Hi-DiT: Hybrid Latent-Pixel Diffusion Transformer for Image Generation

Yedong Shen[1⋆], Yehao Li[2], Yingwei Pan[2], Yanyong Zhang[1], and Ting Yao[2]

[1] University of Science and Technology of China
[2] HiDream.ai Inc
sydong2002@mail.ustc.edu.cn, {liyehao,pandy}@hidream.ai,
yanyongz@ustc.edu.cn, tiyao@hidream.ai

Abstract. Current diffusion models face a fundamental tension between the computational efficiency of latent-space generation and the fine-detail fidelity of pixel-space generation. Latent Diffusion Models (LDMs) rely on VAE compression, which can discard high-frequency visual cues and limit pixel-level quality, while pixel-space diffusion often suffers from difficult optimization when global structure and high-frequency details must be learned jointly. In this paper, we propose the Hybrid Latent-Pixel Diffusion Transformer (Hi-DiT), a unified Diffusion Transformer that bridges these two regimes within a single architecture. Motivated by the temporal heterogeneity of denoising—early timesteps primarily establish coarse global structure, whereas late timesteps increasingly emphasize high-frequency detail—we introduce a dual-stream design with temporal specialization. Hi-DiT performs semantic planning in a compact latent pathway at early stages, and activates a pixel pathway at later stages to synthesize high-frequency details using time-consistent noisy pixel embeddings. A Time-Gated Injection mechanism schedules the participation of pixel tokens only in the low-noise regime, and a lightweight sub-pixel prediction head enables efficient dense detail generation. Extensive experiments demonstrate that Hi-DiT achieves state-of-the-art performance, obtaining an FID of 1.06 on ImageNet 256×256, 1.26 on ImageNet 512×512, and 5.15 on MS-COCO text-to-image generation. Code is available at: *https://github.com/HiDream-ai/Hi-DiT.*

Keywords: Image generation · Diffusion model · Time-Gated denoising

1 Introduction

Denoising diffusion models have established a dominant paradigm for high-fidelity image generation [14]. To scale these architectures to high resolutions, the community has predominantly converged on Latent Diffusion Models (LDMs) [32, 36]. By performing the generative process within a compressed manifold induced by a pre-trained Variational Autoencoder (VAE) [17], LDMs significantly reduce computational overhead and often improve training stability. However, this semantic efficiency comes with a practical trade-off: optimizing in a compressed

⋆ This work was performed at HiDream.ai.

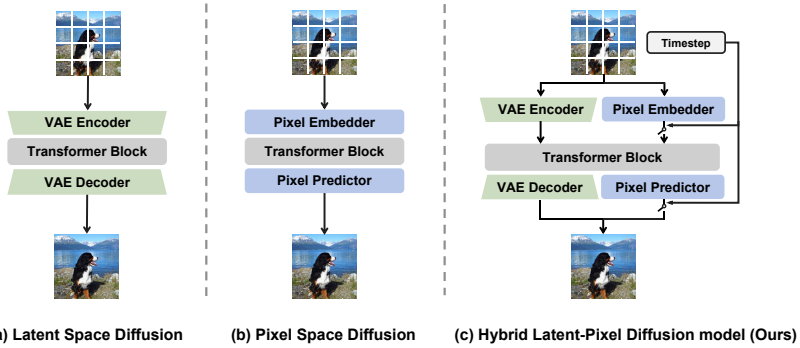

Fig. 1: Architectural comparison of different diffusion paradigms. (a): Latent Space Diffusion operates within a compressed manifold to model low-frequency semantics. (b): Pixel Space Diffusion acts directly on raw visual tokens to jointly generate global structures and local details. (c): Hi-DiT introduces a dual-stream framework: the latent stream establishes low-frequency global semantics, while a time-gated pixel stream dynamically injects high-frequency details guided by the denoising timestep.

representation can limit pixel-level fidelity. In practice, spatial downsampling and limited latent capacity make fine-grained textures and sharp boundaries harder to preserve and reconstruct consistently after decoding [11,39]. As a result, standard latent models often struggle to fully reproduce crisp high-frequency details even when global semantics are correct.

To avoid the latent compression bottleneck, a natural alternative is to model the diffusion trajectory directly in raw pixel space [7,16,52]. While this formulation does not rely on VAE tokenization, directly modeling the high-entropy pixel distribution introduces formidable optimization challenges. The model must simultaneously capture global low-frequency structure and complex high-frequency variations within a very high-dimensional space, which can lead to capacity competition and inefficient learning dynamics [7,21,55]. As a result, pure pixel-space models typically exhibit slower convergence and harder optimization under comparable compute when compared to their latent counterparts.

In light of the optimization difficulty of pixel-space diffusion, especially its need to learn global structure and fine details simultaneously, we revisit the denoising trajectory from a temporal perspective. Prior analyses suggest an empirically observed spectral-temporal tendency in diffusion denoising [1, 27, 57]: early, high-noise stages are dominated by coarse low-frequency structure, while later, low-noise stages increasingly require high-frequency detail synthesis. This temporal heterogeneity implies that a temporally uniform denoiser must handle two qualitatively different objectives with a single parameter set, which can lead to objective interference and inefficient capacity allocation [46, 51, 59]. These observations motivate an architecture that uses a compact semantic pathway to establish global structure early, and reserves pixel-level modeling capacity primarily for late-stage high-frequency prediction.

Guided by this insight, we introduce the Hybrid Latent-Pixel Diffusion Transformer (Hi-DiT), a unified dual-stream framework driven by temporal specialization. Hi-DiT decomposes denoising into two complementary streams within a parameter-shared transformer backbone. The Latent Stream operates on VAE latents and focuses on establishing global structure and semantic layout, which dominate the high-noise regime. In parallel, the Pixel Stream maintains access to raw pixels and specializes in fine-detail refinement, which becomes most relevant in the low-noise regime. By coordinating both streams inside a single backbone, Hi-DiT enables efficient feature interaction while keeping each stream's role time-specialized across the entire denoising trajectory.

To align stream participation with diffusion time, we propose a Time-Gated Injection mechanism that suppresses pixel-stream features during early semantic planning and activates them in late-stage denoising, which is conceptually related to curriculum learning [3] and promotes more stable and specialized optimization across different stages. In addition, to efficiently translate transformer representations into fine-grained details, we introduce a lightweight High-Frequency Pixel Predictor based on hierarchical sub-pixel decoding [40]. Together, these designs mitigate capacity competition and enable high-frequency synthesis without activating the pixel pathway throughout the entire denoising trajectory.

In summary, our contributions are:

- We propose Hi-DiT, a hybrid latent–pixel Diffusion Transformer that unifies the semantic efficiency of latent space diffusion and the visual precision of pixel space diffusion via temporal specialization in a parameter-shared backbone, while retaining direct access to pixel-level information.
- We introduce a Time-Gated Injection strategy that activates the pixel branch only in the low-noise regime when high-frequency refinement becomes dominant, together with a lightweight high-frequency pixel predictor based on hierarchical sub-pixel decoding for efficient detail synthesis.
- Extensive experiments on ImageNet and MS-COCO demonstrate that Hi-DiT improves fine-detail fidelity while achieving state-of-the-art generation quality, with consistent gains across resolutions.

2 Related works

Diffusion models [14, 41] have become a foundational paradigm for generative modeling. They formulate image generation as an iterative denoising process that progressively transforms noise into structured data through multi-step inference.

2.1 Latent Diffusion Model

To reduce the computational burden and optimization difficulty of pixel-space denoising, most modern systems adopt Latent Diffusion Models (LDMs) [32,36]. LDMs compress images into a low-dimensional latent space using a VAE [17,44] and perform denoising in that space. Since VAE training is typically decoupled from diffusion modeling, reconstruction errors from a suboptimal VAE can

upper-bound the final generation quality. Early LDMs [32,36,54] mainly relied on U-Nets [37], while DiT [31] improved scalability by replacing U-Nets with Transformer backbones. SiT [25] further validated Transformer-based backbones under a linear flow matching formulation. To strengthen semantic learning in the latent space, recent works align diffusion representations with vision foundation models. REPA [51] aligns intermediate diffusion features to DINOv2. REPA-E [19] explores end-to-end joint optimization of the VAE and DiT, but the evolving latent space can make denoising targets unstable. VA-VAE [49] and RAE [58] instead construct highly semantic latents guided by frozen foundation models to alleviate the reconstruction–generation conflict and accelerate convergence. Despite these advances, VAE compression inevitably discards pixel information, which limits latent diffusion in synthesizing fine-grained high-frequency details.

2.2 Pixel Diffusion Model

Early diffusion models [5, 9, 14] directly denoised in pixel space using U-Net backbones with full-resolution feature maps and long-range skip connections, but their dense computation is expensive. With the success of Transformers [30, 45,47], pixel-level DiTs [4] have emerged; however, tokenizing individual pixels is computationally prohibitive, so these methods typically patchify images. A key challenge is recovering dense pixel-level details from compressed patch tokens. To mitigate this, SiD2 [16] and PixelFlow [6] adopt hierarchical designs with smaller patch sizes, at the cost of heavier computation and reduced architectural simplicity. Other methods, including PixelDiT [52], Deco [26], and DiP [7], introduce specialized pixel prediction modules to improve decoding, which increases parameters and overhead. JiT [21] highlights the importance of prediction targets by predicting x_0 to better reconstruct high-dimensional signals from patch tokens. Nevertheless, pixel-space diffusion still learn global low-frequency semantics and local high-frequency details simultaneously, often resulting in slower convergence and weaker visual quality than latent-based models.

3 Method

3.1 Motivation

We begin by revisiting the standard formulation of generative modeling. Let $x \in \mathbb{R}^{H\times W\times 3}$ denote an image drawn from the data distribution $q(x)$. Diffusion-based generative models [14,25,31] construct a time-dependent probability path $p_t(x)$ that gradually transforms a simple noise distribution $\epsilon \sim \mathcal{N}(0, I)$ into the target data distribution $x_0 \sim q(x)$ over the time horizon $t \in [0, 1]$. Taking the flow matching framework as a representative example, the generative dynamics can be formulated as an Ordinary Differential Equation (ODE):

$$dx_t = v_\theta(x_t, t, c)dt, \tag{1}$$

where c denotes the conditioning input (e.g., a class label or text embedding), and v_θ is a learnable velocity field trained with the flow matching objective:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{t,c,x_0,\epsilon}\left[\|v_\theta(x_t,t,c) - (\epsilon - x_0)\|^2\right], \quad x_t = (1-t)x_0 + t\epsilon. \tag{2}$$

To improve computational efficiency, modern large-scale generative models typically perform this process in a latent representation space rather than directly in pixel space. Latent Diffusion Models [17, 36, 56] first encode images into a compact latent representation $z = \mathcal{E}(x)$ using a pre-trained VAE. The generative process is then defined in the latent space, where a neural network learns a time-dependent velocity field $v_\theta^z(z_t, t, c)$ along the diffusion trajectory z_t that gradually transforms Gaussian noise into the latent data distribution. The generated latent $\hat{z}$ is finally decoded back to the image space via $\hat{x} = \mathcal{D}(\hat{z})$.

Due to spatial downsampling and channel compression in the VAE, the reconstruction $\hat{x}$ cannot perfectly preserve all information from the original image x. Empirically, the decoder tends to faithfully recover coarse spatial structures while losing part of the fine-grained details. Motivated by this observation, we adopt a frequency-based perspective and decompose an image into low- and high-frequency components:

$$x = x^{\text{LF}} + x^{\text{HF}}, \qquad x^{\text{LF}} \approx \hat{x}, \tag{3}$$

where x^{LF} denotes the low-frequency component that captures slowly varying spatial structures of the image, while x^{HF} represents the high-frequency component containing rapidly varying signals such as edges, textures, and fine details.

Importantly, diffusion denoising exhibits temporal heterogeneity: early (high-noise) steps mainly recover coarse structure, while later (low-noise) steps increasingly synthesize fine-scale details [1, 27, 34, 42]. From a frequency perspective, different stages emphasize different bands, so a single temporally uniform denoiser can be suboptimal. Motivated by this, we specialize denoising across time using two complementary streams that share a Transformer backbone: a Latent Stream operating on z_t to predict the velocity field $v_\theta^z(z_t, t, c)$ and capture low-frequency structure, and a Pixel Stream operating in pixel space (without VAE compression) that is activated only in late stages to refine high-frequency details.

3.2 Overview of Hi-DiT

Motivated by these observations, we propose Hi-DiT (Hybrid Latent-Pixel Diffusion Transformer), a unified framework that integrates latent and pixel representations within a single Diffusion Transformer. Unlike prior approaches that operate strictly in either compressed latents or raw pixels, Hi-DiT employs a shared Transformer backbone to orchestrate denoising through two streams with time-specialized functionality. As illustrated in Figure 2, Hi-DiT features a temporally adaptive architecture that transitions between two distinct denoising regimes. In the early stage (high-noise), the model exclusively employs a Latent Stream to consolidate coarse spatial layout and global semantics within the compressed latent space. As the diffusion process progresses toward the low-noise regime,

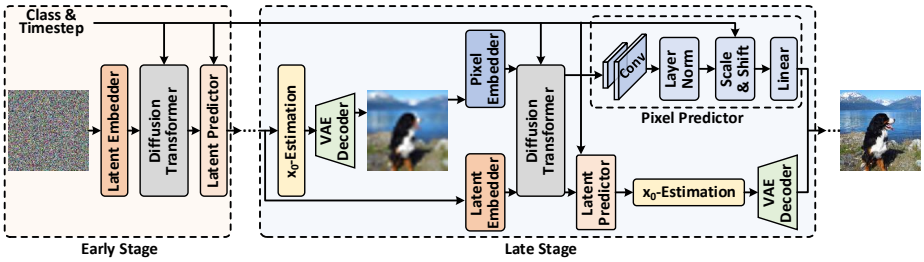

Fig. 2: Overview of the Hi-DiT framework. Hi-DiT employs a temporally specialized two-stage architecture that seamlessly transitions from structural formation to detail refinement. In the early stage (high-noise regime), the model exclusively employs the Latent Stream in compressed latent space to model low-frequency structures and global semantics. As the diffusion process progresses toward the late stage (low-noise regime), a Pixel Stream is activated to operate directly in pixel space. By processing the decoded image scaffold alongside latent tokens, the Transformer performs dual prediction: the latent predictor maintains structural consistency, while the pixel predictor recovers sharp, high-frequency textures. The final high-resolution output is synthesized by fusing the VAE-decoded structure with these pixel-level refinements.

fine-scale details become increasingly important. To address this, we introduce a Pixel Stream that operates directly in pixel space and is activated only during the late denoising stages to refine high-frequency details. Crucially, both streams share the same Transformer parameters for efficient feature interaction, while using stream-specific input projections and output heads to match their respective objectives: the latent predictor anchors structural consistency, whereas the pixel predictor recovers sharp details. The final high-resolution synthesis is achieved by fusing the VAE-decoded scaffold with these pixel-level refinements.

3.3 Latent Stream

The Latent Stream is dedicated to modeling low-frequency content and global semantics across the denoising trajectory. Given an image x, we first obtain a compressed representation $z = \mathcal{E}(x)$ via a pre-trained VAE encoder. Operating within the latent manifold, the Diffusion Transformer learns the semantic denoising dynamics necessary to establish a stable structural scaffold. Specifically, the latent tokens z are processed by a Latent Embedder to align their dimensionality with the Transformer's hidden space, yielding the initial states h_z^0. These states are propagated through the Transformer blocks to produce the final hidden states h, which are subsequently mapped by a dedicated Latent Predictor to the target velocity space. To supervise this denoising process and ensure structural integrity, we optimize the model at each timestep t using the rectified flow matching objective, which encourages the predicted velocity v_θ^z to align with the ground-truth probability path:

$$\mathcal{L}_{\text{lat}} = \mathbb{E}_{t,c,z_0,\epsilon}[\|v_\theta^z(z_t, t) - (\epsilon - z_0)\|^2], \tag{4}$$

where $z_t = (1-t)z_0 + t\epsilon$ represents the interpolated latent state.

Since the VAE decoder $\mathcal{D}(\cdot)$ is inherently lossy and fails to recover sharp high-frequency textures, the decoded $\hat{x}_{\text{base}} = \mathcal{D}(\hat{z})$ serves only as a coarse structural anchor. To compensate for these reconstruction artifacts and restore fine-grained details, we introduce a Pixel Stream that is activated during the low-noise regime. By operating directly in pixel space, this stream refines the blurred VAE output, effectively decoupling global structural planning from high-frequency texture recovery within a unified model.

3.4 Pixel Stream

To compensate for the information loss during latent compression, the Pixel Stream is introduced to provide explicit pixel-level cues essential for high-frequency synthesis in the late denoising regime. At each low-noise timestep t, we construct a noisy pixel tokens x_t that corresponds to the noise level of the latent tokens z_t. These pixel-level inputs are mapped into a sequence of pixel tokens via a Pixel Embedder. Subsequently, the Diffusion Transformer concurrently processes both pixel and latent tokens, capturing the intricate dependencies between global structures and local textures. The resulting hidden states h are then fed into a dedicated Pixel Predictor to recover sharp, high-fidelity details.

Pixel Embedder. We employ a bottleneck patch embedding [21] to inject pixel-level information into the Diffusion Transformer. Specifically, the noisy pixel tokens x_t is partitioned into non-overlapping patches with size $p \times p$, which are subsequently mapped to align with the Transformer's hidden dimension via the bottleneck structure. These resulting tokens h_x^0, along with the latent sequence h_z^0 are then fed into the Diffusion Transformer for joint processing. Crucially, by setting patch size p to match the VAE's downsampling factor, we ensure that both pixel and latent tokens maintain the same sequence length, facilitating a spatial correspondence that simplifies the learning of high-frequency details.

Pixel Predictor. Previous pixel-space diffusion baselines [21] typically rely on a naive linear projection to reconstruct large $p \times p$ image patches directly from individual tokens. However, such direct high-resolution regression is often intractable. Inspired by the hierarchical decoding architecture of VAEs, we propose a progressive upsampling strategy to mitigate this bottleneck. Instead of a single-step projection, our Pixel Predictor employs a sequence of convolutional refinement blocks to gradually expand the spatial resolution, culminating in a final linear layer that predicts localized sub-patches. Specifically, given the Transformer hidden states h, we project and reshape them into an initial 2D feature grid $\mathbf{G}_0 \in \mathbb{R}^{C_{\text{mid}} \times \frac{H}{p} \times \frac{W}{p}}$. Each refinement block $\mathcal{B}_i$ ($i \in \{1, 2\}$) then performs a convolution and a PixelShuffle operation with a scale factor of 2, followed by GroupNorm and SiLU activation:

$$\mathbf{G}_i = \text{SiLU}(\text{GroupNorm}(\mathcal{PS}(\text{Conv}(\mathbf{G}_{i-1})))), \tag{5}$$

where Conv and $\mathcal{PS}$ denote the convolution and PixelShuffle operations, respectively, and $\mathbf{G}_{i-1}$ represents the input to the i-th refinement block. After two

consecutive stages of upsampling, the spatial resolution is expanded by a factor of 4, yielding a dense feature grid $\mathbf{G}_2 \in \mathbb{R}^{C_{\text{mid}} \times \frac{H}{p/4} \times \frac{W}{p/4}}$. By virtue of this prior spatial expansion, the final linear projection head is only required to predict a localized sub-patch of size $\frac{p}{4} \times \frac{p}{4}$ at each spatial location. This strategy significantly alleviates the regression burden and introduces a stronger inductive bias for synthesizing fine-grained, spatially local textures.

Following the conditioning mechanism of the Latent Predictor [31], we incorporate Adaptive Layer Normalization (AdaLN) to modulate the upsampled features. This ensures that the high-frequency synthesis is conditioned on both the denoising step and the global semantic context. Specifically, a global conditioning vector $\hat{c}$ is derived from the summation of the timestep and condition embeddings. This vector is processed by an MLP to yield instance-specific scale and shift parameters (γ, β), which are subsequently used to modulate the upsampled feature maps $\mathbf{G}_2$. Finally, the modulated features are linearly projected to obtain the high-frequency patch vectors. The operation is computed as:

$$\begin{aligned} [\gamma, \beta] &= \text{MLP}(\hat{c}) \\ x^{\text{HF}} &= \text{Linear}(\text{LayerNorm}(\mathbf{G}_2) \odot (1 + \gamma) + \beta), \end{aligned} \tag{6}$$

where $\odot$ denotes element-wise multiplication. By decomposing the high-dimensional mapping into progressive upsampling and localized projections, this architecture significantly streamlines the pixel prediction process, enabling the model to recover sharp, high-frequency details with greater efficiency and accuracy.

3.5 Time-Gated Injection Mechanism

Diffusion models typically follow a coarse-to-fine trajectory: early stages determine global layout, whereas later stages focus on local textures [1, 27]. To effectively unify latent and pixel tokens, we propose a Time-Gated Injection mechanism. This mechanism explicitly controls the timing of pixel token integration into the shared transformer backbone, reflecting the varying structural needs across the denoising timeline. Let h_z^0 and h_x^0 denote the latent and pixel embedding sequences produced by the Latent Embedder and Pixel Embedder, respectively. The input to the Diffusion Transformer is then constructed as:

$$h_{zx}^0 = h_z^0 + \mathcal{G}(t) \cdot h_x^0, \tag{7}$$

where $\mathcal{G}(t)$ controls the contribution of pixel tokens. We adopt a hard schedule $\mathcal{G}(t) = \mathbb{1}(t < \tau)$ with a fixed threshold τ. The fused tokens h_{zx}^0 are then processed by the shared transformer backbone to produce hidden features.

We also employ the same hard scheduling strategy to aggregate the outputs from the Latent and Pixel Predictors. For the Latent Stream, we first estimate the clean latent $\hat{z}_0$ based on the velocity field $v_\theta^z(z_t, t, c)$ predicted by the Latent Predictor. Subsequently, this latent estimate is decoded and combined with the high-frequency output x^{HF} from the Pixel Predictor to produce the final result: $x_{\text{final}} = \mathcal{D}(\hat{z}_0) + \mathbb{1}(t < \tau) \cdot x^{\text{HF}}$. This design ensures that pixel-level refinements are only integrated during the later stages of the denoising process, consistent with the Time-Gated Injection mechanism used in the input stage.

3.6 Optimization

Our framework is jointly optimized to synchronize latent semantic priors with pixel-space high-frequency synthesis. For the Latent Stream, we employ a standard Flow Matching objective, training the backbone to regress the conditional velocity field $v_\theta^z(z_t, t, c)$. This yields the latent loss $\mathcal{L}_{\text{lat}}$, which ensures the robust construction of the global layout.To explicitly supervise the high-frequency rendering of the Pixel Stream, we formulate a composite pixel-level loss. The pixel loss $\mathcal{L}_{\text{pix}}$ is then computed using reconstruction and perceptual constraints:

$$\mathcal{L}_{\text{pix}} = \lambda_{\text{rec}} \|x_{\text{final}} - x_{\text{GT}}\|^2 + \lambda_{\text{per}} \text{LPIPS}(x_{\text{final}}, x_{\text{GT}}), \tag{8}$$

where λ_{rec} and λ_{per} are balancing hyperparameters. During high-noise timesteps ($t \geq \tau$), we optimize only the Latent Stream via $\mathcal{L}_{\text{lat}}$, effectively disabling the Pixel Stream. In contrast, during low-noise timesteps ($t < \tau$), the entire network is optimized using both $\mathcal{L}_{\text{lat}}$ and $\mathcal{L}_{\text{pix}}$. Consequently, the unified training objective is: $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{lat}} + \mathbb{1}(t < \tau) \cdot \mathcal{L}_{\text{pix}}$ This staged optimization strictly decouples the generative burden, ensuring high-fidelity detail synthesis without compromising early semantic convergence.

3.7 Inference

At inference time, we apply the same time-gated strategy as in training: sampling starts in latent space to form global semantics, and pixel refinement is activated only in the low-noise regime. Let t_0 be the first sampled timestep after crossing the threshold, i.e., the first step with $t < \tau$. When the gate opens, we initialize the pixel branch by decoding the current latent state, $\mathbf{x}_{t_0}^{\text{dec}} = \mathcal{D}(\mathbf{z}_{t_0})$, where $\mathcal{D}(\cdot)$ denotes the VAE decoder, and use it as the initial pixel input.

Note that $\mathbf{x}_{t_0}^{\text{dec}}$ can still contain artifacts since $\mathbf{z}_{t_0}$ remains partially noisy. This is consistent with training, where the pixel branch is also exposed to pixel inputs at corresponding noisy timesteps. Therefore, the activation conditions of the pixel branch are aligned between training and inference. For timesteps $t < \tau$, latent and pixel representations are jointly processed by the shared Transformer backbone. The pixel head predicts a high-frequency component x^{HF}, and we update the pixel state by combining it with the decoded latent at the current step: $\mathbf{x}_{t-1}^{input} = \mathbf{x}_t^{\text{dec}} + x^{\text{HF}}$. The updated $\mathbf{x}_{t-1}^{input}$ is fed back as the pixel input for the next timestep, enabling synchronous refinement along the latent trajectory while remaining grounded on the semantic scaffold provided by latent space.

4 Experiment

4.1 Dataset

To demonstrate the efficacy of our hybrid latent-pixel approach, we evaluate Hi-DiT against state-of-the-art (SOTA) methods on two standard datasets: ImageNet [8] and MS-COCO [23]. For class-conditional image generation, we conduct experiments on the ImageNet dataset at both 256×256 and 512×512

resolutions. This dataset consists of 1,281,167 training images distributed across 1,000 distinct classes. For text-to-image generation, we evaluate our model on the MS-COCO dataset at a resolution of 256×256. The dataset comprises 82,783 training images and 40,504 validation images, with each image paired with five descriptive captions. Following the standard protocol established by Stable Diffusion [36], we employ a pre-trained CLIP [33] text encoder to project the image captions into sequences of text embeddings.

4.2 Experimental Settings

Network Architectures. Following the Hi-DiT framework outlined in Section 3, our architecture comprises three primary components: a pre-trained VAE for latent encoding and decoding, a parameter-shared Diffusion Transformer backbone for the denoising process, and specialized pixel-level modules. To process the latent representations, we adopt the highly effective encoder and decoder from VAVAE [49]. Specifically, the encoder utilizes a spatial downsampling factor of 16 and a latent channel dimension of 32. For the parameter-shared denoising backbone, we build upon the SiT [25] architecture. The main network is constructed as a 28-layer SiT transformer block structure with a patch size of 1. We maintain this identical backbone and VAE configuration for both the 256×256 and the higher-resolution 512×512 ImageNet experiments. For the text-to-image generation task on the MS-COCO dataset, we adapt the backbone to follow the U-ViT [2] configuration. Specifically, we utilize the U-ViT-Small variant, which is composed of 16 transformer blocks.

Training Setup. All experiments are conducted on a cluster of 8 NVIDIA A100 (80GB) GPUs. For the ImageNet experiments, we strictly utilize the official training split. The image data for both 256×256 and 512×512 resolutions is preprocessed following the identical protocol established in ADM [9], where original images are center-cropped and resized to the target resolution. During training, we employ the pretrained VAE and train the latent backbone (configured as SiT-XL/1) together with the full model for a total of 800 epochs. For optimization, we utilize the AdamW optimizer [24] with a learning rate of 1e-4 and a global batch size of 256. To ensure training stability, we apply gradient clipping and maintain an Exponential Moving Average (EMA) of the model weights. Furthermore, to enhance semantic structural learning, we incorporate an alignment loss using DINOv2 [29] as the external visual feature extractor. Following the pre-training of the latent backbone, we transition to the joint fine-tuning phase. We train the entire unified model using the same learning rate. For the high-resolution 512×512 experiments, we initialize the backbone with the pre-trained 256×256 weights and fine-tune the model for 50k steps while keeping the training configurations consistent. For the text-to-image generation task on the MS-COCO dataset, we align our training hyperparameters with the U-ViT setup. Specifically, the model is optimized using AdamW with a learning rate of 2e-4, a weight decay of 0.03, and a 5,000-step linear warmup.

Evaluation Metrics. For the evaluation on the ImageNet dataset, we employ Fréchet Inception Distance (FID) [13], Inception Score (IS) [38], and Preci-

Table 1: Quantitative comparison of generation@256 on ImageNet. "↓" or "↑" indicate lower or higher values are better. Metrics include Fréchet Inception Distance (FID), Inception Score (IS), precision and recall.

Model	Class	Params	Generation@256 w/o CFG				Generation@256 w/ CFG			
			FID ↓	IS ↑	Precision ↑	Recall ↑	FID ↓	IS ↑	Precision ↑	Recall ↑
VAR [43]	AR	600M	-	-	-	-	2.57	302.6	0.83	0.56
MAR [22]	AR	943M	-	-	-	-	1.55	303.7	0.81	0.62
RAR [50]	AR	955M	-	-	-	-	1.50	306.9	0.80	0.62
DiT-XL [31]	Latent Diffusion	675M	9.62	121.5	0.67	0.67	2.27	278.2	0.83	0.57
SiT-XL [25]	Latent Diffusion	675M	8.61	131.7	0.68	0.67	2.06	270.3	0.82	0.59
MaskDiT [59]	Latent Diffusion	675M	5.69	177.9	0.74	0.60	2.28	276.6	0.80	0.61
VA-VAE [49]	Latent Diffusion	675M	2.17	205.6	0.77	0.65	1.35	295.3	0.79	0.65
REPA [51]	Latent Diffusion	675M	5.78	158.3	0.70	0.68	1.29	306.3	0.79	0.65
PixelFlow [6]	Pixel Diffusion	677M	-	-	-	-	1.98	282.1	0.81	0.60
PixelDiT [52]	Pixel Diffusion	797M	-	-	-	-	1.61	292.7	0.78	0.64
DiP [7]	Pixel Diffusion	631M	-	-	-	-	1.79	281.9	0.80	0.63
Deco [26]	Pixel Diffusion	682M	-	-	-	-	1.62	301.0	0.80	0.62
JiT-G [21]	Pixel Diffusion	2B	-	-	-	-	1.82	292.6	0.79	0.62
Hi-DiT(CFG=2.4)	Latent+Pixel	689M	**1.66**	**213.6**	0.77	0.65	**1.06**	296.6	0.79	0.66
Hi-DiT(CFG=3.0)		689M					1.10	**319.4**	0.79	0.65

sion/Recall [18] to quantitatively assess the visual quality of the synthesized images at both 256×256 and 512×512 resolutions. Following standard evaluation protocols, these metrics are computed on 50,000 randomly generated samples with an equal number of samples per class. For the text-to-image generation task, we randomly sample 30,000 text prompts from the validation set to guide the generation process. We report the FID score as the main metric.

4.3 Result on Class-Conditional Image Generation

Table 1 and Table 3 present the quantitative evaluation of class-conditional image generation on ImageNet 256×256 and 512×512, respectively. We compare Hi-DiT against state-of-the-art autoregressive (AR), pure pixel-space, and latent-space diffusion paradigms under both standard and Classifier-Free Guidance (CFG) [15] settings.

Overall, Hi-DiT establishes a new state-of-the-art across all evaluated resolutions and metrics. On ImageNet 256×256, Hi-DiT achieves an exceptional FID of 1.06 with CFG. Crucially, it accomplishes this unparalleled visual fidelity with only 689M parameters, significantly outperforming models with massively scaled capacities, such as the 2B-parameter JiT-G (1.82). Furthermore, our framework exhibits remarkable base generation capability: even without CFG, Hi-DiT achieves a highly competitive FID of 1.66, drastically bypassing standard LDMs like DiT (9.62) and representation-aligned methods like VA-VAE (2.17).

This performance advantage seamlessly extends to higher resolutions. As shown in Table 3, Hi-DiT yields a gFID of 1.26 on ImageNet 512×512, establishing a substantial margin over both the best-performing pixel models and latent models. These results emphatically validate that our dual-stream architecture successfully bridges the semantic efficiency of latent models and the lossless

Table 2: FID results of different models on MS-COCO 256 × 256 validation.

Type	Model	FID ↓
GAN	AttnGAN [48]	35.49
	DM-GAN [20]	32.64
	XMC-GAN [53]	9.33
AR	AutoNAT-S [28]	5.36
	MAR [22]	6.36
Diffusion	VQ-Diffusion [12]	19.75
	Frido [10]	8.97
	U-ViT-S/2 [2]	5.48
Hi-DiT	Ours	**5.15**

Table 3: Quantitative comparison of Generation@512 on ImageNet. All experiments are conducted under classifier-free guidance (CFG).

Method	**Class**	**Generation@512**			
		FID ↓	IS ↑	Prec. ↑	Rec. ↑
VAR [43]	AR	2.63	303.2	–	–
xAR [35]	AR	1.70	281.5	–	–
DiT [31]	Latent Diff.	3.04	240.8	0.84	0.54
SiT [25]	Latent Diff.	2.62	252.2	0.84	0.57
REPA [51]	Latent Diff.	2.08	274.6	0.83	0.58
ADM [9]	Pixel Diff.	3.85	221.7	0.84	0.53
SiD2 [16]	Pixel Diff.	1.50	–	–	–
JiT-G [21]	Pixel Diff.	1.78	306.8	–	–
Hi-DiT(CFG=4.6)	Latent+Pixel	**1.26**	295.0	0.80	0.62
Hi-DiT(CFG=5.8)	Latent+Pixel	1.31	**310.2**	0.80	0.61

precision of pixel models, unlocking highly scalable and robust image generation. Moreover, slightly increasing the CFG scale can significantly improve IS with only a marginal impact on FID. Therefore, we report results under two different CFG settings for experiments at both resolutions.

A closer analysis reveals complementary bottlenecks in existing paradigms. Pixel-space diffusion models typically patchify raw images and denoise visual tokens, but must learn global low-frequency structure and local high-frequency details simultaneously, often leading to slower convergence and weaker performance under comparable compute than VAE-based approaches. In contrast, latent diffusion models are constrained by VAE reconstruction distortion: even recent efforts such as VA-VAE [49] that improve the autoencoder via representation alignment still denoise exclusively in a compressed latent manifold, which limits the recovery of fine-grained pixel details. This motivates our hybrid design that retains latent-space semantic efficiency while introducing a late-stage, time-gated pixel pathway to restore high-frequency details.

4.4 Result on Text-to-Image Generation

Table 2 presents a performance comparison of text-to-image generation on the MS-COCO dataset. For a fair comparison, we configure a lightweight Hi-DiT model by strictly adopting the parameter settings and Transformer backbone of the U-ViT-S/2 (Deep) architecture. The evaluated baselines are broadly categorized into three paradigms: GAN-based, autoregressive (AR), and diffusion-based models. While traditional diffusion models heavily relied on convolutional U-Nets, recent architectures like U-ViT have replaced them with Transformer blocks, achieving superior results. This underscores the powerful performance and scalability of Diffusion Transformers as high-capacity generative backbones. Building upon the U-ViT foundation, our Hi-DiT framework integrates raw pixel inputs and high-frequency pixel outputs. This dual-stream design effectively mit-

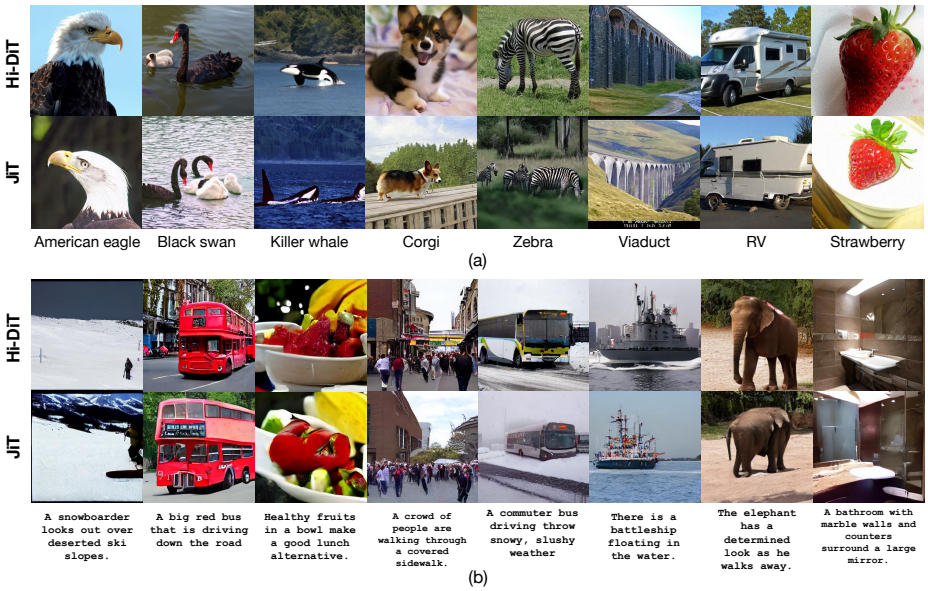

Fig. 3: Qualitative results on class-conditional and text-to-image generation. (a) shows a comparison between our method and pixel-space baseline on ImageNet-256 generation, while (b) presents text-to-image results on MS-COCO.

igates the loss of fine-grained details inherent in standard U-ViT, which conducts the denoising process exclusively within a compressed latent space.

4.5 Experimental Analysis

Ablation Study. In this section, we study how individual components contribute to Hi-DiT. For rapid and fair validation, all variants in Table 4 are trained from scratch for 80 epochs under identical settings. Starting from a vanilla pixel-stream baseline without the latent branch or our specialized pixel predictor, performance is markedly poor, reflecting the difficulty of directly optimizing raw image distributions from patchified pixel tokens alone. Adding the parallel latent branch yields a substantial improvement, indicating that the latent pathway provides an effective semantic scaffold in early denoising stages and makes optimization more tractable. Finally, equipping the pixel stream with our advanced pixel predictor further boosts performance, whereas replacing it with a simple MLP causes a noticeable drop. This confirms that naive projection is insufficient for recovering fine details from token features, and that our upsampling design reduces decoding difficulty and enables more effective high-frequency refinement.

Qualitative Results. To qualitatively assess the generative performance of Hi-DiT, Figure 3 presents a visual comparison of 16 images synthesized by the pure pixel-space baseline JiT and our latent-pixel Hi-DiT on ImageNet 256×256 and MS-COCO. Visual inspection shows that Hi-DiT achieves superior image

Table 4: Ablation study of Hi-DiT. "✗" or "✓" indicate w/o or w latent input.

Latent Input	Pixel Predictor	Params	FID↓
✗	MLP	682M	3.38
✓	MLP	682M	1.74
✓	HF Pixel Predictor	689M	**1.65**

Table 5: Ablation on the Time-Gating Threshold.

Timestep	FID↓	Inception Score↑
0.7	1.73	264.3
0.5	1.68	266.1
0.3	**1.65**	267.7
0.1	1.75	265.8

Table 6: Training and inference cost.

Model	Epoch	FID↓	Time/epoch↓	Time/img↓	Peak mem.
SiT [25]	1400	2.06	17.1min	1.52s	32.90G
VA-VAE [49]	800	1.35	17.7min	1.58s	34.43G
Hi-DiT	800	1.06	18.1min	1.67s	35.52G

quality with fewer distortion artifacts, benefiting from the pixel stream's capacity to model high-frequency pixel information. Moreover, in text-to-image comparisons, our generated images show tighter semantic alignment with input prompts, driven by the latent stream's robust low-frequency semantic modeling. These qualitative results validate the strong generative capability and generalization of our hybrid architecture.

Ablation on the Time-Gating Threshold τ. To gain deeper insights into our time-gating mechanism, we investigate the impact of different threshold values $\tau \in 0.1, 0.3, 0.5, 0.7$. All variants are trained with the backbone for 80 epochs under identical settings. As detailed in Table 5, the activation timing of the pixel stream significantly influences the model's generative capability. If the pixel stream is introduced too late in the denoising trajectory (e.g., $\tau = 0.1$), its participation in modeling low-frequency semantic information is insufficient, leading to a performance degradation. Conversely, activating the pixel stream too early (e.g., $\tau = 0.7$) forces the shared backbone to simultaneously predict low-frequency semantics and high-frequency pixel details during early high-noise stages. This reintroduces severe capacity competition, which detrimentally affects the model's overall efficacy. Consequently, we select a well-balanced threshold of $\tau = 0.3$ as the default setting for all Hi-DiT configurations.

Training and inference cost. Table 6 reports the training and inference costs of different models under the same evaluation setting. Compared with SiT [25] and VA-VAE [49], Hi-DiT introduces only a modest computational overhead. Specifically, Hi-DiT takes 1.67 seconds per image during inference, compared with 1.52 seconds for SiT and 1.58 seconds for VA-VAE. The training time also increases slightly, from 17.1/17.7 minutes per epoch to 18.1 minutes per epoch, while the peak memory rises to 35.52G. Despite this small overhead, Hi-DiT achieves the best generation quality with a 1.06 FID, suggesting a favorable trade-off between efficiency and performance.

FID–IS trade-off under CFG. Since different generative models may favor different classifier-free guidance (CFG) scales, we further evaluate the robustness of Hi-DiT under varying CFG strengths in a controlled manner. Specifically, we sweep CFG scales over a wide range for Hi-DiT and representative baselines, including autoregressive models, latent diffusion models, and pixel-space diffusion models. As shown in Figure 4, Hi-DiT consistently forms a better FID-IS trade-off curve than all baselines across different CFG settings. This indicates that the performance gain of Hi-DiT is robust to CFG selection, rather than being tied to a particular guidance scale.

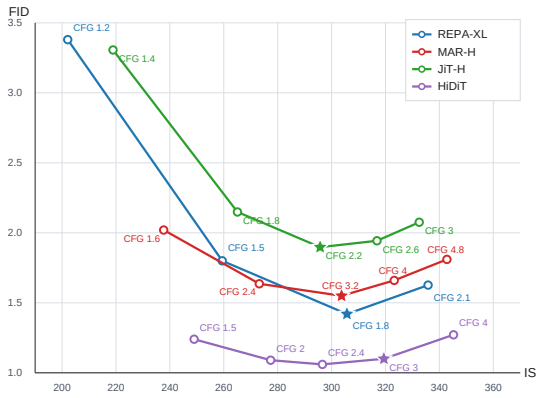

Fig. 4: FID–IS trade-off under CFG. Better methods lie toward the lower-right, $\star$ denote the default CFG setting of each method.

5 Conclusion

We propose the Hybrid Latent-Pixel Diffusion Transformer (Hi-DiT), a dual-stream framework that alleviates capacity competition in high-resolution image generation through temporal specialization. Within a shared backbone, Hi-DiT uses a Latent Stream for early global semantic formation and a Pixel Stream for late-stage fine-detail refinement. With Time-Gated Injection and a lightweight High-Frequency Pixel Predictor, Hi-DiT mitigates objective interference. Extensive ImageNet experiments show that this time-aware capacity allocation improves generation quality and detail fidelity, providing an efficient paradigm for next-generation diffusion models.

Acknowledgements

This work was supported by the Key Science and Technology Project of Anhui Province No. 202523o09050002.

References

1. Balaji, Y., Nah, S., Huang, X., Vahdat, A., Song, J., Zhang, Q., Kreis, K., Aittala, M., Aila, T., Laine, S., et al.: ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. arXiv preprint arXiv:2211.01324 (2022)
2. Bao, F., Nie, S., Xue, K., Cao, Y., Li, C., Su, H., Zhu, J.: All are worth words: A vit backbone for diffusion models. In: CVPR. pp. 22669–22679 (2023)
3. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: ICML. pp. 41–48 (2009)
4. Cai, Q., Chen, J., Gao, C., Gong, Z., Li, Y., Pan, Y., Peng, Y., Qiu, Z., Yu, K., Zhang, Y., et al.: Hidream-o1-image: A natively unified image generative foundation model with pixel-level unified transformer. arXiv preprint arXiv:2605.11061
5. Cai, Q., Li, Y., Pan, Y., Yao, T., Mei, T.: Hidream-i1: An open-source high-efficient image generative foundation model. In: ACMMM. pp. 13636–13639 (2025)
6. Chen, S., Ge, C., Zhang, S., Sun, P., Luo, P.: Pixelflow: Pixel-space generative models with flow. arXiv preprint arXiv:2504.07963 (2025)
7. Chen, Z., Zhu, J., Chen, X., Zhang, J., Hu, X., Zhao, H., Wang, C., Yang, J., Tai, Y.: Dip: Taming diffusion models in pixel space. arXiv preprint arXiv:2511.18822 (2025)
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255. IEEE (2009)
9. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. NeurIPS pp. 8780–8794 (2021)
10. Fan, W.C., Chen, Y.C., Chen, D., Cheng, Y., Yuan, L., Wang, Y.C.F.: Frido: Feature pyramid diffusion for complex scene image synthesis. In: AAAI. vol. 37, pp. 579–587 (2023)
11. Gao, Y., Shen, Y., Zhang, S., Yu, W., Duan, Y., Wu, J., Deng, J., Zhang, Y., et al.: Drift-based policy optimization: Native one-step policy learning for online robot control. arXiv preprint arXiv:2604.03540 (2026)
12. Gu, S., Chen, D., Bao, J., Wen, F., Zhang, B., Chen, D., Yuan, L., Guo, B.: Vector quantized diffusion model for text-to-image synthesis. In: CVPR. pp. 10696–10706 (2022)
13. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. NeurIPS (2017)
14. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. NeurIPS pp. 6840–6851 (2020)
15. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)
16. Hoogeboom, E., Mensink, T., Heek, J., Lamerigts, K., Gao, R., Salimans, T.: Simpler diffusion: 1.5 fid on imagenet512 with pixel-space diffusion. In: CVPR. pp. 18062–18071 (2025)
17. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
18. Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., Aila, T.: Improved precision and recall metric for assessing generative models. NeurIPS (2019)
19. Leng, X., Singh, J., Hou, Y., Xing, Z., Xie, S., Zheng, L.: Repa-e: Unlocking vae for end-to-end tuning of latent diffusion transformers. In: ICCV. pp. 18262–18272 (2025)

20. Li, R., Zhang, F., Li, T., Zhang, N., Zhang, T.: Dmgan: Dynamic multi-hop graph attention network for traffic forecasting. IEEE TKDE **35**(9), 9088–9101 (2022)
21. Li, T., He, K.: Back to basics: Let denoising generative models denoise. arXiv preprint arXiv:2511.13720 (2025)
22. Li, T., Tian, Y., Li, H., Deng, M., He, K.: Autoregressive image generation without vector quantization. NeurIPS pp. 56424–56445 (2024)
23. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV. pp. 740–755. Springer (2014)
24. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
25. Ma, N., Goldstein, M., Albergo, M.S., Boffi, N.M., Vanden-Eijnden, E., Xie, S.: Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In: ECCV. pp. 23–40. Springer (2024)
26. Ma, Z., Wei, L., Wang, S., Zhang, S., Tian, Q.: Deco: Frequency-decoupled pixel diffusion for end-to-end image generation. arXiv preprint arXiv:2511.19365 (2025)
27. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.Y., Ermon, S.: Sdedit: Guided image synthesis and editing with stochastic differential equations. arXiv preprint arXiv:2108.01073 (2021)
28. Ni, Z., Wang, Y., Zhou, R., Guo, J., Hu, J., Liu, Z., Song, S., Yao, Y., Huang, G.: Revisiting non-autoregressive transformers for efficient image synthesis. In: CVPR. pp. 7007–7016 (2024)
29. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
30. Pan, Y., Li, Y., Yao, T., Ngo, C.W., Mei, T.: Stream-vit: learning streamlined convolutions in vision transformer. IEEE Transactions on Multimedia (2025)
31. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: ICCV. pp. 4195–4205 (2023)
32. Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., Rombach, R.: Sdxl: Improving latent diffusion models for high-resolution image synthesis. ICLR (2025)
33. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763 (2021)
34. Rahaman, N., Baratin, A., Arpit, D., Draxler, F., Lin, M., Hamprecht, F., Bengio, Y., Courville, A.: On the spectral bias of neural networks. In: ICML. pp. 5301–5310 (2019)
35. Ren, S., Yu, Q., He, J., Shen, X., Yuille, A., Chen, L.C.: Beyond next-token: Next-x prediction for autoregressive visual generation. In: ICCV. pp. 15781–15791 (2025)
36. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR. pp. 10684–10695 (2022)
37. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI. pp. 234–241. Springer (2015)
38. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. NeurIPS (2016)
39. Shen, Y., Zhang, X., Duan, Y., Zhang, S., Li, H., Wu, Y., Ji, J., Zhang, Y., Jin, H.: Og-gaussian: Occupancy based street gaussians for autonomous driving. In: ICRA. pp. 3291–3297 (2025)

40. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: CVPR. pp. 1874–1883 (2016)
41. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)
42. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456 (2020)
43. Tian, K., Jiang, Y., Yuan, Z., Peng, B., Wang, L.: Visual autoregressive modeling: Scalable image generation via next-scale prediction. NeurIPS pp. 84839–84865 (2024)
44. Van Den Oord, A., Vinyals, O., et al.: Neural discrete representation learning. NeurIPS (2017)
45. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. NeurIPS (2017)
46. Wang, S., Tian, Z., Huang, W., Wang, L.: Ddt: Decoupled diffusion transformer. arXiv preprint arXiv:2504.05741 (2025)
47. Wu, A., Qiu, Z., Yao, T., Mei, T.: Ps-sr: Pseudo-single-step video super-resolution via speculative diffusion. In: CVPR. pp. 38218–38227 (2026)
48. Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., He, X.: Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In: CVPR. pp. 1316–1324 (2018)
49. Yao, J., Yang, B., Wang, X.: Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In: CVPR. pp. 15703–15712 (2025)
50. Yu, Q., He, J., Deng, X., Shen, X., Chen, L.C.: Randomized autoregressive visual generation. In: CVPR. pp. 18431–18441 (2025)
51. Yu, S., Kwak, S., Jang, H., Jeong, J., Huang, J., Shin, J., Xie, S.: Representation alignment for generation: Training diffusion transformers is easier than you think. arXiv preprint arXiv:2410.06940 (2024)
52. Yu, Y., Xiong, W., Nie, W., Sheng, Y., Liu, S., Luo, J.: Pixeldit: Pixel diffusion transformers for image generation. arXiv preprint arXiv:2511.20645 (2025)
53. Zhang, H., Koh, J.Y., Baldridge, J., Lee, H., Yang, Y.: Cross-modal contrastive learning for text-to-image generation. In: CVPR. pp. 833–842 (2021)
54. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: ICCV. pp. 3836–3847 (2023)
55. Zhang, S., Zhang, S., Deng, J., Shen, Y., Ma, M., Zhang, Y.: Pgov3d: Open-vocabulary 3d semantic segmentation with partial-to-global curriculum. In: ACMMM. pp. 5070–5079 (2025)
56. Zhang, Y., Qiu, Z., Liu, Z., Pan, Y., Yao, T., Mei, T.: Evoid: Reinforced evolution for identity-preserving video generation. In: CVPR (2026)
57. Zhang, Z., Long, F., Pan, Y., Qiu, Z., Yao, T., Cao, Y., Mei, T.: Trip: Temporal residual learning with image noise prior for image-to-video diffusion models. In: CVPR. pp. 8671–8681 (2024)
58. Zheng, B., Ma, N., Tong, S., Xie, S.: Diffusion transformers with representation autoencoders. arXiv preprint arXiv:2510.11690 (2025)
59. Zheng, H., Nie, W., Vahdat, A., Anandkumar, A.: Fast training of diffusion models with masked transformers. arXiv preprint arXiv:2306.09305 (2023)