section IV

A. Preliminary

Definitions 1–2 give the definitions of metric entropy and Hölder space.

Definition 1 (Metric Entropy [88]): The covering number $\mathcal{N}(\mathcal{Q}, \epsilon)$ measures the complexity of a set $\mathcal{Q}$. The logarithm of the covering numbers $\log \mathcal{N}(\mathcal{Q}, \epsilon)$ is called the metric entropy of $\mathcal{Q}$.

Definition 2 (Hölder Space): Define $\mathcal{C}^s(U)$ as the Hölder space of functions on a local manifold patch $U \subseteq \mathcal{M}$ with s-order differentiability and bounded functional norm. If this norm is strictly bounded by a constant M, we denote the corresponding subset as $\mathcal{C}^s_M(U)$.

TABLE S-1: **Ablation study of sampling configurations.** All models are evaluated without guidance.

Model	Img-num	Schedule	Top-p	Top-k	ρ_1	ρ_2	FID ↓	IS ↑	Prec. ↑	Rec. ↑
MIND-B	2K	**Linear**	**0.3**	**200**	1.0	0.1	**41.06**	54.02	0.538	**0.635**
MIND-B	2K	Linear	0.3	100	1.0	0.1	41.13	**54.46**	**0.546**	**0.635**
MIND-B	2K	Linear	0.8	100	1.0	0.1	44.10	46.91	0.502	0.630
MIND-B	2K	Linear	0.8	200	1.0	0.1	45.32	46.33	0.504	0.631
MIND-B	2K	Shift-Cos	0.3	200	1.0	0.1	43.23	50.67	0.519	0.632
MIND-B	2K	Shift-Cos	0.3	100	1.0	0.1	44.55	50.05	0.511	0.624
MIND-B	2K	Shift-Cos	0.8	100	1.0	0.1	45.10	47.65	0.478	0.623
MIND-B	2K	Shift-Cos	0.8	200	1.0	0.1	45.15	46.59	0.495	0.631
MIND-B	2K	Cosine	0.8	100	1.0	0.1	44.51	47.26	0.515	0.633
MIND-B	50K	Linear	0.8	100	1.0	0.0	26.06	50.11	0.51	0.59
MIND-B	50K	Linear	0.8	100	1.0	0.1	25.75	49.51	0.52	**0.60**
MIND-B	50K	Linear	0.3	200	1.0	0.1	**22.73**	**56.17**	**0.55**	0.58
MIND-S	50K	Linear	0.8	100	1.0	0.1	**34.76**	**36.17**	0.43	0.58
MIND-S	50K	Linear	0.8	100	0.7	0.1	40.72	31.51	**0.48**	**0.61**

TABLE S-2: **Ablation study on Guidance Space (Logits vs. Feature), embedding dimension (L) and diffusion scaling coefficients (c_1, c_2).** All models are evaluated on 2,000 generated samples at cfg = 4.0. Bold values indicate the best performance.

L	d_{sub}	c_1 (Noise)	c_2 (Signal)	Guidance Space	FID ↓	IS ↑	Precision ↑	Recall ↑
8	2	0.6	1.0	Logits	125.40	17.21	0.26	0.29
				Feature	122.71	18.87	0.26	0.26
8	2	1.0	1.0	Logits	206.19	7.52	0.26	0.28
				Feature	233.21	8.30	0.30	0.23
16	4	0.6	1.0	Logits	**34.91**	**119.14**	**0.81**	**0.56**
				Feature	37.62	100.91	0.76	0.51
32	8	0.6	1.0	Logits	38.95	89.63	0.71	0.53
				Feature	42.90	70.55	0.66	0.49
64	16	0.6	1.0	Logits	43.03	72.49	0.63	0.51
				Feature	48.41	56.79	0.56	0.45
64	16	0.8	1.0	Logits	44.07	67.24	0.59	0.51
				Feature	49.89	50.87	0.53	0.45
128	32	1.0	1.0	Logits	52.03	51.35	0.50	0.46
				Feature	61.24	37.93	0.42	0.39

Next, we analyze with one token for brevity and will be extended to the full sequence in Section S3-C. Definitions 3–5 define the data priors, diffusion process, and loss functions of MIND and CDM.

Definition 3 (Data prior of MIND): Denote the data manifold of MIND as a discrete codebook supported on a hypersphere of radius $R > 0$ and dimension $L > 0$,

$$\mathcal{A} = \{\boldsymbol{a}_1, \dots, \boldsymbol{a}_V\} \subset \mathbb{S}^{L-1}(R), \tag{S-1}$$

where $\mathbb{S}^{L-1}(R) := \{\boldsymbol{x} \in \mathbb{R}^L : \|\boldsymbol{x}\| = R\}$. The data prior for MIND is defined as

$$p_0^{\text{disc}}(d\boldsymbol{x}_0) = \sum_{v=1}^{V} \varpi_v \delta_{\boldsymbol{a}_v}(d\boldsymbol{x}_0), \quad \varpi_v > 0, \sum_{v=1}^{V} \varpi_v = 1, \tag{S-2}$$

where $\delta_{\boldsymbol{a}_v}$ denotes the Dirac measure centered at the anchor $\boldsymbol{a}_v \in \mathbb{S}^{L-1}(R)$. For any Borel measurable set $A \subseteq \mathbb{R}^L$, it is defined as

$$\delta_{\boldsymbol{a}_v}(A) = \begin{cases} 1, & \text{if } \boldsymbol{a}_v \in A \\ 0, & \text{if } \boldsymbol{a}_v \notin A. \end{cases} \tag{S-3}$$

Remark 2: For any bounded measurable function $f : \mathbb{R}^d \to \mathbb{R}$, we have $\int_{\mathbb{R}^d} f(\boldsymbol{x}_0)\delta_{\boldsymbol{a}_v}(d\boldsymbol{x}_0) = f(\boldsymbol{a}_v)$. Consequently, the discrete prior $p_0^{\text{disc}}(d\boldsymbol{x}_0)$ defines a convex combination of Dirac measures. The anchor $\boldsymbol{a}_v = \mathcal{P}_{\boldsymbol{\theta}}(v)$ is determined by the learnable $\mathcal{P}_{\boldsymbol{\theta}}$, which encodes discrete tokens into continuous latent and plays the same role as the VAE encoder in the CDM. We analyze with the same setting that the encoders are pretrained and focus on the learning of the score function.

Definition 4 (Forward Diffusion Process): We adopt a unified Variance Preserving schedule for $t \in (0, 1]$. Let $\boldsymbol{x}_0$ denote a generic clean latent ($\boldsymbol{x}_0 \in \mathbb{R}^L$ for MIND, or $\boldsymbol{x}_0 \in \mathbb{R}^{L_c}$ for CDM), the forward process is defined as

$$\boldsymbol{x}_t = \alpha_t \boldsymbol{x}_0 + \sigma_t \boldsymbol{w}, \tag{S-4}$$

where $\alpha_t := c_2\sqrt{1-t}, \sigma_t := c_1\sqrt{t}, \boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_m)$, $c_1, c_2 > 0$ are scheduling constants, and $m \in \{L, L_c\}$ denotes the respective latent dimension. Let $\varphi_\sigma(\boldsymbol{x}) := (2\pi\sigma^2)^{-m/2} \exp(-\|\boldsymbol{x}\|^2/2\sigma^2)$ denote the Gaussian smoothing kernel, yielding the transition density $p_{0t}(\boldsymbol{x}_t \mid \boldsymbol{x}_0) = \varphi_{\sigma_t}(\boldsymbol{x}_t - \alpha_t \boldsymbol{x}_0)$.

Definition 5 (Loss Function): CDM optimizes a network $\boldsymbol{s}_{\boldsymbol{\phi}}^{\text{cont}} : \mathbb{R}^{L_c} \to \mathbb{R}^{L_c}$ to minimize

$$\mathcal{L}_{\text{CDM}}(\boldsymbol{s}_{\boldsymbol{\phi}}^{\text{cont}}) = \mathbb{E}_{\boldsymbol{x}_0, t}\left[\|\sigma_t \boldsymbol{s}_{\boldsymbol{\phi}}^{\text{cont}}(\boldsymbol{x}_t, t) + (\boldsymbol{x}_t - \alpha_t \boldsymbol{x}_0)\|^2\right]. \tag{S-5}$$

Let $\tilde{\boldsymbol{x}}_{0t} = \boldsymbol{s}_{\boldsymbol{\phi}}(\boldsymbol{x}_t, t) \in \mathbb{R}^V$ denote the predicted unnormalized logits of MIND. The denoised latent is $\dot{\boldsymbol{x}}_{0t} =$

TABLE S-3: Summary of Training Configurations and Inference Parameters.

Category	MIND-B	MIND-S
1. Diffusion & Noise Settings		
Noise Scale Factor c_1	0.6	0.8
Signal Scale Factor c_2	1.0	1.0
Training Timestep Range (t)	$t \in [0.05, 0.8]$	$t \in [0.05, 0.8]$
2. Embedding & Vocabulary		
Vocabulary Size V	16,384	8192
Embedding Dimension L	16	8
Embedding Subspace	4	2
3. Network Architecture		
Total Parameters(M)	130.48	35.21
Number of Blocks	14	14
Hidden Size	768	384
Attention Heads	12	6
Condition Embedding Dimension	128	128
4. Training & Optimization		
Optimizer	DeepSpeed AdamW	
Base Learning Rate	1.0×10^{-3}	1.0×10^{-3}
Batch Size	1024	2048
Training Epochs	80	80
5. Inference		
Timestep schedule	Linear	Linear
Temperature τ	0.99	0.99
Sampling steps	250	250
η	0.99	0.99
(Top-p, Top-k, ρ_1, ρ_2)($cfg = 1.0$)	(200, 0.3, 1.0, 0.1)	(100, 0.8, 0.7, 0.1)
(Top-p, Top-k, ρ_1, ρ_2)($cfg = 1.5$)	(100, 0.8, 0.9, 0.1)	(100, 0.8, 0.8, 0.01)
(Top-p, Top-k, ρ_1, ρ_2)($cfg = 2.0$)	(100, 0.8, 0.9, 0.1)	(300, 0.8, 0.7, 0.1)

TABLE S-4: Training and Inference Parameters in System-level Comparison.

Setting	Batch Size	Base learning rate	Epoch	Top-k	Top-p	cfg	$(C_{\max} - C_{\min})$
MIND-B	2048	3e-4	1000	100	0.8	5.0	0.6 - 0.3
MIND-B-G	1536	3e-4	1700	100	0.8	3.0	0.8 - 0.4
MIND-XL	2048	2e-4	1000	200	0.8	2.5	0.8 - 0.3
MIND-XL-G	2048	2e-4	1500	100	0.8	2.25	0.8 - 0.3

$\sum_{j=1}^{V_0} \hat{\pi}_j(\boldsymbol{x}_t)\boldsymbol{a}_j \in \mathbb{R}^L$, where $\hat{\pi}_j(\boldsymbol{x}_t) = \frac{\exp(\tilde{x}_{0t}^{(j)})}{\sum_{i=1}^{V_0}\exp(\tilde{x}_{0t}^{(i)})}$ with $0 < V_0 \le V$. Given the ground-truth $\boldsymbol{x}_0 = \boldsymbol{a}_v$ and weight $\lambda > 0$, MIND optimizes

$$\mathcal{L}_{\mathrm{MIND}}(\boldsymbol{s}_{\boldsymbol{\phi}}) = \mathbb{E}_{\boldsymbol{x}_0,t}[\underbrace{-\log \hat{\pi}_v(\boldsymbol{x}_t)}_{\mathcal{L}_{\mathrm{CE}}} + \lambda \underbrace{\|\dot{\boldsymbol{x}}_{0t} - \boldsymbol{x}_0\|^2}_{\mathcal{L}_{\mathrm{MSE}}}]. \quad \text{(S-6)}$$

Remark 3: The above two loss functions are mathematically equivalent to learning score function. Specifically, the objective $\mathcal{L}_{\mathrm{CDM}}$ explicitly trains a network to fit a scaled score function, the objective $\mathcal{L}_{\mathrm{MIND}}$ optimized to reconstruct the ground-truth signal $\boldsymbol{x}_0$ compels approximation of the score function. According to the Tweedie's formula, the posterior mean and the score function are strictly coupled via a deterministic affine transformation. Therefore, in our subsequent theoretical development, we unify the analysis by bounding the metric entropy and generalization properties of the hypothesis class associated with the expected posterior (or equivalently, the score function) without loss of generality.

Lemma 1 (Tweedie's Formula [89]): Let $p(\boldsymbol{x}_0)$ be the prior data distribution of the clean data $\boldsymbol{x}_0$, the marginal density of the noisy latent $\boldsymbol{x}_t$ is given by

$$p_t(\boldsymbol{x}_t) = \int_{\mathcal{M}} \varphi_{\sigma_t}(\boldsymbol{x}_t - \alpha_t \boldsymbol{x}_0) p(\boldsymbol{x}_0)\, d\boldsymbol{x}_0, \quad \text{(S-7)}$$

where $\varphi_{\sigma_t}(\boldsymbol{x})$ and α_t, σ_t is defined in Definition 4. Then $p_t \in C^\infty(\mathbb{R}^m)$ is an infinitely differentiable smooth function, and we have almost everywhere

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t) = \frac{\mathbb{E}[\alpha_t \boldsymbol{x}_0 \mid \boldsymbol{x}_t] - \boldsymbol{x}_t}{\sigma_t^2}. \quad \text{(S-8)}$$

B. Proof of Theorem 1

Now, we analyze the function complexity of the score functions approximated by MIND and CDM.

Lemma 2 (MIND Exact Score): For $t \in (0, 1]$, the exact score function of MIND admits a closed-form representation

$$\nabla \log p_t^{\mathrm{disc}}(\boldsymbol{x}_t) = -\frac{\boldsymbol{x}_t}{\sigma_t^2} + \frac{\alpha_t}{\sigma_t^2} \sum_{v=1}^{V} \pi_v(\boldsymbol{x}_t)\boldsymbol{a}_v, \quad \text{(S-9)}$$

where $\pi_v(\boldsymbol{x}_t) \propto \varpi_v \exp(-\|\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_v\|^2/2\sigma_t^2)$.

Proof: Given the discrete prior $p_0^{\mathrm{disc}} = \sum_{v=1}^{V} \varpi_v \delta_{\boldsymbol{a}_v}$, the marginal density of the noisy latent $\boldsymbol{x}_t \in \mathbb{R}^L$ is exactly a Gaussian mixture model $p_t^{\mathrm{disc}}(\boldsymbol{x}_t) = \sum_{v=1}^{V} \varpi_v \varphi_{\sigma_t}(\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_v)$. Applying Bayes' rule, the posterior assignment probability for the v-th anchor is

$$\pi_v(\boldsymbol{x}_t) := \mathbb{P}(\boldsymbol{x}_0 = \boldsymbol{a}_v \mid \boldsymbol{x}_t) = \frac{\varpi_v \varphi_{\sigma_t}(\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_v)}{\sum_{j=1}^{V} \varpi_j \varphi_{\sigma_t}(\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_j)}. \quad \text{(S-10)}$$

Consequently, the conditional expectation analytically collapses to a convex combination of anchors $\mathbb{E}[\alpha_t \boldsymbol{x}_0 \mid \boldsymbol{x}_t] = \alpha_t \sum_{v=1}^{V} \pi_v(\boldsymbol{x}_t)\boldsymbol{a}_v$. Substituting this into Tweedie's formula (S-8) directly yields the closed-form score field. ∎

Lemma 3 proves that the noisy latent of MIND and CDM is supported on a compact domain with high probability.

Lemma 3 (Compact Domain for MIND and CDM): For any $\delta \in (0,1)$, under both the MIND and CDM, there exists a high-probability compact domain $\mathcal{K}_\delta \subset \mathbb{R}^m$ ($m = L$ for MIND, $m = L^c$ for CDM) of radius $R_\mathcal{K}$ such that the noisy latent satisfies $\boldsymbol{x}_t \in \mathcal{K}_\delta$ with probability at least $1-\delta$.

Proof: The forward diffusion process yields $\boldsymbol{x}_t = \alpha_t \boldsymbol{x}_0 + \sigma_t \boldsymbol{w}$, where $\boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_m)$ and $m \in \{L, L_c\}$. Let R_0 denote the absolute upper bound of the data prior, where $R_0 = R$ for MIND and $R_0 = R_\mathcal{M}$ for CDM with compact data manifold having bounded diameter $\mathrm{diam}(\mathcal{M}) \le 2R_\mathcal{M}$. Based on Thm. 2.26 of [90] and the 1-Lipschitz property of the Euclidean norm function $\|\cdot\|_2$, the standard Gaussian vector $\boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_m)$ satisfies

$$\mathbb{P}\left(\|\boldsymbol{w}\|_2 - \mathbb{E}[\|\boldsymbol{w}\|_2] \ge \delta'\right) \le \exp(-(\delta')^2/2), \tag{S-11}$$

for any $\delta' \ge 0$. By Jensen's inequality and the concavity of the square root, the expectation is upper-bounded via $\mathbb{E}[\|\boldsymbol{w}\|_2] \le \sqrt{\mathbb{E}[\|\boldsymbol{w}\|_2^2]} = \sqrt{m}$, which directly implies $\mathbb{P}\left(\|\boldsymbol{w}\|_2 - \sqrt{m} \ge \delta'\right) \le \exp(-(\delta')^2/2)$. Define δ such that $\exp(-(\delta')^2/2) = \delta$, which yields the deviation parameter $\delta' = \sqrt{2\ln(1/\delta)}$. Then

$$\mathbb{P}\left(\|\boldsymbol{w}\|_2 \le \sqrt{m} + \sqrt{2\ln(1/\delta)}\right) \ge 1-\delta. \tag{S-12}$$

Let $R_\delta := \sqrt{m} + \sqrt{2\ln(1/\delta)}$. By the triangle inequality, with probability at least $1-\delta$, $\boldsymbol{x}_t$ is strictly confined within a compact Euclidean ball $\mathcal{K}_\delta \subset \mathbb{R}^L$ of radius $R_\mathcal{K}$,

$$\|\boldsymbol{x}_t\|_2 \le \alpha_t \|\boldsymbol{x}_0\|_2 + \sigma_t \|\boldsymbol{w}\|_2 \le \alpha_t R_0 + \sigma_t R_\delta := R_\mathcal{K}. \tag{S-13}$$

∎

Lemma 4 proves that the posterior mean of MIND is Lipschitz with respect to its parameters.

Lemma 4 (Parametric Lipschitz for posterior mean of MIND): Let the parameters of the MIND posterior mean operator $f^*_{\mathrm{dist},\boldsymbol{\xi}}(\boldsymbol{x}_t) = \sum_{v=1}^{V} \pi_v(\boldsymbol{x}_t)\boldsymbol{a}_v$ be concatenated as $\boldsymbol{\xi} = \{\boldsymbol{a}_v, \ln \varpi_v\}_{v=1}^V \in \Xi \subset \mathbb{R}^{K\times L} \times \mathbb{R}^K$. Then there exists a finite constant $L_{\mathrm{map}} > 0$ such that for any $\delta \in (0,1)$ and $\sigma_t > 0$, with probability at least $1-\delta$, we have

$$\|f^*_{\mathrm{dist},\boldsymbol{\xi}_1} - f^*_{\mathrm{dist},\boldsymbol{\xi}_2}\|_\infty \le L_{\mathrm{map}} \cdot \|\boldsymbol{\xi}_1 - \boldsymbol{\xi}_2\|_2 \quad \forall \boldsymbol{\xi}_1, \boldsymbol{\xi}_2 \in \Xi. \tag{S-14}$$

Proof: We prove the lemma by showing that the parametric Jacobian matrix $\mathbf{J}_{\boldsymbol{\xi}} f^*_{\mathrm{dist}}(\boldsymbol{x}_t) := \nabla_{\boldsymbol{\xi}} f^*_{\mathrm{dist}}(\boldsymbol{x}_t)$ is uniformly bounded in the operator norm for all $\boldsymbol{x}_t \in \mathcal{K}_\delta$. Note that $f^*_{\mathrm{dist},\boldsymbol{\xi}}$ is denoted as f^*_{dist} for brevity. Recall the exact softmax responsibility for the k-th anchor from Lemma 2,

$$\pi_v(\boldsymbol{x}_t; \boldsymbol{\xi}) = \frac{\exp(\vartheta_v)}{\sum_{i=1}^{V} \exp(\vartheta_i)}, \tag{S-15}$$

where $\vartheta_v = -\frac{\|\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_v\|_2^2}{2\sigma_t^2} + \theta_v, \theta_v = \ln \varpi_v$.

Step 1: Jacobian with respect to θ_j. The partial derivative of the logit is $\frac{\partial \vartheta_v}{\partial \theta_j} = \delta_{vj}$. Using the quotient rule for the softmax function, the gradient of the assignment probability is $\frac{\partial \pi_v}{\partial \theta_j} = \pi_v(\delta_{vj} - \pi_j)$ as derived in Section S3-D. Applying the product rule to $f^*_{\mathrm{dist}}(\boldsymbol{x}_t)$ yields

$$\begin{aligned}\frac{\partial f^*_{\mathrm{dist}}(\boldsymbol{x}_t)}{\partial \theta_j} &= \sum_{v=1}^{V} \boldsymbol{a}_v \pi_v(\delta_{vj} - \pi_j) \\ &= \pi_j \boldsymbol{a}_j - \pi_j \sum_{v=1}^{V} \pi_v \boldsymbol{a}_v = \pi_j(\boldsymbol{a}_j - \boldsymbol{m}),\end{aligned} \tag{S-16}$$

where $\boldsymbol{m} = \sum_{v=1}^{V} \pi_v \boldsymbol{a}_v$ is the posterior mean. Since $\pi_j \in [0,1]$ and both $\|\boldsymbol{a}_j\|_2 = R$ and $\|\boldsymbol{m}\|_2 \le R$, the norm is strictly bounded by $\left\|\frac{\partial f^*_{\mathrm{dist}}}{\partial \theta_j}\right\|_2 \le 2R$.

Step 2: Jacobian with respect to the anchors $\boldsymbol{a}_j$. We compute the $L \times L$ Jacobian matrix $\mathbf{J}_{\boldsymbol{a}_j} f^*_{\mathrm{dist}}(\boldsymbol{x}_t)$. As derived in Section S3-D,

$$\mathbf{J}_{\boldsymbol{a}_j} f^*_{\mathrm{dist}}(\boldsymbol{x}_t) = \pi_j \mathbf{I}_L + \sum_{v=1}^{V} \boldsymbol{a}_v \left(\nabla_{\boldsymbol{a}_j} \pi_v\right)^\top. \tag{S-17}$$

The gradient of the ϑ_v with respect to $\boldsymbol{a}_j$ is non-zero only when $v = j$, given by $\nabla_{\boldsymbol{a}_j} \vartheta_j = \frac{\alpha_t}{\sigma_t^2}(\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_j)$. Consequently, the gradient of the softmax probability is $\nabla_{\boldsymbol{a}_j} \pi_v = \pi_v(\delta_{vj} - \pi_j)\frac{\alpha_t}{\sigma_t^2}(\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_j)$. Substituting this back into the sum yields a rank-1 update matrix

$$\sum_{v=1}^{V} \boldsymbol{a}_v \left(\nabla_{\boldsymbol{a}_j} \pi_v\right)^\top = \pi_j(\boldsymbol{a}_j - \boldsymbol{m})\frac{\alpha_t}{\sigma_t^2}(\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_j)^\top. \tag{S-18}$$

Thus, the complete Jacobian with respect to the anchor $\boldsymbol{a}_j$ is

$$\mathbf{J}_{\boldsymbol{a}_j} f^*_{\mathrm{dist}}(\boldsymbol{x}_t) = \pi_j \mathbf{I}_L + \frac{\alpha_t}{\sigma_t^2} \pi_j (\boldsymbol{a}_j - \boldsymbol{m})(\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_j)^\top. \tag{S-19}$$

By the triangle inequality, we have

$$\|\mathbf{J}_{\boldsymbol{a}_j} f^*_{\mathrm{dist}}(\boldsymbol{x}_t)\|_{\mathrm{op}} \le \pi_j + \frac{\alpha_t}{\sigma_t^2} \pi_j \|\boldsymbol{a}_j - \boldsymbol{m}\|_2 \|\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_j\|_2. \tag{S-20}$$

Given the compact hypersphere constraint, $\|\boldsymbol{a}_j - \boldsymbol{m}\|_2 \le 2R$. Since $\boldsymbol{x}_t$ is restricted to the high-probability Euclidean ball $\mathcal{K}_\delta$, we have $\|\boldsymbol{x}_t\|_2 \le R_\mathcal{K}$. Hence, $\|\boldsymbol{x}_t - \alpha_t \boldsymbol{a}_j\|_2 \le R_\mathcal{K} + \alpha_t R$. This yields

$$\|\mathbf{J}_{\boldsymbol{a}_j} f^*_{\mathrm{dist}}(\boldsymbol{x}_t)\|_{\mathrm{op}} \le 1 + \frac{2\alpha_t R}{\sigma_t^2}(R_\mathcal{K} + \alpha_t R). \tag{S-21}$$

Step 3: Uniform Spectral Bound and Global Lipschitz Continuity. We analytically extend the parameter space to the strictly convex Cartesian product $\Xi := \{\boldsymbol{a}_v \mid \|\boldsymbol{a}_v\|_2 \le R\}^V \times \mathbb{R}^V$. Since the posterior mean $\boldsymbol{m}$ remains a convex combination within this domain, the supremum bounds from Steps 1 and 2 hold globally over $\Xi \times \mathcal{K}_\delta$.

The total parametric Jacobian is the block matrix $\mathbf{J}_{\boldsymbol{\xi}} f^*_{\mathrm{dist}} = \left[\mathbf{J}_{\boldsymbol{a}_1} f^*_{\mathrm{dist}}, \ldots, \mathbf{J}_{\boldsymbol{a}_V} f^*_{\mathrm{dist}}, \frac{\partial f^*_{\mathrm{dist}}}{\partial \theta_1}, \ldots, \frac{\partial f^*_{\mathrm{dist}}}{\partial \theta_V}\right]$. For any

conformably partitioned unit vector $\boldsymbol{u} = (\boldsymbol{u}_{a_1}, \ldots, u_{\theta_V})$, applying the triangle and Cauchy-Schwarz inequalities yields

$$\|\mathbf{J}_{\boldsymbol{\xi}} f^*_{\text{dist}} \boldsymbol{u}\|_2 \leq \sum_{v=1}^{V} \|\mathbf{J}_{\boldsymbol{a}_v} f^*_{\text{dist}}\|_{\text{op}} \|\boldsymbol{u}_{a_v}\|_2 + \sum_{v=1}^{V} \left\| \frac{\partial f^*_{\text{dist}}}{\partial \theta_v} \right\|_2 |u_{\theta_v}|$$
$$\leq \left(\sum_{v=1}^{V} \|\mathbf{J}_{\boldsymbol{a}_v} f^*_{\text{dist}}\|^2_{\text{op}} + \sum_{v=1}^{V} \left\| \frac{\partial f^*_{\text{dist}}}{\partial \theta_v} \right\|_2^2 \right)^{1/2} \|\boldsymbol{u}\|_2. \quad \text{(S-22)}$$

Let $L_a = 1 + \frac{2\alpha_t R}{\sigma_t^2}(R_{\mathcal{K}} + \alpha_t R)$ and $L_\theta = 2R$. Taking the supremum over $\|\boldsymbol{u}\|_2 = 1$ establishes the uniform spectral bound

$$\sup_{\boldsymbol{\xi} \in \bar{\Xi}, \boldsymbol{x}_t \in \mathcal{K}_\delta} \|\mathbf{J}_{\boldsymbol{\xi}} f^*_{\text{dist}}(\boldsymbol{x}_t)\|_{\text{op}} \leq \sqrt{V(L_a^2 + L_\theta^2)} := L_{\text{map}} < \infty. \quad \text{(S-23)}$$

Because the Jacobian operator norm is uniformly bounded by a finite constant L_{map} across the convex domain Ξ, the mean value theorem directly implies the global Lipschitz continuity of $\boldsymbol{\xi} \mapsto f^*_{\text{dist}}$ with constant L_{map} over $\mathcal{K}_\delta$. ■

Lemma 5 (Metric Entropy Bound via Jacobian Operator Norm): Let $\mathcal{F}_{\text{disc}} = \{f^*_{\text{dist}}(\cdot; \boldsymbol{\xi}) \mid \boldsymbol{\xi} \in \Xi\}$ be the hypothesis class parameterized by $\Xi \subset \mathbb{R}^D$, and let $\mathcal{K}_\delta$ be the input domain. Suppose the parametric Jacobian matrix $\mathbf{J}_{\boldsymbol{\xi}} f^*_{\text{dist}}$ is uniformly bounded in the operator norm:

$$\sup_{\boldsymbol{\xi} \in \bar{\Xi}, \boldsymbol{x}_t \in \mathcal{K}_\delta} \|\mathbf{J}_{\boldsymbol{\xi}} f^*_{\text{dist}}(\boldsymbol{x}_t)\|_{\text{op}} \leq L_{\text{map}} < \infty. \quad \text{(S-24)}$$

Then, the functional covering number of $\mathcal{F}_{\text{disc}}$ in the norm $\|\cdot\|_\infty$ is bounded by the covering number of the parameter space Ξ in the norm $\|\cdot\|_2$:

$$\mathcal{N}(\mathcal{F}_{\text{disc}}, \varepsilon, \|\cdot\|_\infty) \leq \mathcal{N}\left(\Xi, \frac{\varepsilon}{L_{\text{map}}}, \|\cdot\|_2\right). \quad \text{(S-25)}$$

Proof: **Step 1: Point-wise Lipschitz bound via the mean value theorem.** For any fixed input $\boldsymbol{x}_t \in \mathcal{K}_\delta$, consider the mapping $\boldsymbol{\xi} \mapsto f^*_{\text{dist}}(\boldsymbol{x}_t; \boldsymbol{\xi})$. Since the parameter space Ξ is a convex set, we apply the multivariate mean value theorem for vector-valued functions. For any two parameters $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2 \in \Xi$, there exists a point $\tilde{\boldsymbol{\xi}}$ on the line segment connecting $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_2$ such that the Euclidean distance between the outputs is bounded by the operator norm of the Jacobian evaluated at $\tilde{\boldsymbol{\xi}}$:

$$\|f^*_{\text{dist}}(\boldsymbol{x}_t; \boldsymbol{\xi}_1) - f^*_{\text{dist}}(\boldsymbol{x}_t; \boldsymbol{\xi}_2)\|_2 \leq \|\mathbf{J}_{\tilde{\boldsymbol{\xi}}} f^*_{\text{dist}}(\boldsymbol{x}_t)\|_{\text{op}} \|\boldsymbol{\xi}_1 - \boldsymbol{\xi}_2\|_2. \quad \text{(S-26)}$$

By the given uniform bound condition, we have $\|\mathbf{J}_{\tilde{\boldsymbol{\xi}}} f^*_{\text{dist}}(\boldsymbol{x}_t)\|_{\text{op}} \leq L_{\text{map}}$ for all $\tilde{\boldsymbol{\xi}} \in \Xi$ and all $\boldsymbol{x}_t \in \mathcal{K}_\delta$. Thus, the point-wise mapping strictly satisfies:

$$\|f^*_{\text{dist}}(\boldsymbol{x}_t; \boldsymbol{\xi}_1) - f^*_{\text{dist}}(\boldsymbol{x}_t; \boldsymbol{\xi}_2)\|_2 \leq L_{\text{map}} \|\boldsymbol{\xi}_1 - \boldsymbol{\xi}_2\|_2, \forall \boldsymbol{x}_t \in \mathcal{K}_\delta. \quad \text{(S-27)}$$

Step 2: Transition to the functional supremum norm. In the context of hypothesis classes, the functional distance between two mappings $f_{\boldsymbol{\xi}_1}$ and $f_{\boldsymbol{\xi}_2}$ is defined by the supremum norm over the input domain $\mathcal{K}_\delta$:

$$\|f_{\boldsymbol{\xi}_1} - f_{\boldsymbol{\xi}_2}\|_\infty := \sup_{\boldsymbol{x}_t \in \mathcal{K}_\delta} \|f^*_{\text{dist}}(\boldsymbol{x}_t; \boldsymbol{\xi}_1) - f^*_{\text{dist}}(\boldsymbol{x}_t; \boldsymbol{\xi}_2)\|_2. \quad \text{(S-28)}$$

Taking the supremum over $\boldsymbol{x}_t \in \mathcal{K}_\delta$ on both sides of inequality (S-27), we obtain the global Lipschitz continuity of the functional mapping:

$$\|f_{\boldsymbol{\xi}_1} - f_{\boldsymbol{\xi}_2}\|_\infty \leq L_{\text{map}} \|\boldsymbol{\xi}_1 - \boldsymbol{\xi}_2\|_2. \quad \text{(S-29)}$$

Step 3: Forward mapping of the covering sets. Let $K = \mathcal{N}(\Xi, \frac{\varepsilon}{L_{\text{map}}}, \|\cdot\|_2)$ denote the minimal covering number of the parameter space. By definition, there exists a finite set of parameter centers $C_\Xi = \{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_K\}$ such that for any $\boldsymbol{\xi} \in \Xi$, there exists at least one center $\boldsymbol{\xi}_i \in C_\Xi$ satisfying:

$$\|\boldsymbol{\xi} - \boldsymbol{\xi}_i\|_2 \leq \frac{\varepsilon}{L_{\text{map}}}. \quad \text{(S-30)}$$

We define the corresponding set of functions mapped from these centers as $C_{\mathcal{F}} = \{f_{\boldsymbol{\xi}_1}, f_{\boldsymbol{\xi}_2}, \ldots, f_{\boldsymbol{\xi}_K}\}$. For any function $f_{\boldsymbol{\xi}} \in \mathcal{F}_{\text{disc}}$, we associate it with the center $f_{\boldsymbol{\xi}_i}$. Substituting the Euclidean distance bound into the functional Lipschitz inequality (S-29) yields:

$$\|f_{\boldsymbol{\xi}} - f_{\boldsymbol{\xi}_i}\|_\infty \leq L_{\text{map}} \left(\frac{\varepsilon}{L_{\text{map}}}\right) = \varepsilon. \quad \text{(S-31)}$$

This mathematically guarantees that $C_{\mathcal{F}}$ constitutes a valid ε-covering set for the function class $\mathcal{F}_{\text{disc}}$ in the supremum norm. Since the cardinality of $C_{\mathcal{F}}$ is at most K, the minimal covering number of the hypothesis class is fundamentally bounded by K:

$$\mathcal{N}(\mathcal{F}_{\text{disc}}, \varepsilon, \|\cdot\|_\infty) \leq K = \mathcal{N}\left(\Xi, \frac{\varepsilon}{L_{\text{map}}}, \|\cdot\|_2\right). \quad \text{(S-32)}$$

This completes the proof. ■

Based on the above lemmas, we prove Theorem 1.

Proof: **(1) MIND.** Let $\boldsymbol{\xi}$ denote the concatenated parameter vector dictating f^*_{disc} consist of the V latent anchors and the V routing logits, residing in a finite-dimensional parameter space $\Xi \subset \mathbb{R}^D$ with $D = V \times L + V$. Since the individual discrete anchors are strictly constrained to a hypersphere of radius R and the logits are norm-bounded by network regularization, the entire parameter space Ξ is compact and contained within a high-dimensional Euclidean ball $B_2^D(R_\Xi)$.

Based on Cor. 4.2.13 of [88], the metric entropy of this Euclidean parameter space is $\log \mathcal{N}(\Xi, \delta'', \|\cdot\|_2) \leq D \log\left(1 + \frac{2R_\Xi}{\delta''}\right)$. According to Lemma 4, the functional mapping $\boldsymbol{\xi} \mapsto f^*_{\text{disc}}$ is globally Lipschitz continuous with constant $L_{\text{map}} > 0$. Consequently, the hypothesis class $\mathcal{F}_{\text{disc}}$ is a Lipschitz image of the compact parameter space Ξ. Thus, its complexity is rigorously bounded by that of the scaled parameter space $L_{\text{map}}\Xi$. Based on Lemma 5 and the scaling property of covering numbers in normed spaces $\mathcal{N}(c \cdot \mathcal{K}, \varepsilon, \|\cdot\|) = \mathcal{N}(\mathcal{K}, \varepsilon/c, \|\cdot\|)$ for any scalar $c > 0$, we can get the functional metric entropy in the parameter space,

$$\mathcal{N}(\mathcal{F}_{\text{disc}}, \varepsilon, \|\cdot\|_\infty) \leq \mathcal{N}(L_{\text{map}} \cdot \Xi, \varepsilon, \|\cdot\|_2) = \mathcal{N}\left(\Xi, \frac{\varepsilon}{L_{\text{map}}}, \|\cdot\|_2\right). \quad \text{(S-33)}$$

Substituting $\delta'' = \varepsilon / L_{\text{map}}$, the metric entropy is strictly bounded by $D \log(1 + 2R_\Xi L_{\text{map}}/\varepsilon)$. Since parameters R_Ξ and L_{map} are constant scalings, the asymptotic functional complexity is $\mathcal{O}(VL \log(\sqrt{V}\varepsilon^{-1}))$.

(2) CDM. We first prove that the posterior mean belongs to the Hölder space. By Bayes' theorem, the exact posterior mean operator can be formulated as $\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t] = \boldsymbol{F}_t(\boldsymbol{x}_t)/p_t(\boldsymbol{x}_t)$, where the marginal density p_t and the numerator $\boldsymbol{F}_t$ are respectively given by

$$\begin{aligned} p_t(\boldsymbol{x}_t) &= \int_{\mathcal{M}} p_0^{\text{cont}}(\boldsymbol{x}_0)\varphi_{\sigma_t}(\boldsymbol{x}_t - \alpha_t\boldsymbol{x}_0)d\boldsymbol{x}_0, \\ \boldsymbol{F}_t(\boldsymbol{x}_t) &= \int_{\mathcal{M}} \boldsymbol{x}_0 p_0^{\text{cont}}(\boldsymbol{x}_0)\varphi_{\sigma_t}(\boldsymbol{x}_t - \alpha_t\boldsymbol{x}_0)d\boldsymbol{x}_0. \end{aligned} \tag{S-34}$$

Assume the data prior satisfies $p_0^{\text{cont}} \in \mathcal{C}^s_{M_1}(\mathcal{M})$. Applying a change of variables $\boldsymbol{y} = \alpha_t\boldsymbol{x}_0$. The marginal density can then be rewritten as a standard convolution

$$\begin{aligned} p_t(\boldsymbol{x}_t) &= \frac{1}{\alpha_t^{L_c}} \int_{\alpha_t\mathcal{M}} p_0^{\text{cont}}(\boldsymbol{y}/\alpha_t)\varphi_{\sigma_t}(\boldsymbol{x}_t - \boldsymbol{y})d\boldsymbol{y} \\ &= \frac{1}{\alpha_t^{L_c}} \left(\tilde{p}_0 * \varphi_{\sigma_t}\right)(\boldsymbol{x}_t), \end{aligned} \tag{S-35}$$

where $\tilde{p}_0(\boldsymbol{y}) := p_0^{\text{cont}}(\boldsymbol{y}/\alpha_t)$ is the rescaled prior density supported on $\alpha_t\mathcal{M}$ (with $\tilde{p}_0(\boldsymbol{y}) = 0$ for $\boldsymbol{y} \notin \alpha_t\mathcal{M}$), and $*$ denotes the convolution operator defined as $(f * g)(\boldsymbol{x}) := \int f(\boldsymbol{y})g(\boldsymbol{x} - \boldsymbol{y})d\boldsymbol{y}$. Based on the differentiation property of convolutions, $\nabla^k(f * g) = (\nabla^k f) * g$, we can apply the k-th order spatial derivative $\nabla^k_{\boldsymbol{x}_t}$ (for any $k \le s$) exclusively to $\tilde{p}_0$,

$$\nabla^k_{\boldsymbol{x}_t} p_t(\boldsymbol{x}_t) = \frac{1}{\alpha_t^{L_c}} \left(\nabla^k_{\boldsymbol{y}}\tilde{p}_0 * \varphi_{\sigma_t}\right)(\boldsymbol{x}_t), \tag{S-36}$$

where applying the chain rule to $\tilde{p}_0(\boldsymbol{y}) = p_0^{\text{cont}}(\boldsymbol{y}/\alpha_t)$ yields $\nabla^k_{\boldsymbol{y}}\tilde{p}_0(\boldsymbol{y}) = \frac{1}{\alpha_t^k}(\nabla^k p_0^{\text{cont}})(\boldsymbol{y}/\alpha_t)$. Substituting this functional evaluation back yields

$$\nabla^k_{\boldsymbol{x}_t} p_t(\boldsymbol{x}_t) = \frac{1}{\alpha_t^{L_c}} \int_{\alpha_t\mathcal{M}} \frac{1}{\alpha_t^k}(\nabla^k p_0^{\text{cont}})(\boldsymbol{y}/\alpha_t)\varphi_{\sigma_t}(\boldsymbol{x}_t - \boldsymbol{y})d\boldsymbol{y}. \tag{S-37}$$

Finally, reverting the change of variables via $\boldsymbol{y} = \alpha_t\boldsymbol{x}_0$ (which restores the differential $d\boldsymbol{y} = \alpha_t^{L_c}d\boldsymbol{x}_0$ and perfectly cancels the $1/\alpha_t^{L_c}$ Jacobian compensation factor), we obtain the exact derivative transfer:

$$\nabla^k_{\boldsymbol{x}_t} p_t(\boldsymbol{x}_t) = \frac{1}{\alpha_t^k} \int_{\mathcal{M}} (\nabla^k p_0^{\text{cont}})(\boldsymbol{x}_0)\varphi_{\sigma_t}(\boldsymbol{x}_t - \alpha_t\boldsymbol{x}_0)d\boldsymbol{x}_0. \tag{S-38}$$

Taking the ℓ_∞-norm over the high-probability ambient domain $\mathcal{K}_\delta$ and applying Young's convolution inequality with $\|\varphi_{\sigma_t}\|_{L^1} = 1$, we have

$$\|\nabla^k_{\boldsymbol{x}_t} p_t\|_{\ell_\infty(\mathcal{K}_\delta)} \le \frac{1}{\alpha_t^k}\|\nabla^k p_0^{\text{cont}}\|_{\ell_\infty(\mathcal{M})}\|\varphi_{\sigma_t}\|_{L^1} \le \frac{M_1}{\alpha_t^k}. \tag{S-39}$$

Thus, $p_t \in \mathcal{C}^s_{M_p}(\mathcal{K}_\delta)$, where $M_p = M_1/\alpha_t^s$. Similarly, for the numerator, let $\boldsymbol{g}(\boldsymbol{x}_0) = \boldsymbol{x}_0 p_0^{\text{cont}}(\boldsymbol{x}_0)$. Since $\mathcal{M}$ is compact, it follows that $\boldsymbol{g} \in \mathcal{C}^s_{M_2}(\mathcal{M})$ for some finite constant M_2. An identical convolution derivation yields $\|\boldsymbol{F}_t\|_{\mathcal{C}^s(\mathcal{K}_\delta)} \le M_2/\alpha_t^s$. Furthermore, the compactness of $\mathcal{M}$ and $\mathcal{K}_\delta$, combined with the global strict positivity of the Gaussian kernel ($\varphi_{\sigma_t} > 0$), ensures that $p_t(\boldsymbol{x}_t)$ is uniformly bounded away from zero

$$\inf_{\boldsymbol{x}_t \in \mathcal{K}_\delta} p_t(\boldsymbol{x}_t) \ge c_t > 0. \tag{S-40}$$

According to the quotient rule for Hölder spaces, the division of two $\mathcal{C}^s$-bounded functions with a strictly positive denominator preserves the $\mathcal{C}^s$-smoothness. Consequently, the exact posterior mean operator intrinsically satisfies

$$\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t] \in \mathcal{C}^s_M(\mathcal{K}_\delta \cap \mathcal{M}), \tag{S-41}$$

where the final Hölder norm bound $M < \infty$ is a constant entirely determined by M_1, M_2, α_t, and c_t.

Next, we analyze the metric entropy of the posterior mean function class over the high-probability manifold domain $\mathcal{K}_\delta \cap \mathcal{M}$. Let (U, ϕ) be a local chart mapping a compact patch $U \subset \mathcal{K}_\delta \cap \mathcal{M}$ diffeomorphically to a convex Euclidean domain $V \subset \mathbb{R}^{L_c}$. The inverse mapping $\phi^{-1} : V \to U$ is bi-Lipschitz, ensuring that the Hölder smoothness is preserved up to structural constants. Consequently, the covering number of the Hölder space on the manifold patch is asymptotically equivalent to that on the Euclidean domain:

$$\log\mathcal{N}(\mathcal{C}^s(U), \varepsilon, \|\cdot\|_\infty) \asymp \log\mathcal{N}(\mathcal{C}^s(V), \varepsilon, \|\cdot\|_\infty). \tag{S-42}$$

By transforming the problem to the Euclidean space V, we can directly evaluate the capacity of the vector-valued Hölder space. The posterior mean outputs vector fields bounded within a compact subset $B \subset \mathbb{R}^{L_c}$, which inherently satisfies the homogeneity condition. Invoking the metric entropy bounds for smooth vector-valued functions over Euclidean domains (Theorem 4 in [91] and Theorem 8.4(a) of [92]), the metric entropy scales linearly with the dimension L_c and inversely with the smoothness s. Absorbing the chart-induced geometric constants into the asymptotic notation, we obtain:

$$\log\mathcal{N}\left(\mathcal{C}^s(U), \varepsilon, \|\cdot\|_\infty\right) = \Omega\left(\varepsilon^{-L_c/s}\right). \tag{S-43}$$

Finally, the global metric entropy over the high-probability manifold domain $\mathcal{K}_\delta \cap \mathcal{M}$ is rigorously established through the topological relationship between the global manifold and its local charts. Since the local patch is a subset of the global domain ($U \subset \mathcal{K}_\delta \cap \mathcal{M}$), any ε-packing set of functions constructed on U intrinsically forms an ε-packing set on the global domain. Therefore, the global functional capacity is strictly bounded from below by the local complexity. Conversely, since the intersection $\mathcal{K}_\delta \cap \mathcal{M}$ is a compact set, it admits a finite subcover consisting of K local charts $\{U_i\}_{i=1}^K$. The global covering number is fundamentally bounded by the sum of the local covering numbers. Since K is a finite structural constant, the asymptotic rate remains invariant. The global metric entropy strictly follows the local scaling rate:

$$\log\mathcal{N}(\mathcal{C}^s_M(\mathcal{K}_\delta \cap \mathcal{M}), \varepsilon, \|\cdot\|_\infty) = \Omega\left(\varepsilon^{-L_c/s}\right). \tag{S-44}$$

■

C. Proof of Theorem 2

Proof: **(1) MIND.** For the sequence formulation, the concatenated parameter vector $\boldsymbol{\xi}_{\text{seq}}$ consists of the V shared latent anchors (each of dimension L) and the global $N \times V$ contextual routing logits. This extended parameter vector resides in a parameter space $\Xi_{\text{seq}} \subset \mathbb{R}^{D_{\text{seq}}}$ with total dimensionality $D_{\text{seq}} = VL + NV$.

Since the individual anchors are constrained to a hypersphere of radius R and the logits are norm-bounded by network regularization, the expanded space Ξ_{seq} is strictly contained within a compact Euclidean ball $B_2^{D_{\text{seq}}}(R_{\Xi_{\text{seq}}})$. Based on Cor. 4.2.13 of [88], the covering number is bounded by $\log \mathcal{N}(\Xi_{\text{seq}}, \delta'', \|\cdot\|_2) \le D_{\text{seq}} \log\left(1 + \frac{2R_{\Xi_{\text{seq}}}}{\delta''}\right)$.

We evaluate the parametric Jacobian matrix $\mathbf{J}_{\boldsymbol{\xi}_{\text{seq}}} \boldsymbol{F}^*_{\text{disc}}(\boldsymbol{x}_t)$. The sequence output is a concatenated vector $\boldsymbol{F}^*_{\text{disc}} = [\boldsymbol{f}_1^\top, \dots, \boldsymbol{f}_N^\top]^\top \in \mathbb{R}^{NL}$, where the n-th token output is $\boldsymbol{f}_n = \sum_{v=1}^V \pi_{n,v} \boldsymbol{a}_v$.

Step 1: Jacobian with respect to the sequence logits $\theta_{k,j}$. The network directly predicts the contextual logits $\theta_{n,v}$, the assignment probability for the n-th token is strictly defined by the token-wise softmax normalization:

$$\pi_{n,v}(\boldsymbol{x}_t; \boldsymbol{\xi}_{\text{seq}}) = \frac{\exp(\theta_{n,v})}{\sum_{i=1}^V \exp(\theta_{n,i})}. \tag{S-45}$$

Because the normalization is performed independently across the sequence dimension, the partial derivative with respect to any arbitrary logit $\theta_{k,j}$ evaluates piecewise:

$$\frac{\partial \pi_{n,v}}{\partial \theta_{k,j}} = \begin{cases} \pi_{n,v}(\delta_{v,j} - \pi_{n,j}) & \text{if } k = n, \\ 0 & \text{if } k \neq n. \end{cases} \tag{S-46}$$

Applying the product rule to the token output $\boldsymbol{f}_n = \sum_v \pi_{n,v} \boldsymbol{a}_v$, it fundamentally follows that for $k \neq n$, $\frac{\partial \boldsymbol{f}_n}{\partial \theta_{k,j}} = \mathbf{0}$. For $k = n$, the derivative evaluates analytically:

$$\frac{\partial \boldsymbol{f}_n}{\partial \theta_{n,j}} = \sum_{v=1}^V \boldsymbol{a}_v \pi_{n,v}(\delta_{v,j} - \pi_{n,j}) = \pi_{n,j}(\boldsymbol{a}_j - \boldsymbol{m}_n), \tag{S-47}$$

where $\boldsymbol{m}_n = \sum_v \pi_{n,v} \boldsymbol{a}_v$. Since $\|\boldsymbol{a}_j\|_2 \le R$ and the convex combination $\|\boldsymbol{m}_n\|_2 \le R$, the Euclidean norm of the column block vector $\frac{\partial \boldsymbol{F}^*_{\text{disc}}}{\partial \theta_{n,j}} \in \mathbb{R}^{NL}$ (which is zero everywhere except at the n-th block) is strictly bounded by:

$$\left\| \frac{\partial \boldsymbol{F}^*_{\text{disc}}}{\partial \theta_{n,j}} \right\|_2 = \left\| \frac{\partial \boldsymbol{f}_n}{\partial \theta_{n,j}} \right\|_2 \le \pi_{n,j}(\|\boldsymbol{a}_j\|_2 + \|\boldsymbol{m}_n\|_2) \le 2R := L_\theta. \tag{S-48}$$

Step 2: Jacobian with respect to the shared anchors $\boldsymbol{a}_j$. Because the sequence logits $\theta_{n,v}$ may be generated via projection architectures involving the anchors, we apply the product rule coupled with Assumption 2. The Jacobian block for the n-th token with respect to $\boldsymbol{a}_j$ is given by

$$\mathbf{J}_{\boldsymbol{a}_j} \boldsymbol{f}_n = \pi_{n,j} \mathbf{I}_L + \sum_{v=1}^V \boldsymbol{a}_v \left(\nabla_{\boldsymbol{a}_j} \pi_{n,v}\right)^\top. \tag{S-49}$$

By the triangle inequality and the bounded gradient assumption, the operator norm evaluates to:

$$\|\mathbf{J}_{\boldsymbol{a}_j} \boldsymbol{f}_n\|_{\text{op}} \le \pi_{n,j} + \sum_{v=1}^V \|\boldsymbol{a}_v\|_2 \|\nabla_{\boldsymbol{a}_j} \pi_{n,v}\|_2 \le 1 + R C_\pi := L_a. \tag{S-50}$$

For the entire sequence, the block column Jacobian is $\mathbf{J}_{\boldsymbol{a}_j} \boldsymbol{F}^*_{\text{disc}} = [\mathbf{J}_{\boldsymbol{a}_j} \boldsymbol{f}_1; \dots; \mathbf{J}_{\boldsymbol{a}_j} \boldsymbol{f}_N] \in \mathbb{R}^{NL \times L}$. For any vector $\boldsymbol{u}_{a_j} \in \mathbb{R}^L$, we have $\|\mathbf{J}_{\boldsymbol{a}_j} \boldsymbol{F}^*_{\text{disc}} \boldsymbol{u}_{a_j}\|_2^2 = \sum_{n=1}^N \|\mathbf{J}_{\boldsymbol{a}_j} \boldsymbol{f}_n \boldsymbol{u}_{a_j}\|_2^2 \le \sum_{n=1}^N L_a^2 \|\boldsymbol{u}_{a_j}\|_2^2 = N L_a^2 \|\boldsymbol{u}_{a_j}\|_2^2$. Hence, the spectral bound is mathematically amplified by the sequence length: $\|\mathbf{J}_{\boldsymbol{a}_j} \boldsymbol{F}^*_{\text{disc}}\|_{\text{op}} \le \sqrt{N} L_a$.

Step 3: Uniform Spectral Bound of the Sequence Mapping. The total sequence Jacobian is the aggregated block matrix $\mathbf{J}_{\boldsymbol{\xi}_{\text{seq}}} \boldsymbol{F}^*_{\text{disc}}$. For any conformably partitioned unit vector $\boldsymbol{u} = (\boldsymbol{u}_{a_1}, \dots, \boldsymbol{u}_{a_V}, u_{\theta_{1,1}}, \dots, u_{\theta_{N,V}})$, applying the triangle and Cauchy-Schwarz inequalities yields

$$\begin{aligned} &\|\mathbf{J}_{\boldsymbol{\xi}_{\text{seq}}} \boldsymbol{F}^*_{\text{disc}} \boldsymbol{u}\|_2 \\ &\le \sum_{j=1}^V \|\mathbf{J}_{\boldsymbol{a}_j} \boldsymbol{F}^*_{\text{disc}}\|_{\text{op}} \|\boldsymbol{u}_{a_j}\|_2 + \sum_{n=1}^N \sum_{j=1}^V \left\| \frac{\partial \boldsymbol{F}^*_{\text{disc}}}{\partial \theta_{n,j}} \right\|_2 |u_{\theta_{n,j}}| \\ &\le \left(\sum_{j=1}^V N L_a^2 + \sum_{n=1}^N \sum_{j=1}^V L_\theta^2 \right)^{1/2} \|\boldsymbol{u}\|_2. \end{aligned} \tag{S-51}$$

Taking the supremum over $\|\boldsymbol{u}\|_2 = 1$ establishes the uniform spectral bound

$$\begin{aligned} \sup_{\boldsymbol{\xi}_{\text{seq}}, \boldsymbol{x}_t} \|\mathbf{J}_{\boldsymbol{\xi}_{\text{seq}}} \boldsymbol{F}^*_{\text{disc}}(\boldsymbol{x}_t)\|_{\text{op}} &\le \sqrt{V N L_a^2 + N V L_\theta^2} \\ &= \sqrt{V N (L_a^2 + L_\theta^2)} := L^{\text{seq}}_{\text{map}} < \infty. \end{aligned} \tag{S-52}$$

Because the operator norm is uniformly bounded by $L^{\text{seq}}_{\text{map}}$, the Mean Value Theorem directly implies the global Lipschitz continuity of $\boldsymbol{\xi}_{\text{seq}} \mapsto \boldsymbol{F}^*_{\text{disc}}$. Consequently, the hypothesis class $\mathcal{F}^{\text{seq}}_{\text{disc}}$ is a Lipschitz image of Ξ_{seq}. Scaling the precision via $\delta'' = \varepsilon / L^{\text{seq}}_{\text{map}}$, the metric entropy is bounded by

$$\mathcal{N}(\mathcal{F}^{\text{seq}}_{\text{disc}}, \varepsilon, \|\cdot\|_\infty) \le \mathcal{N}\left(\Xi_{\text{seq}}, \frac{\varepsilon}{L^{\text{seq}}_{\text{map}}}, \|\cdot\|_2\right). \tag{S-53}$$

Substituting $D_{\text{seq}} = V(L + N)$, the asymptotic functional complexity evaluates to $\mathcal{O}(V(L+N) \log \varepsilon^{-1})$.

(2) CDM. In the continuous regime, evaluating the sequence jointly expands the functional domain to the Cartesian product $\mathcal{M}_{\text{seq}} = \mathcal{M}^N \subset \mathbb{R}^{N \times L_c}$.

As rigorously derived in the single-token proof via the differentiation property of convolutions, the exact posterior mean preserves the Hölder smoothness of the data prior. Because the sequence-level diffusion process operates in the joint space $\mathbb{R}^{N \times L_c}$, this structural preservation translates perfectly to the multidimensional domain through isomorphic analytical transfer. Thus, the sequence-level mapping intrinsically satisfies $\mathbb{E}[\boldsymbol{x}_0 \mid \boldsymbol{x}_t] \in \mathcal{C}^s_{M_{\text{seq}}}(\mathcal{K}^N_\delta \cap \mathcal{M}_{\text{seq}})$.

Invoking the standard minimax metric entropy bounds for $\mathcal{C}^s$-smooth vector-valued functions over Euclidean domains (e.g., Theorem 8.4 of [92]), the capacity fundamentally scales exponentially with the intrinsic dimension. Substituting the expanded domain dimension $d = NL_c$, the global functional capacity strictly inherits this exponential scaling:

$$\log \mathcal{N}(\mathcal{F}^{\text{seq}}_{\text{cont}}, \varepsilon, \|\cdot\|_\infty) = \Omega\left(\varepsilon^{-\frac{NL_c}{s}}\right). \tag{S-54}$$

This concludes the proof. ■

D. Auxiliary Derivation

1. Gradient of the Posterior Routing Probabilities. Let the routing probability be defined as $\pi_v = \frac{\exp(\vartheta_v)}{\sum_{k=1}^V \exp(\vartheta_k)}$. We calculate the partial derivative $\frac{\partial \pi_v}{\partial \vartheta_j}$ using the quotient rule. The derivation splits naturally into two cases depending on whether the target index j equals v:

Case 1: $v = j$**.** Applying the quotient rule yields

$$\begin{aligned}\frac{\partial \pi_v}{\partial \vartheta_v} &= \frac{\exp(\vartheta_v)\sum_{k=1}^{V}\exp(\vartheta_k) - \exp(\vartheta_v)\exp(\vartheta_v)}{\left(\sum_{k=1}^{V}\exp(\vartheta_k)\right)^2} \\ &= \frac{\exp(\vartheta_v)}{\sum_{k=1}^{V}\exp(\vartheta_k)}\left(1 - \frac{\exp(\vartheta_v)}{\sum_{k=1}^{V}\exp(\vartheta_k)}\right) \\ &= \pi_v(1-\pi_v). \end{aligned} \tag{S-55}$$

Case 2: $v \neq j$**.** The numerator does not contain ϑ_j and is treated as a constant.

$$\begin{aligned}\frac{\partial \pi_v}{\partial \vartheta_j} &= \frac{0 - \exp(\vartheta_v)\exp(\vartheta_j)}{\left(\sum_{k=1}^{V}\exp(\vartheta_k)\right)^2} \\ &= -\left(\frac{\exp(\vartheta_v)}{\sum_{k=1}^{V}\exp(\vartheta_k)}\right)\left(\frac{\exp(\vartheta_j)}{\sum_{k=1}^{V}\exp(\vartheta_k)}\right) \\ &= -\pi_v\pi_j. \end{aligned} \tag{S-56}$$

Both cases can be unified into a single expression

$$\frac{\partial \pi_v}{\partial \vartheta_j} = \pi_v(\delta_{vj} - \pi_j). \tag{S-57}$$

2. Jacobian Matrix with Respect to Anchor Vectors.

Recall that the predicted continuous state $f^*_{\text{dist}}(\boldsymbol{x}_t)$ under the discrete diffusion formulation is given by the posterior expectation over all anchor vectors:

$$f^*_{\text{dist}}(\boldsymbol{x}_t) = \sum_{v=1}^{V} \pi_v \boldsymbol{a}_v, \tag{S-58}$$

where $\boldsymbol{a}_v \in \mathbb{R}^L$ is the v-th anchor vector and π_v is its corresponding posterior routing probability. Applying the product rule for vector calculus to the summation yields:

$$\mathbf{J}_{\boldsymbol{a}_j} f^*_{\text{dist}}(\boldsymbol{x}_t) = \sum_{v=1}^{V} \frac{\partial(\pi_v \boldsymbol{a}_v)}{\partial \boldsymbol{a}_j} = \sum_{v=1}^{V}\left[\pi_v \frac{\partial \boldsymbol{a}_v}{\partial \boldsymbol{a}_j} + \boldsymbol{a}_v\left(\nabla_{\boldsymbol{a}_j}\pi_v\right)^\top\right]. \tag{S-59}$$

Suppose the anchor vectors $\boldsymbol{a}_v$ are independent parameters, we have $\frac{\partial \boldsymbol{a}_v}{\partial \boldsymbol{a}_j} = \delta_{vj}\mathbf{I}_L$. The exact formulation of the Jacobian matrix is

$$\mathbf{J}_{\boldsymbol{a}_j} f^*_{\text{dist}}(\boldsymbol{x}_t) = \pi_j \mathbf{I}_L + \sum_{v=1}^{V} \boldsymbol{a}_v\left(\nabla_{\boldsymbol{a}_j}\pi_v\right)^\top. \tag{S-60}$$