DiffiT: Diffusion Vision Transformers for Image Generation

Ali Hatamizadeh, Jiaming Song, Guilin Liu, Jan Kautz, Arash Vahdat

NVIDIA
{ahatamizadeh, jiamings, guilinl, jkautz, avahdat}@nvidia.com
https://github.com/NVlabs/DiffiT

Fig. 1 – Uncurated generated images by latent DiffiT on ImageNet [15] dataset.

Abstract. Diffusion models with their powerful expressivity and high sample quality have achieved State-Of-The-Art (SOTA) performance in the generative domain. The pioneering Vision Transformer (ViT) has also demonstrated strong modeling capabilities and scalability, especially for recognition tasks. In this paper, we study the effectiveness of ViTs in diffusion-based generative learning and propose a new model denoted as Diffusion Vision Transformers (DiffiT). Specifically, we propose a methodology for finegrained control of the denoising process and introduce the Time-dependant Multihead Self Attention (TMSA) mechanism. DiffiT is surprisingly effective in generating high-fidelity images with significantly better parameter efficiency. We also propose *latent* and *image* space DiffiT models and show SOTA performance on a variety of class-conditional and unconditional synthesis tasks at different resolutions. The Latent DiffiT model achieves a new SOTA FID score of **1.73** on **ImageNet-256** dataset while having **19.85**%, **16.88**% less parameters than other Transformer-based diffusion models such as MDT and DiT, respectively.

1   Introduction

Diffusion models [28,68,71] have revolutionized the domain of generative learning, with successful frameworks in the front line such as DALL·E 3 [58], Imagen [27] and Stable diffusion [59,60] and achieving state-of-the-art (SOTA) performance in various tasks. They have enabled generating diverse complex scenes in high fidelity which were once considered out of reach for prior models. Specifically, synthesis in diffusion models is formulated as an iterative process in which random image-shaped Gaussian noise is denoised gradually towards realistic samples [28,68,71]. The core building block in this process is a *denoising autoencoder network* that takes a noisy image and predicts the denoising direction, equivalent to *the score function* [32,76]. This network, which is shared across different time steps of the denoising process, is often a variant of Convolutional Neural Network (CNN)-based U-Net [28,61]. However, with a lack of standard design pattern, several architecture variants [16,53,70] have been proposed for the denoising network.

Vision Transformers (ViTs) [20] have demonstrated SOTA performance for various recognition tasks and offer compelling advantages such as long-range dependency modeling and scalability. Recently, a number of efforts such as Diffusion Transformers (DiT) [56] and Masked Diffusion Transformer (MDT) [23] have proposed to leverage the strong modeling capability and scalability of ViTs for diffusion-based image generation. In DiT and MDT, Adaptive LayerNorm (AdaLN) [57] is used for input noise conditioning. However, this scheme significantly increases the number of parameters and does not effectively model the unique temporal dynamics of the denoising process [13,44]. Specifically, in the beginning of denoising, the high-frequency content of the image is completely perturbed as the denoising network primarily focuses on predicting the low-frequency content. Towards the end of denoising, in which most of the image structure is generated, the network tends to focus on predicting high-frequency details. The conditioning in DiT is realized by modulating the input with channel-wise scale and shift parameters predicted by adaLN layers. However, this mechanism cannot
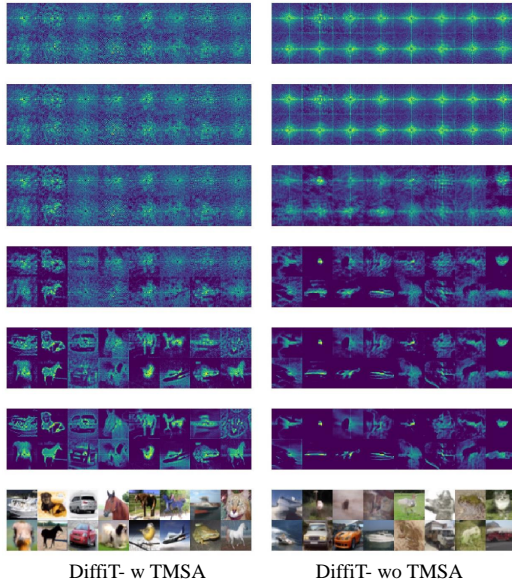

DiffiT- w TMSA   DiffiT- wo TMSA

Fig. 2 – Side-by-side qualitative comparison of attention maps during the denoising process for models with and without TMSA. The denoising process starts from the top row in each column.

optimally capture the dynamics of the denoising process since it does not effectively model the joint spatial and temporal dependencies. In this work, we introduce the Time-dependant Multihead Self-Attention (TMSA) mechanism which allows for fine-grained control over spatial and temporal dependencies and their interaction during the denoising process. Specifically, our TMSA proposes to integrate the temporal component into the self-attention where the key, query, and value weights are adapted per time step during denoising. This allows the denoising network to dynamically change its attention mechanism in different stages by considering both spatial and temporal components and their correspondence. In Fig. 2, we visualize attention maps from a token at the center of a feature map to all surrounding tokens during the sampling trajectory of a models that are trained on CIFAR10 [45] dataset. The DiffiT model with TMSA has a better image generation quality and its attention maps demonstrate a progressive localization towards detailed salient features. However, the model without TMSA is not capable of recovering such details.

In addition, employing TMSA significantly improves the parameter efficiency as it only learns three temporal components for query, key and value in each block. In comparison, AdaLN requires learning the shift, scale and gate parameters for self-attention as well as MLP (*i.e.* six components per Transformer block). We also extend TMSA to a window-based scheme without cross-communication among the local regions. This design is surprisingly effective and decreases the computational cost of self-attention by reducing the token sequence length.

Using TMSA as a core building block, we introduce a novel ViT-based diffusion model, called DiffiT (pronounced *di-feet*), for image generation in latent and image space. DiffiT achieves a new SOTA performance in terms of FID score using ImageNet-256 [15] dataset (see Fig.1) with **19.85%**, **16.88%** less parameters than MDT and DiT models, respectively. DiffiT also achieves SOTA performance for image space generation tasks on FFHQ-64 [36] and CIFAR10 [45] datasets.

The following summarizes our contributions in this work:

- We introduce TMSA which is a novel time-dependent self-attention mechanism and is specifically tailored to capture both temporal and spatial dependencies as well as their interaction. Our proposed time-dependent self-attention dynamically adapts its behavior over sampling time steps.
- We introduce a new ViT-based diffusion model, denoted as DiffiT, which unifies the design patterns of denoising networks and can be used in a variety of image generation tasks in the latent and image space.
- We demonstrate that DiffiT can achieve SOTA performance on a variety of datasets for both conditional and unconditional generation tasks in the latent and image space. The latent DiffiT model achieves a new FID score of 1.73 with significantly less number of parameters than competing approaches.

2   Related Work

Diffusion Image Generation Diffusion models [28, 68, 71] have driven significant advances in various domains, such as text-to-image generation [8, 58, 63], natural

language processing [49], text-to-speech synthesis [43], 3D point cloud generation [82,83,89], time series modeling [72], molecular conformal generation [79], and machine learning security [54]. These models synthesize samples via an iterative denoising process and thus are also known in the community as noise-conditioned score networks. Since its initial success on small-scale datasets like CIFAR-10 [28], diffusion models have been gaining popularity compared to other existing families of generative models. Compared with variational autoencoders [42], diffusion models divide the synthesis procedure into small parts that are easier to optimize, and have better coverage of the latent space [4, 67, 74]; compared with generative adversarial networks [25], diffusion models have better training stability and are much easier to invert [22, 69]. Diffusion models are also well-suited for image restoration, editing and re-synthesis tasks with minimal modifications to the existing architecture [5,6,14,22,37–39,52,62,75], making it well-suited for various downstream applications.

Transformers in Generative Modeling Transformer-based models have achieved competitive performance in different generative learning models in the visual domain [12, 17, 18, 29, 84, 85]. A number of transformer-based architectures have emerged for GANs [47, 48, 80, 87]. TransGAN [33] proposed to use a pure transformer-based generator and discriminator architecture for pixel-wise image generation. Gansformer [31] introduced a bipartite transformer that encourages the similarity between latent and image features. Styleformer [55] uses Linformers [77] to scale the synthesis to higher resolution images. Recently, a number of efforts [9, 23, 50, 56] have leveraged Transformer-based architectures for diffusion models and achieved competitive performance. In particular, DiT [56] proposed a latent diffusion model in which the regular U-Net backbone is replaced with a Transformer. Using the DiT architecture, MDT [23] introduced a masked latent modeling approach to effectively capture contextual information. In comparison to DiT, although MDT achieves faster learning speed and better FID scores on ImageNet-256 dataset [15], it has a more complex training pipeline. Recently with a similar architecture to DiT, SiT [51] was proposed to incorporate flow matching [1, 2]. Unlike DiT, MDT or SiT, the proposed DiffiT does not use shift and scale, as in AdaLN formulation, for conditioning. Instead, DiffiT proposes a time-dependent self-attention (*i.e.* TMSA) to jointly learn the spatial and temporal dependencies. In addition, DiffiT proposes both image and latent space models for different image generation tasks with different resolutions with SOTA performance.

3 Methodology

3.1 Diffusion Model Preliminaries

Diffusion models [28, 68, 71] are a family of generative models that synthesize samples via an iterative denoising process. Given a data distribution as $q_0(\mathbf{z}_0)$, a family of random variables $\mathbf{z}_t$ for $t \in [0, T]$ are defined by injecting Gaussian noise to $\mathbf{z}_0$, *i.e.*, $q_t(\mathbf{z}_t) = \int q(\mathbf{z}_t|\mathbf{z}_0)q_0(\mathbf{z}_0)\mathrm{d}\mathbf{z}_0$, where $q(\mathbf{z}_t|\mathbf{z}_0) = \mathcal{N}(\mathbf{z}_0, \sigma_t^2 \boldsymbol{I})$ is a

Gaussian distribution. Typically, σ_t is chosen as a non-decreasing sequence such that $\sigma_0 = 0$ and σ_T being much larger than the data variance. This is called the "Variance-Exploding" noising schedule in the literature [71]; for simplicity, we use these notations throughout the paper, but we note that it can be equivalently converted to other commonly used schedules (such as "Variance-Preserving" [28]) by simply rescaling the data with a scaling term, dependent on t [34, 69].

The distributions of these random variables are the marginal distributions of forward diffusion processes (Markovian or not [69]) that gradually reduces the "signal-to-noise" ratio between the data and noise. As a generative model, diffusion models are trained to approximate the reverse diffusion process, that is, to transform from the initial noisy distribution (that is approximately Gaussian) to a distribution that is close to the data one.

Training Despite being derived from different perspectives, diffusion models can generally be written as learning the following denoising autoencoder objective [76]

$$\mathbb{E}_{q_0(\mathbf{z}_0), t \sim p(t), \epsilon \sim \mathcal{N}(0, \boldsymbol{I})}[\lambda(t)\|\epsilon - \epsilon_\theta(\mathbf{z}_0 + \sigma_t \epsilon, t)\|_2^2]. \tag{1}$$

Intuitively, given a noisy sample from $q(\mathbf{z}_t)$ (generated via $\mathbf{z}_t := \mathbf{z}_0 + \sigma_t \epsilon$), a neural network ϵ_θ is trained to predict the amount of noise added (*i.e.*, ϵ). Equivalently, the neural network can also be trained to predict $\mathbf{z}_0$ instead [28, 64]. The above objective is also known as denoising score matching [76], where the goal is to try to fit the data score (*i.e.*, $\nabla_{\mathbf{z}_t} \log q(\mathbf{z}_t)$) with a neural network, also known as the score network $s_\theta(\mathbf{z}_t, t)$. The score network can be related to ϵ_θ via the relationship $s_\theta(\mathbf{z}_t, t) := -\epsilon_\theta(\mathbf{z}_t, t)/\sigma_t$.

Sampling Samples from the diffusion model can be simulated by the following family of stochastic differential equations that solve from $t = T$ to $t = 0$ [19, 26, 34, 86]:

$$d\mathbf{z} = -(\dot{\sigma}_t + \beta_t)\sigma_t s_\theta(\mathbf{z}, t) dt + \sqrt{2\beta_t} \sigma_t d\omega_t, \tag{2}$$

where ω_t is the reverse standard Wiener process, and β_t is a function that describes the amount of stochastic noise during the sampling process. If $\beta_t = 0$ for all t, then the process becomes a probabilistic ordinary differential equation [3] (ODE), and can be solved by ODE integrators such as denoising diffusion implicit models (DDIM [69]). Otherwise, solvers for stochastic differential equations (SDE) can be used, including the one for the original denoising diffusion probabilistic models (DDPM [28]). Typically, ODE solvers can converge to high-quality samples in fewer steps and SDE solvers are more robust to inaccurate score models [34].

3.2 DiffiT Model

Time-dependent Self-Attention At every layer, our transformer block receives $\{\mathbf{x_s}\}$, a set of tokens arranged spatially on a 2D grid in its input. It also receives
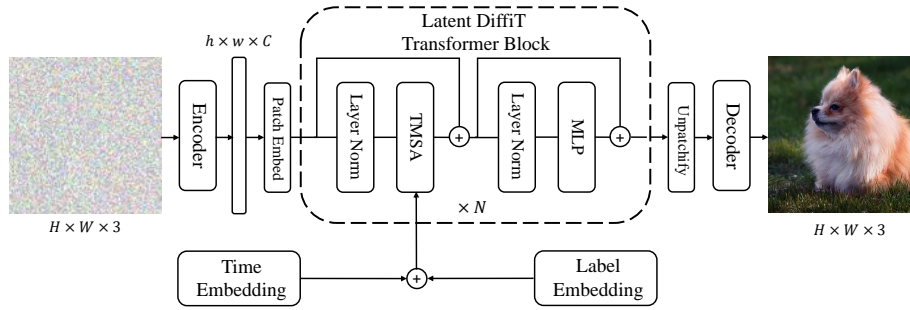
Fig. 3 – Overview of the latent DiffiT framework.

$\mathbf{x_t}$, a time token representing the time step. Similar to [28], we obtain the time token by feeding positional time embeddings to a small MLP with swish activation [21]. This time token is passed to all layers in our denoising network. We introduce our time-dependent multi-head self-attention, which captures both long-range spatial and temporal dependencies by projecting feature and time token embeddings in a shared space. Specifically, time-dependent queries $\mathbf{q}$, keys $\mathbf{k}$ and values $\mathbf{v}$ in the shared space are computed by a linear projection of spatial and time embeddings $\mathbf{x_s}$ and $\mathbf{x_t}$ via

$$\mathbf{q_s} = \mathbf{x_s}\mathbf{W}_{qs} + \mathbf{x_t}\mathbf{W}_{qt}, \tag{3}$$

$$\mathbf{k_s} = \mathbf{x_s}\mathbf{W}_{ks} + \mathbf{x_t}\mathbf{W}_{kt}, \tag{4}$$

$$\mathbf{v_s} = \mathbf{x_s}\mathbf{W}_{vs} + \mathbf{x_t}\mathbf{W}_{vt}, \tag{5}$$

where $\mathbf{W}_{qs}$, $\mathbf{W}_{qt}$, $\mathbf{W}_{ks}$, $\mathbf{W}_{kt}$, $\mathbf{W}_{vs}$, $\mathbf{W}_{vt}$ denote spatial and temporal linear projection weights for their corresponding queries, keys, and values respectively.

We note that the operations listed in Eq. 3 to 5 are equivalent to a linear projection of each spatial token, concatenated with the time token. As a result, key, query, and value are all linear functions of both time and spatial tokens and they can adaptively modify the behavior of attention for different time steps. We define $\mathbf{Q} := \{\mathbf{q_s}\}$, $\mathbf{K} := \{\mathbf{k_s}\}$, and $\mathbf{V} := \{\mathbf{v_s}\}$ which are stacked form of query, key, and values in rows of a matrix. The self-attention is then computed as follows

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}} + \mathbf{B}\right)\mathbf{V}. \tag{6}$$

In which, d is a scaling factor for keys $\mathbf{K}$, and $\mathbf{B}$ corresponds to a relative position bias [66]. For computing the attention, the relative position bias allows for the encoding of information across each attention head. Note that although the relative position bias is implicitly affected by the input time embedding, directly integrating it with this component may result in sub-optimal performance as it needs to capture both spatial and temporal information. Please see Sec. 5.4 for more analysis.

DiffiT Transformer Block The transformer block is a core building block of the proposed DiffiT architecture and is defined as

$$\hat{\mathbf{x}}_{\mathbf{s}} = \text{TMSA}\left(\text{LN}\left(\mathbf{x}_{\mathbf{s}}\right), \mathbf{x}_{\mathbf{t}}\right) + \mathbf{x}_{\mathbf{s}}, \tag{7}$$

$$\mathbf{x}_{\mathbf{s}} = \text{MLP}\left(\text{LN}\left(\hat{\mathbf{x}}_{\mathbf{s}}\right)\right) + \hat{\mathbf{x}}_{\mathbf{s}}, \tag{8}$$

where TMSA denotes time-dependent multi-head self-attention, as described in the above, $\mathbf{x}_{\mathbf{t}}$ is the time-embedding token, $\mathbf{x}_{\mathbf{s}}$ is a spatial token, and LN and MLP denote Layer Norm [7] and MLP respectively.

Latent Space Recently, latent diffusion models have been shown effective in generating high-quality large-resolution images [59, 74]. In Fig. 3, we show the architecture of latent DiffiT model. We first encode the images using a pre-trained variational auto-encoder network [59]. The feature maps are then converted into non-overlapping patches and projected into a new embedding space. Similar to the DiT model [56], we use a vision transformer, without upsampling or downsampling layers, as the denoising network in the latent space. In addition, we also utilize a three-channel classifier-free guidance to improve the quality of generated samples. The final stage is a linear layer to decode the output.

Image Space

DiffiT Architecture As shown in Fig. 4, DiffiT uses a symmetrical U-Shaped encoder-decoder architecture in which the contracting and expanding paths are connected to each other via skip connections at every resolution. Specifically, each resolution of the encoder or decoder paths consists of L consecutive DiffiT blocks, containing our proposed time-dependent self-attention modules. In the beginning of each path, for both the encoder and decoder, a convolutional layer is employed to match the number of feature maps. A convolutional upsampling or downsampling layer is also used for transitioning between each resolution. We speculate that the use of these convolutional layers embeds inductive image bias that can further improve the performance. In the remainder of this section, we discuss the DiffiT Transformer block and our proposed time-dependent self-attention mechanism. We use our proposed Transformer block as the residual cells when constructing the U-shaped denoising architecture.

Local Attention The quadratic cost of attention scales poorly when the number of spatial tokens is large, especially in the case of large feature maps. Without loss of generality, the above Transformer block can be applied to local regions, in which the self-attention is computed within non-overlapping partitioned windows. Although these partitioned windows do not allow information to be propagated between different regions, the U-Net structure with bottleneck layers permits information sharing between different regions.
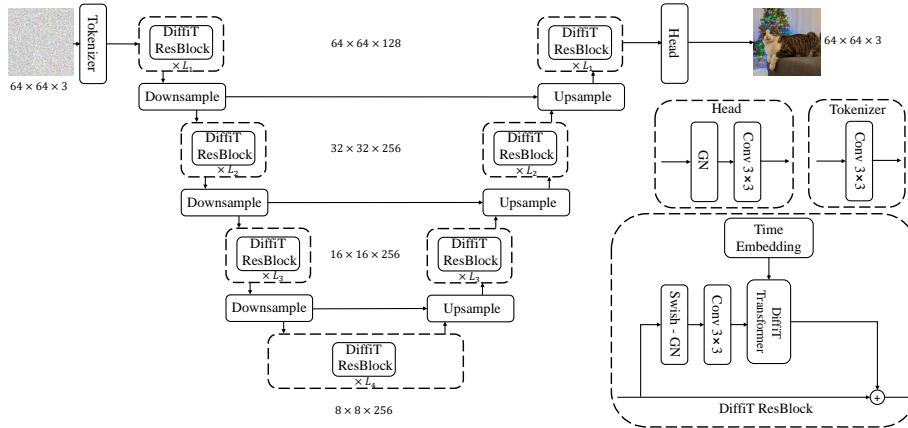
Fig. 4 – Overview of the image-space DiffiT model. Downsample and Upsample denote convolutional downsampling and upsampling layers, respectively. Please see the supplementary materials for more information regarding the DiffiT architecture.

DiffiT ResBlock We define our final residual cell by combining our proposed DiffiT Transformer block with an additional convolutional layer in the form:

$$\hat{\mathbf{x}}_{\mathbf{s}} = \mathrm{Conv}_{3 \times 3} \left(\mathrm{Swish} \left(\mathrm{GN} \left(\mathbf{x}_{\mathbf{s}} \right) \right) \right), \tag{9}$$

$$\mathbf{x}_{\mathbf{s}} = \mathrm{DiffiT\text{-}Block} \left(\hat{\mathbf{x}}_{\mathbf{s}}, \mathbf{x}_{\mathbf{t}} \right) + \mathbf{x}_{\mathbf{s}}, \tag{10}$$

where GN denotes the group normalization operation [78] and DiffiT-Transformer is defined in Eq. 7 and Eq. 8. Our residual cell for image space diffusion models is a hybrid cell combining both a convolutional layer and our Transformer block.

4   Results

4.1   Latent Space

We have trained the latent DiffiT model on ImageNet-512 and ImageNet-256 dataset respectively. In Table. 1, we present a comparison against other approaches using various image quality metrics. For this comparison, we select the best performance metrics from each model which may include techniques such as classifier-free guidance. In ImageNet-256 dataset, the latent DiffiT model outperforms competing approaches, such as SiT-XL [51], MDT-G [23], DiT-XL/2-G [56] and StyleGAN-XL [65], in terms of FID score and sets a new SOTA FID score of **1.73**. In terms of other metrics such as IS and sFID, the latent DiffiT model shows a competitive performance, hence indicating the effectiveness of the proposed time-dependent self-attention. In the ImageNet-512 dataset, the latent DiffiT model significantly outperforms DiT-XL/2-G in terms of both FID and Inception Score (IS). Although StyleGAN-XL [65] shows better performance in

Model	Class	ImageNet-256				ImageNet-512			
		FID ↓	IS ↑	Precision ↑	Recall ↑	FID ↓	IS ↑	Precision ↑	Recall ↑
LDM-4 [59]	Diffusion	10.56	103.49	0.71	**0.62**	-	-	-	-
BigGAN-Deep [10]	GAN	6.95	171.40	**0.87**	0.28	8.43	177.90	**0.88**	0.29
MaskGIT [11]	Masked Modeling	4.02	**355.60**	<u>0.83</u>	0.44	4.46	**342.00**	0.83	0.50
RQ-Transformer [46]	Autoregressive	3.80	<u>323.70</u>	-	-	-	-	-	-
ADM-G-U [16]	Diffusion	3.94	215.84	<u>0.83</u>	0.53	3.85	221.72	<u>0.84</u>	<u>0.53</u>
LDM-4-G [59]	Diffusion	3.60	247.67	**0.87**	0.48	-	-	-	-
Simple Diffusion [30]	Diffusion	2.77	211.80	-	-	3.54	205.30	-	-
DiT-XL/2-G [56]	Diffusion	2.27	<u>278.24</u>	<u>0.83</u>	0.57	3.04	240.82	<u>0.84</u>	0.54
StyleGAN-XL [65]	GAN	2.30	265.12	0.78	0.53	**2.41**	<u>267.75</u>	0.77	0.52
MDT-G [23]	Diffusion	<u>1.79</u>	283.01	0.81	<u>0.61</u>	-	-	-	-
SiT-XL [51]	Diffusion	2.06	270.27	0.82	0.59	-	-	-	-
DiffiT	Diffusion	**1.73**	276.49	0.80	**0.62**	<u>2.67</u>	252.12	0.83	**0.55**

Table 1 – Comparison of image generation performance against state-of-the-art models on ImageNet-256 and ImageNet-512 dataset. The latent DiffiT model achieves SOTA performance in terms of FID score on ImageNet-256 dataset.

terms of FID and IS, GAN-based models are known to suffer from issues such as low diversity that are not captured by the FID score. These issues are reflected in sub-optimal performance of StyleGAN-XL in terms of both Precision and Recall. In addition, in Fig. 5, we show a visualization of uncurated images that are generated on ImageNet-256 and ImageNet-512 dataset. We observe that the latent DiffiT model is capable of generating diverse high quality images across different classes.

4.2 Image Space

We have trained the image space DiffiT model on FFHQ-64 [36] and CIFAR10 [45] datasets. In Table. 2, we compare the performance of our model against a variety of different generative models including other score-based diffusion models as well as GANs, and VAEs. DiffiT achieves a state-of-the-art image generation FID score of 1.95 on the CIFAR-10 dataset, outperforming state-of-the-art diffusion models such as EDM [34] and LSGM [74]. In comparison to two recent ViT-based diffusion models, our proposed DiffiT significantly outperforms U-ViT [9] and GenViT [81] models in terms of FID score

Table 2 – FID performance comparison against various generative approaches on the CIFAR10, FFHQ-64 datasets. VP and VE denote Variance Preserving and Variance Exploding respectively.

Method	Class	Space Type	CIFAR-10 32×32	FFHQ 64×64
NVAE [73]	VAE	-	23.50	-
GenViT [81]	Diffusion	Image	20.20	-
AutoGAN [24]	GAN	-	12.40	-
TransGAN [33]	GAN	-	9.26	-
INDM [40]	Diffusion	Latent	3.09	-
DDPM++ (VE) [71]	Diffusion	Image	3.77	25.95
U-ViT [9]	Diffusion	Image	3.11	-
DDPM++ (VP) [71]	Diffusion	Image	3.01	3.39
StyleGAN2 w/ ADA [35]	GAN	-	2.92	-
LSGM [74]	Diffusion	Latent	2.01	-
EDM (VE) [34]	Diffusion	Image	2.01	2.53
EDM (VP) [34]	Diffusion	Image	1.99	2.39
DiffiT (Ours)	Diffusion	Image	**1.95**	**2.22**

in CIFAR-10 dataset. Additionally, DiffiT significantly outperforms EDM [34] and DDPM++ [71] models, both on VP and VE training configurations, in terms of FID score. In Fig. 6, we illustrate the generated images on FFHQ-64 dataset. Please see supplementary materials for CIFAR-10 generated images.

Fig. 5 – Visualization of uncurated generated images on ImageNet-256 and ImageNet-512 datasets by latent DiffiT model.

5   Ablation

In this section, we provide additional ablation studies to provide insights into DiffiT. We address different questions such as: (1) What strikes the right balance between time and feature token dimensions ? (2) How do different components of DiffiT contribute to the final generation performance, (3) What is the optimal way of introducing time dependency in our Transformer block? and (4) How does our time-dependent attention behave as a function of time?

5.1   Time and Feature Token Dimensions

We conduct experiments to study the effect of the size of time and feature token dimensions on the overall performance. As shown below, we observe degradation of performance when the token

Time Dimension	Dimension	CIFAR10	FFHQ64
512	512	1.99	2.27
256	256	2.13	2.41
512	512	1.95	2.22

Table 3 – Ablation study on the effectiveness of time and feature dimensions.

dimension is increased from 256 to 512. Furthermore, decreasing the time embedding dimension from 512 to 256 impacts the performance negatively.

Fig. 6 – Visualization of uncurated generated images for FFHQ-64 dataset. Best viewed in color.

5.2 Effect of Architecture Design

As presented in Table 4, we study the effect of various components of both encoder and decoder in the architecture design on the image generation performance in terms of FID score on CIFAR-10. For these experiments, the projected temporal component is adaptively scaled and simply added to the spatial component in each stage. We start from the original ViT [20] base model with 12 layers and employ it as the encoder (config A). For the decoder, we use the Multi-Level Feature Aggregation variant of SETR [88] (SETR-MLA) to generate images in the input resolution. Our experiments show this architecture is sub-optimal as it yields a final FID score of 5.34. We hypothesize this could be due to the isotropic architecture of ViT which does not allow learning representations at multiple scales. We then extend the encoder ViT into 4 different multi-resolution stages with a convolutional layer in between each stage for downsampling (config B). We denote this setup as Multi-Resolution and observe that these changes and learning multi-scale feature representations in the encoder sub-

Config	Encoder	Decoder	FID Score
A	ViT [20]	SETR-MLA [88]	5.34
B	+ Multi-Resolution	SETR-MLA [88]	4.64
C	Multi-Resolution	+ Multi-Resolution	3.71
D	+ DiffiT Encoder	Multi-Resolution	2.27
E	+ DiffiT Encoder	+ DiffiT Decoder	**1.95**

Table 4 – Ablation study on the effectiveness of encoder and decoder architecture.

stantially improve the FID score to 4.64. In addition, instead of SETR-MLA [88] decoder, we construct a symmetric U-like architecture by using the same Multi-Resolution setup except for using convolutional layers between stages for upsampling (config C). These changes further improve the FID score to 3.71. Furthermore, we first add the DiffiT Transformer blocks and construct a DiffiT Encoder and observe that FID scores substantially improve to 2.27 (config D). As a result, this validates the effectiveness of the proposed TMSA in which the self-attention models both spatial and temporal dependencies. Using the DiffiT decoder further improves the FID score to 1.95 (config E), hence demonstrating the importance of DiffiT Transformer blocks for decoding.

5.3   Time-Dependent Self-Attention

We evaluate the effectiveness of our proposed TMSA layers in a generic denoising network. Specifically, using the DDPM++ [71] model, we replace the original self-attention layers with TMSA

Model	TMSA	FID Score
DDPM++(VE) [71]	No	3.77
DDPM++(VE) [71]	Yes	3.49
DDPM++(VP) [71]	No	3.01
DDPM++(VP) [71]	Yes	2.76

Table 5 – Effectiveness of TMSA.

layers for both VE and VP settings for image generation on the CIFAR10 dataset. Note that we did not change the original hyper-parameters for this study. As shown in Table 5 employing TMSA decreases the FID scores by 0.28 and 0.25 for VE and VP settings respectively. These results demonstrate the effectiveness of the proposed TMSA to dynamically adapt to different sampling steps and capture temporal information.

5.4   Impact of Self-Attention Components

In Table 6, we study different design choices for introducing time-dependency in self-attention layers. In the first baseline, we remove the temporal component from our proposed TMSA and we only

Config	Component	FID Score
F	Relative Position Bias	3.97
G	MLP	3.81
H	TMSA	**1.95**

Table 6 – Effectiveness of different components.

add the temporal tokens to relative positional bias (config F). We observe a significant increase in the FID score to 3.97 from 1.95. In the second baseline, instead of using relative positional bias, we add temporal tokens to the MLP layer of DiffiT Transformer block (config G). We observe that the FID score slightly improves to 3.81, but it is still suboptimal compared to our proposed TMSA (config H). Hence, this experiment validates the effectiveness of our proposed TMSA that integrates time tokens directly with spatial tokens when forming queries, keys, and values in self-attention layers.

5.5   Time Token in TMSA

We investigate if treating time embedding as a separate token in TMSA is a beneficial choice. Specifically, we apply self-attention to spatial and time tokens separately to un-

Model	TMSA Design	CIFAR10	FFHQ64
DiffiT	Separate	2.28	2.59
DiffiT	Mixed	**1.95**	**2.22**

Table 7 – Impact of time embedding.

derstand the impact of decoupling them. As shown in Table 7, we observe the degradation of performance for CIFAR10, FFHQ64 datasets, in terms of FID score. Hence, the decoupling of spatial and temporal information in TMSA leads to suboptimal performance.
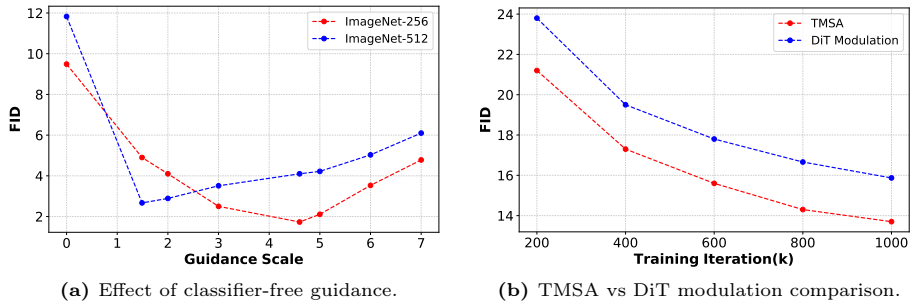
(a) Effect of classifier-free guidance.

(b) TMSA vs DiT modulation comparison.

Fig. 7 – (a) Effect of classifier-free guidance scale on FID score for ImageNet-256 and ImageNet-512. (b) Performance comparison of TMSA and DiT modulation.

5.6   Time Embedding

We study the sensitivity of the DiffiT model to different time embeddings such as Fourier and positional time embeddings. As shown in Table 8, using a Fourier time embedding leads to degradation of performance in terms of FID score for both CIFAR10 [45] and FFHQ-64 [36] datasets.

Model	Time Embedding	CIFAR10	FFHQ64
DiffiT	Fourier	2.02	2.37
DiffiT	Positional	**1.95**	**2.22**

Table 8 – Effectiveness of TMSA time token.

5.7   Computational Efficiency

In Table 9, we study the significance of model capacity in generating high-quality images by comparing the number of parameters for models that are trained on

Model	Parameters (M)	FLOPs (G)	FID
DiT-XL/2-G [56]	675	119	2.27
SiT-XL [51]	675	119	2.06
MDT-G [23]	700	121	1.79
DiffiT	**561**	**114**	**1.73**

Table 9 – Computational efficiency comparison.

ImageNet-256 dataset. All models use the same number of function evaluations for sample generation for fair comparisons. We also use the same global window size for computing self-attention. We observe that DiffiT has 19.85%, 16.88% and 16.88% less number of parameters and 6.14%, 4.38% and 4.38% less number of FLOPs in comparison to MDT-G, SiT-XL and DiT-XL/2-G models, respectively while demonstrating a better FID score.

5.8   Effect of Classifier-Free Guidance

As shown in Fig. 7 (a), we investigate the effect of classifier-free guidance scale on the quality of generated samples in terms of FID score. For the ImageNet-256 experiment, we used the improved classifier-free guidance [23] which uses a power-cosine schedule to increase the diversity of generated images in early sampling stages. This scheme was not used for the ImageNet-512 experiment,

since it did not result in any significant improvements. The guidance scales of 4.6 and 1.49 correspond to best FID scores of 1.73 and 2.67 for ImageNet-256 and ImageNet-512 experiments, respectively. Increasing the guidance scale beyond these values results in degradation of FID score.

5.9   TMSA and DiT Modulation

We directly compare the performance of TMSA and DiT modulation mechanisms in Fig. 7 (b). For this purpose, we employ a DiT-XL as the base model with TMSA as well as its original modulation and train both models for 1000K iterations on ImageNet-256 dataset. The model with TMSA consistently shows better FID scores in different training iterations, hence validating the effectiveness of TMSA.

5.10   Effect of Window Size

As illustrated in Fig. 8, we study the impact of window size in TMSA on the FID score of generated images for models that are trained on CIFAR10 (32×32 resolution) and FFHQ-64 (64×64 resolution) datasets. Increasing the TMSA window size from 2 to 4 decreases the FID score by 23.23% and 12.17% for CIFAR10 and FFHQ-64 models, respectively. As expected, increasing the effective receptive field seems to improve the generation performance.

Fig. 8 – Impact of TMSA window size on FID.

However, increasing the window size further from 4 to 8 only results in marginal improvement of 1.53% and 3.60% for CIFAR10 and FFHQ-64, respectively. This is due to the spatial redundancy of adjacent pixels which may not contribute significantly to the generation quality upon increasing the receptive field. As also discussed in the supplementary materials, for image space experiments, we have used our window-based TMSA formulation to benefit from the efficiency gains while maintaining high image generation quality.
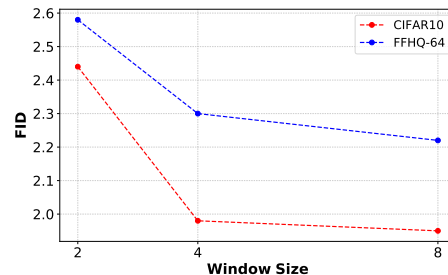
6   Conclusion

In this work, we presented DiffiT which is a novel ViT-based diffusion model for both latent and image space generation tasks. Specifically, we proposed the TMSA which allows self-attention to dynamically adapt to different stages of denoising while learning spatial and temporal dependencies and their interaction. The proposed TMSA also significantly improves the parameter efficiency. DiffiT achieves a new SOTA performance on ImageNet-256 dataset while having significantly less number of parameters in comparison to other competitive Transformer-based diffusion models such as SiT, MDT and DiT.

References

1. Albergo, M.S., Boffi, N.M., Vanden-Eijnden, E.: Stochastic interpolants: A unifying framework for flows and diffusions. arXiv preprint arXiv:2303.08797 (2023) 4
2. Albergo, M.S., Vanden-Eijnden, E.: Building normalizing flows with stochastic interpolants. arXiv preprint arXiv:2209.15571 (2022) 4
3. Anderson, B.D.: Reverse-time diffusion equation models. Stochastic Processes and their Applications **12**(3), 313–326 (1982) 5
4. Aneja, J., Schwing, A., Kautz, J., Vahdat, A.: A contrastive learning approach for training variational autoencoder priors. Advances in neural information processing systems **34**, 480–493 (2021) 4
5. Avrahami, O., Fried, O., Lischinski, D.: Blended latent diffusion. ACM Transactions on Graphics (TOG) **42**(4), 1–11 (2023) 4
6. Avrahami, O., Lischinski, D., Fried, O.: Blended diffusion for text-driven editing of natural images. In: Proc. CVPR (2022) 4
7. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (July 2016) 7
8. Balaji, Y., Nah, S., Huang, X., Vahdat, A., Song, J., Kreis, K., Aittala, M., Aila, T., Laine, S., Catanzaro, B., et al.: ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. arXiv preprint arXiv:2211.01324 (2022) 3
9. Bao, F., Li, C., Cao, Y., Zhu, J.: All are worth words: a vit backbone for score-based diffusion models. In: NeurIPS 2022 Workshop on Score-Based Methods (2022) 4, 9
10. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096 (2018) 9
11. Chang, H., Zhang, H., Jiang, L., Liu, C., Freeman, W.T.: Maskgit: Masked generative image transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11315–11325 (2022) 9
12. Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., Sutskever, I.: Generative pretraining from pixels. In: International Conference on Machine Learning. pp. 1691–1703. PMLR (2020) 4
13. Choi, J., Lee, J., Shin, C., Kim, S., Kim, H., Yoon, S.: Perception prioritized training of diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11472–11481 (2022) 2
14. Couairon, G., Verbeek, J., Schwenk, H., Cord, M.: DiffEdit: Diffusion-based semantic image editing with mask guidance. arXiv preprint arXiv:2210.11427 (2022) 4
15. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009) 1, 3, 4, 20, 23, 25, 26, 27, 28, 29, 30, 31
16. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Advances in Neural Information Processing Systems **34**, 8780–8794 (2021) 2, 9, 22
17. Ding, M., Yang, Z., Hong, W., Zheng, W., Zhou, C., Yin, D., Lin, J., Zou, X., Shao, Z., Yang, H., et al.: Cogview: Mastering text-to-image generation via transformers. Advances in Neural Information Processing Systems **34**, 19822–19835 (2021) 4
18. Ding, M., Zheng, W., Hong, W., Tang, J.: Cogview2: Faster and better text-to-image generation via hierarchical transformers. Advances in Neural Information Processing Systems **35**, 16890–16902 (2022) 4
19. Dockhorn, T., Vahdat, A., Kreis, K.: Score-Based generative modeling with Critically-Damped langevin diffusion. arXiv preprint arXiv:2112.07068 (December 2021) 5

20. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2020) 2, 11
21. Elfwing, S., Uchibe, E., Doya, K.: Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. Neural Networks **107**, 3–11 (2018) 5
22. Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A.H., Chechik, G., Cohen-Or, D.: An image is worth one word: Personalizing text-to-image generation using textual inversion. arXiv preprint arXiv:2208.01618 (2022) 4
23. Gao, S., Zhou, P., Cheng, M., Yan, S.: Masked diffusion transformer is a strong image synthesizer. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 23107–23116 (oct 2023) 2, 4, 8, 9, 13
24. Gong, X., Chang, S., Jiang, Y., Wang, Z.: Autogan: Neural architecture search for generative adversarial networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3224–3234 (2019) 9
25. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014) 4
26. Grenander, U., Miller, M.I.: Representations of knowledge in complex systems. Journal of the Royal Statistical Society: Series B (Methodological) **56**(4), 549–581 (1994) 5
27. Ho, J., Chan, W., Saharia, C., Whang, J., Gao, R., Gritsenko, A., Kingma, D.P., Poole, B., Norouzi, M., Fleet, D.J., Salimans, T.: Imagen Video: High definition video generation with diffusion models. arXiv preprint arXiv:2210.02303 (2022) 2
28. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. arXiv preprint arXiv:2006.11239 (June 2020) 2, 3, 4, 5, 22
29. Hong, W., Ding, M., Zheng, W., Liu, X., Tang, J.: Cogvideo: Large-scale pretraining for text-to-video generation via transformers. arXiv preprint arXiv:2205.15868 (2022) 4
30. Hoogeboom, E., Heek, J., Salimans, T.: simple diffusion: End-to-end diffusion for high resolution images. arXiv preprint arXiv:2301.11093 (2023) 9
31. Hudson, D.A., Zitnick, L.: Generative adversarial transformers. In: International conference on machine learning. pp. 4487–4499. PMLR (2021) 4
32. Hyvärinen, A.: Estimation of non-normalized statistical models by score matching. JMLR **6**(24), 695–709 (2005) 2
33. Jiang, Y., Chang, S., Wang, Z.: Transgan: Two transformers can make one strong gan. arXiv preprint arXiv:2102.07074 (2021) 4, 9
34. Karras, T., Aittala, M., Aila, T., Laine, S.: Elucidating the design space of diffusion-based generative models. In: Proc. NeurIPS (2022) 5, 9, 21, 22
35. Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., Aila, T.: Training generative adversarial networks with limited data. Advances in neural information processing systems **33**, 12104–12114 (2020) 9
36. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401–4410 (2019) 3, 9, 13, 20, 21, 22, 24
37. Kawar, B., Elad, M., Ermon, S., Song, J.: Denoising diffusion restoration models. arXiv preprint arXiv:2201.11793 (2022) 4
38. Kawar, B., Song, J., Ermon, S., Elad, M.: Jpeg artifact correction using denoising diffusion restoration models. arXiv preprint arXiv:2209.11888 (2022) 4

39. Kawar, B., Zada, S., Lang, O., Tov, O., Chang, H., Dekel, T., Mosseri, I., Irani, M.: Imagic: Text-based real image editing with diffusion models. arXiv preprint arXiv:2210.09276 (2022) 4

40. Kim, D., Na, B., Kwon, S.J., Lee, D., Kang, W., Moon, I.C.: Maximum likelihood training of implicit nonlinear diffusion models. arXiv preprint arXiv:2205.13699 (2022) 9

41. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (December 2014) 21

42. Kingma, D.P., Welling, M.: Auto-Encoding variational bayes. arXiv preprint arXiv:1312.6114v10 (December 2013) 4

43. Kong, Z., Ping, W., Huang, J., Zhao, K., Catanzaro, B.: Diffwave: A versatile diffusion model for audio synthesis. arXiv preprint arXiv:2009.09761 (2020) 4

44. Kreis, K., Gao, R., Vahdat, A.: CVPR tutorial on denoising diffusion-based generative modeling: Foundations and applications. https://cvpr2022-tutorial-diffusion-models.github.io/ (July 2022) 2

45. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009) 3, 9, 13, 20, 21, 22, 23

46. Lee, D., Kim, C., Kim, S., Cho, M., Han, W.S.: Autoregressive image generation using residual quantization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11523–11532 (2022) 9

47. Lee, K., Chang, H., Jiang, L., Zhang, H., Tu, Z., Liu, C.: Vitgan: Training gans with vision transformers. arXiv preprint arXiv:2107.04589 (2021) 4

48. Li, S., Chen, X., He, D., Hsieh, C.J.: Can vision transformers perform convolution? arXiv preprint arXiv:2111.01353 (2021) 4

49. Li, X.L., Thickstun, J., Gulrajani, I., Liang, P., Hashimoto, T.B.: Diffusion-lm improves controllable text generation. arXiv preprint arXiv:2205.14217 (2022) 4

50. Luhman, T., Luhman, E.: Improving diffusion model efficiency through patching. arXiv preprint arXiv:2207.04316 (2022) 4

51. 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. arXiv preprint arXiv:2401.08740 (2024) 4, 8, 9, 13

52. 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) 4

53. Nichol, A.Q., Dhariwal, P.: Improved denoising diffusion probabilistic models. In: International Conference on Machine Learning. pp. 8162–8171. PMLR (2021) 2

54. Nie, W., Guo, B., Huang, Y., Xiao, C., Vahdat, A., Anandkumar, A.: Diffusion models for adversarial purification. In: Proc. ICML (2022) 4

55. Park, J., Kim, Y.: Styleformer: Transformer based generative adversarial networks with style vector. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8983–8992 (2022) 4

56. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4195–4205 (2023) 2, 4, 7, 8, 9, 13, 20

57. Perez, E., Strub, F., De Vries, H., Dumoulin, V., Courville, A.: Film: Visual reasoning with a general conditioning layer. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 32 (2018) 2

58. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with CLIP latents. arXiv preprint arXiv:2204.06125 (2022) 2, 3

59. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10684–10695 (2022) 2, 7, 9, 20

60. Rombach, R., Esser, P.: Stable diffusion v1-4. https://huggingface.co/CompVis/stable-diffusion-v1-4 (July 2022) 2

61. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. arXiv preprint arXiv:1505.04597 (May 2015) 2

62. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation. arXiv preprint arXiv:2208.12242 (2022) 4

63. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S.K.S., Ayan, B.K., Mahdavi, S.S., Lopes, R.G., et al.: Photorealistic text-to-image diffusion models with deep language understanding. arXiv preprint arXiv:2205.11487 (2022) 3

64. Salimans, T., Ho, J.: Progressive distillation for fast sampling of diffusion models. arXiv preprint arXiv:2202.00512 (February 2022) 5

65. Sauer, A., Schwarz, K., Geiger, A.: Stylegan-xl: Scaling stylegan to large diverse datasets. In: ACM SIGGRAPH 2022 conference proceedings. pp. 1–10 (2022) 8, 9

66. Shaw, P., Uszkoreit, J., Vaswani, A.: Self-attention with relative position representations. arXiv preprint arXiv:1803.02155 (2018) 6

67. Sinha, A., Song, J., Meng, C., Ermon, S.: D2c: Diffusion-decoding models for few-shot conditional generation. Advances in Neural Information Processing Systems **34**, 12533–12548 (2021) 4

68. Sohl-Dickstein, J., Weiss, E.A., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. arXiv preprint arXiv:1503.03585 (March 2015) 2, 3, 4

69. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. In: International Conference on Learning Representations (2021) 4, 5

70. Song, Y., Ermon, S.: Generative modeling by estimating gradients of the data distribution. arXiv preprint arXiv:1907.05600 (July 2019) 2

71. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: International Conference on Learning Representations (2021) 2, 3, 4, 5, 9, 12

72. Tashiro, Y., Song, J., Song, Y., Ermon, S.: Csdi: Conditional score-based diffusion models for probabilistic time series imputation. Advances in Neural Information Processing Systems **34**, 24804–24816 (2021) 4

73. Vahdat, A., Kautz, J.: Nvae: A deep hierarchical variational autoencoder. Advances in neural information processing systems **33**, 19667–19679 (2020) 9

74. Vahdat, A., Kreis, K., Kautz, J.: Score-based generative modeling in latent space. arXiv preprint arXiv:2106.05931 (June 2021) 4, 7, 9

75. Valevski, D., Kalman, M., Matias, Y., Leviathan, Y.: UniTune: Text-driven image editing by fine tuning an image generation model on a single image. arXiv preprint arXiv:2210.09477 (2022) 4

76. Vincent, P.: A connection between score matching and denoising autoencoders. Neural computation **23**(7), 1661–1674 (July 2011) 2, 5

77. Wang, S., Li, B., Khabsa, M., Fang, H., Ma, H.: Linformer: Self-attention with linear complexity. arXiv preprint arXiv:2006.04768 (2020) 4

78. Wu, Y., He, K.: Group normalization. arXiv preprint arXiv:1803.08494 (March 2018) 8

79. Xu, M., Yu, L., Song, Y., Shi, C., Ermon, S., Tang, J.: GeoDiff: A geometric diffusion model for molecular conformation generation. In: Proc. ICLR (2022) 4
80. Xu, R., Xu, X., Chen, K., Zhou, B., Loy, C.C.: Stransgan: An empirical study on transformer in gans. arXiv preprint arXiv:2110.13107 (2021) 4
81. Yang, X., Shih, S.M., Fu, Y., Zhao, X., Ji, S.: Your vit is secretly a hybrid discriminative-generative diffusion model. arXiv preprint arXiv:2208.07791 (2022) 9
82. Ye, M., Wu, L., Liu, Q.: First hitting diffusion models. arXiv preprint arXiv:2209.01170 (2022) 4
83. Zeng, X., Vahdat, A., Williams, F., Gojcic, Z., Litany, O., Fidler, S., Kreis, K.: Lion: Latent point diffusion models for 3d shape generation. In: Advances in Neural Information Processing Systems (NeurIPS) (2022) 4
84. Zhang, B., Gu, S., Zhang, B., Bao, J., Chen, D., Wen, F., Wang, Y., Guo, B.: Styleswin: Transformer-based gan for high-resolution image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11304–11314 (2022) 4
85. Zhang, H., Yin, W., Fang, Y., Li, L., Duan, B., Wu, Z., Sun, Y., Tian, H., Wu, H., Wang, H.: Ernie-vilg: Unified generative pre-training for bidirectional vision-language generation. arXiv preprint arXiv:2112.15283 (2021) 4
86. Zhang, Q., Tao, M., Chen, Y.: gddim: Generalized denoising diffusion implicit models. arXiv preprint arXiv:2206.05564 (June 2022) 5
87. Zhao, L., Zhang, Z., Chen, T., Metaxas, D., Zhang, H.: Improved transformer for high-resolution gans. Advances in Neural Information Processing Systems **34**, 18367–18380 (2021) 4
88. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6881–6890 (2021) 11
89. Zhou, L., Du, Y., Wu, J.: 3D shape generation and completion through Point-Voxel diffusion. arXiv preprint arXiv:2104.03670 (April 2021) 4

Appendix

G   Ablation

G.1   Comparison to DiT and LDM

On contrary to LDM [59] and DiT [56], the latent DiffiT does not rely on shift and scale, as in AdaLN [56], or concatenation to incorporate time embedding into the denoising networks. However, DiffiT uses a time-dependent self-attention (*i.e.* TMSA) to jointly learn the spatial and temporal dependencies. In addition, DiffiT proposes both image and latent space models for different image generation tasks with different resolutions with SOTA performance. Specifically, as shown in Table S.1, DiffiT significantly outperforms LDM [59] and DiT [56] by 31.26% and 51.94% in terms of FID score on ImageNet-256 [15] dataset. In addition, DiffiT outperforms DiT [56] by 13.85% on ImageNet-512 [15] dataset. Hence, these benchmarks validate the effectiveness of the proposes architecture and TMSA design in DiffiT model as opposed to previous SOTA for both CNN and Transformer-based diffusion models.

Model	Class	ImageNet-256				ImageNet-512			
		FID ↓	IS ↑	Precision ↑	Recall ↑	FID ↓	IS ↑	Precision ↑	Recall ↑
LDM-4-G [59]	Diffusion	3.60	247.67	0.87	0.48	-	-	-	-
DiT-XL/2-G [56]	Diffusion	2.27	278.24	0.83	0.57	3.04	240.82	0.84	0.54
DiffiT	Diffusion	**1.73**	276.49	0.80	0.62	**2.67**	252.12	0.83	0.55

Table S.1 – Comparison of image generation performance against state-of-the-art models on ImageNet-256 and ImageNet-512 dataset. The latent DiffiT model achieves SOTA performance in terms of FID score on ImageNet-256 dataset.

H   Architecture

H.1   Image Space

We provide the details of blocks and their corresponding output sizes for both the encoder and decoder of the DiffiT model in Table S.2 and Table S.3, respectively. The presented architecture details denote models that are trained with 64×64 resolution. Without loss of generality, the architecture can be extended for 32×32 resolution. For FFHQ-64 [36] dataset, the values of L_1, L_2, L_3 and L_4 are 4, 4, 4, and 4 respectively. For CIFAR-10 [45] dataset, the architecture spans across three different resolution levels (*i.e.* 32, 16, 8), and the values of L_1, L_2, L_3 are 4, 4, 4 respectively. Please refer to the paper for more information regarding the architecture details.

Table S.2 – Detailed description of components in DiffiT encoder for models that are trained at 64×64 resolution.

Component Description	Output size
Input	$64 \times 64 \times 3$
Tokenizer	$64 \times 64 \times 128$
DiffiT ResBlock $\times L_1$	$64 \times 64 \times 128$
Downsampler	$32 \times 32 \times 128$
DiffiT ResBlock $\times L_2$	$32 \times 32 \times 256$
Downsampler	$16 \times 16 \times 128$
DiffiT ResBlock $\times L_3$	$16 \times 16 \times 256$
Downsampler	$8 \times 8 \times 256$
DiffiT ResBlock $\times L_4$	$8 \times 8 \times 256$

Table S.3 – Detailed description of components in DiffiT decoder for models that are trained at 64×64 resolution.

Component Description	Output size
Input	$8 \times 8 \times 256$
Upsampler	$16 \times 16 \times 256$
DiffiT ResBlock $\times L_3$	$16 \times 16 \times 256$
Upsampler	$32 \times 32 \times 256$
DiffiT ResBlock $\times L_2$	$32 \times 32 \times 256$
Upsampler	$64 \times 64 \times 256$
DiffiT ResBlock $\times L_1$	$64 \times 64 \times 128$
Head	$64 \times 64 \times 3$

H.2   Latent Space

In Fig S.1, we illustrate the architecture of the latent DiffiT model. Our model is comparable to DiT-XL/2-G variant which 032 uses a patch size of 2. Specifically, we use a depth of 30 layers with hidden size dimension of 1152, number of heads dimension of 16 and MLP ratio of 4. In addition, for the classifier-free guidance implementation, we only apply the guidance to the first three input channels with a scale of $(1 + \mathbf{x})$ where $\mathbf{x}$ is the input latent.

I   Implementation Details

I.1   Image Space

We strictly followed the training configurations and data augmentation strategies of the EDM [34] model for the experiments on CIFAR10 [45], and FFHQ-64 [36] datasets, all in an unconditional setting. All the experiments were trained for 200000 iterations with Adam optimizer [41] and used PyTorch framework and 8 NVIDIA A100 GPUs. We used batch sizes of 512 and 256, learning rates of
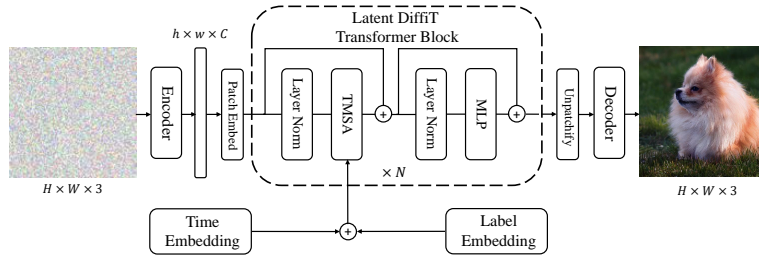
Fig. S.1 – Overview of the latent DiffiT framework.

1×10^{-3} and 2×10^{-4} and training images of sizes 32×32 and 64×64 on experiments for CIFAR10 [45] and FFHQ-64 [36] datasets, respectively.

We use the deterministic sampler of EDM [34] model with 18, 40 and 40 steps for CIFAR-10 and FFHQ-64 datasets, respectively. For FFHQ-64 dataset, our DiffiT network spans across 4 different stages with 1, 2, 2, 2 blocks at each stage. We also use window-based attention TMSA with local window size of 8 at each stage. For CIFAR-10 dataset, the DiffiT network has 3 stages with 2 blocks at each stage. Similarly, we compute attentions on local windows with size 4 at each stage. Note that for all networks, the resolution is decreased by a factor of 2 in between stages. However, except for when transitioning from the first to second stage, we keep the number of channels constant in the rest of the stages to maintain both the number of parameters and latency in our network. Furthermore, we employ traditional convolutional-based downsampling and upsampling layers for transitioning into lower or higher resolutions. We achieved similar image generation performance by using bilinear interpolation for feature resizing instead of convolution. For fair comparison, in all of our experiments, we used the FID score which is computed on 50K samples and using the training set as the reference set.

I.2   Latent Space

We employ learning rates of 3×10^{-4} and 1×10^{-4} and batch sizes of 256 and 512 for ImageNet-256 and ImageNet-512 experiments, respectively. We also use the exponential moving average (EMA) of weights using a decay of 0.9999 for both experiments. We also use the same diffusion hyper-parameters as in the ADM [16] model. For a fair comparison, we use the DDPM [28] sampler with 250 steps and report FID-50K for both ImageNet-256 and ImageNet-512 experiments.

J   Qualitative Results

We illustrate visualization of generated images for CIFAR-10 [45] and FFHQ-64 [36] datasets in Figures S.2 and S.3, respectively. In addition, in Figures S.4, S.5, S.6 and S.7, we visualize the the generated images by the latent DiffiT model

for ImageNet-512 [15] dataset. Similarly, the generated images for ImageNet-256 [15] are shown in Figures S.8, S.9 and S.10. We observe that the proposed DiffiT model is capable of capturing fine-grained details and produce high fidelity images across these datasets.

Fig. S.2 – Visualization of uncurated generated images for CIFAR-10 [45] dataset. Best viewed in color.
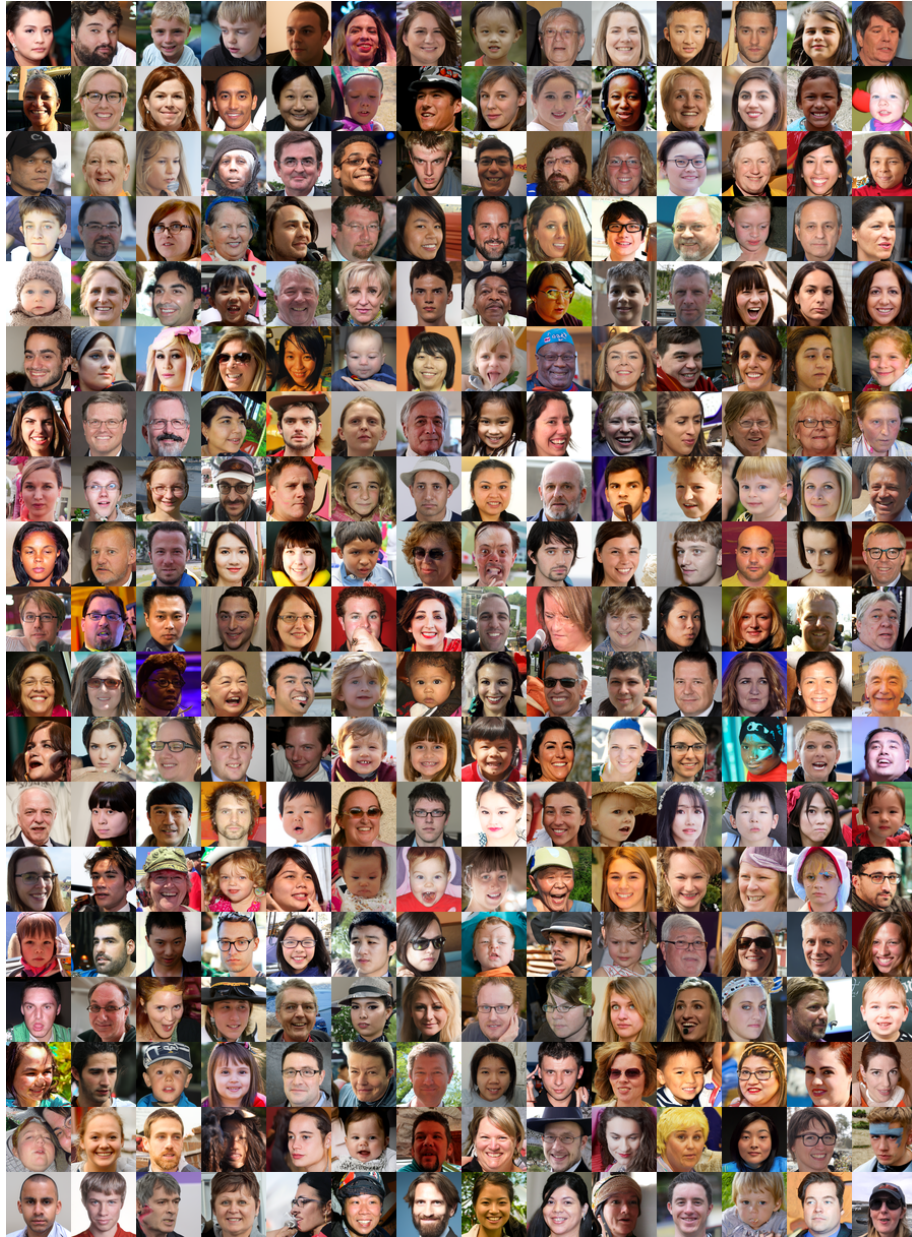
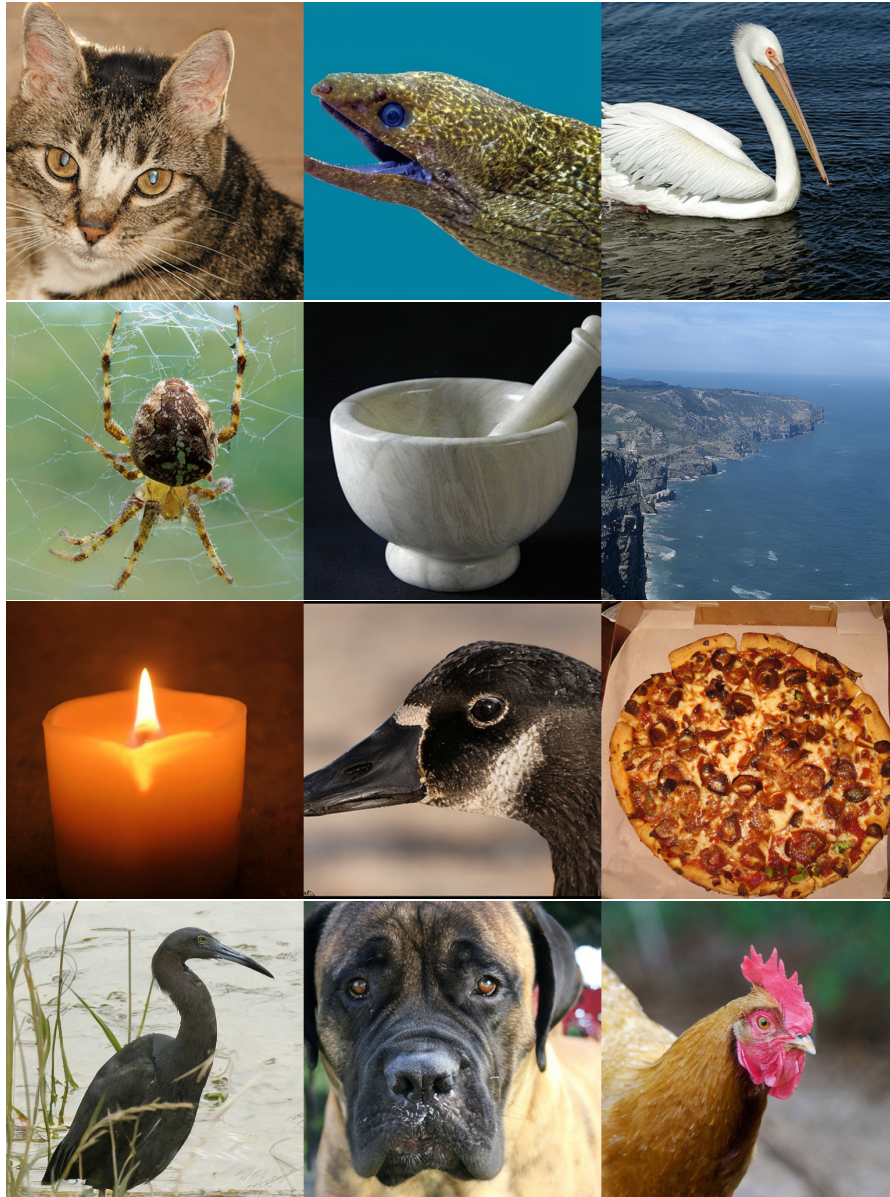**Fig. S.3** – Visualization of uncurated generated images for FFHQ-64 [36] dataset. Best viewed in color.

Fig. S.4 – Visualization of uncurated generated 512×512 images on ImageNet [15] dataset by latent DiffiT model. Images are randomly sampled. Best viewed in color.

Fig. S.5 – Visualization of uncurated generated 512×512 images on ImageNet [15] dataset by latent DiffiT model. Images are randomly sampled. Best viewed in color.
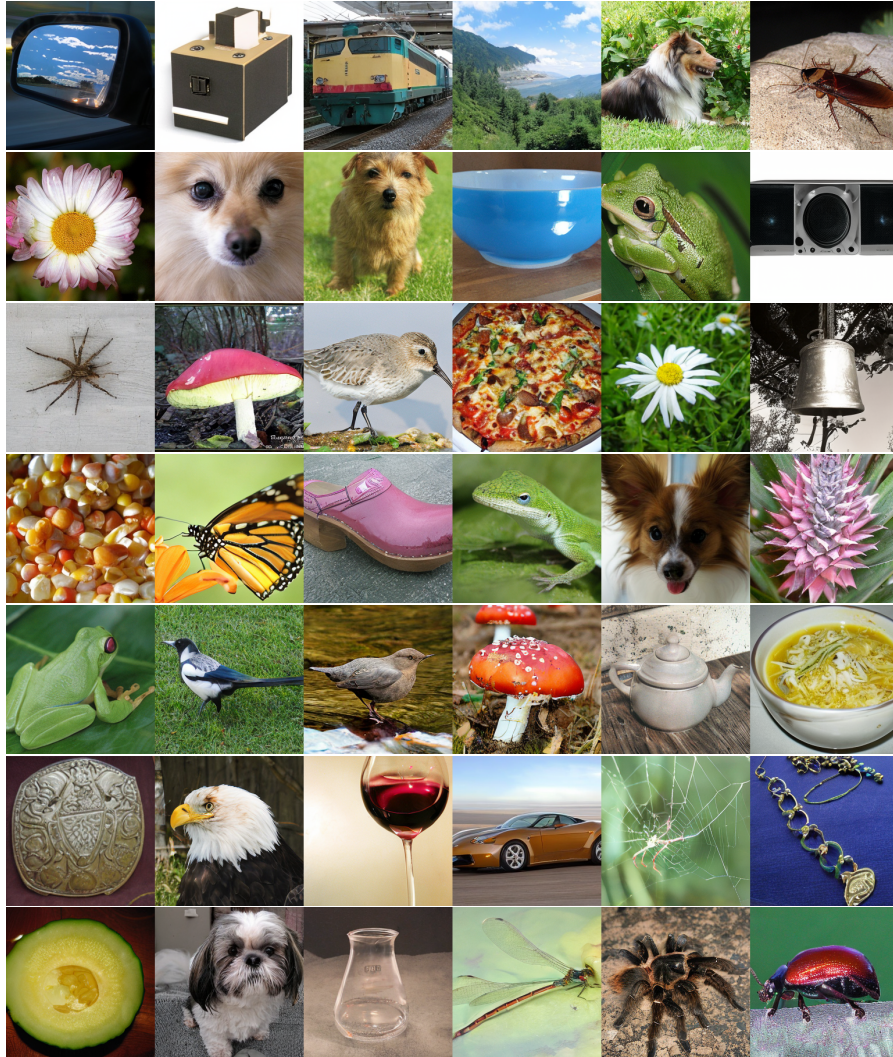
Fig. S.6 – Visualization of uncurated generated 512×512 images on ImageNet [15] dataset by latent DiffiT model. Images are randomly sampled. Best viewed in color.

Fig. S.7 – Visualization of uncurated generated 512×512 images on ImageNet [15] dataset by latent DiffiT model. Images are randomly sampled. Best viewed in color.

Fig. S.8 – Visualization of uncurated generated 256×256 images on ImageNet [15] dataset by latent DiffiT model. Images are randomly sampled. Best viewed in color.
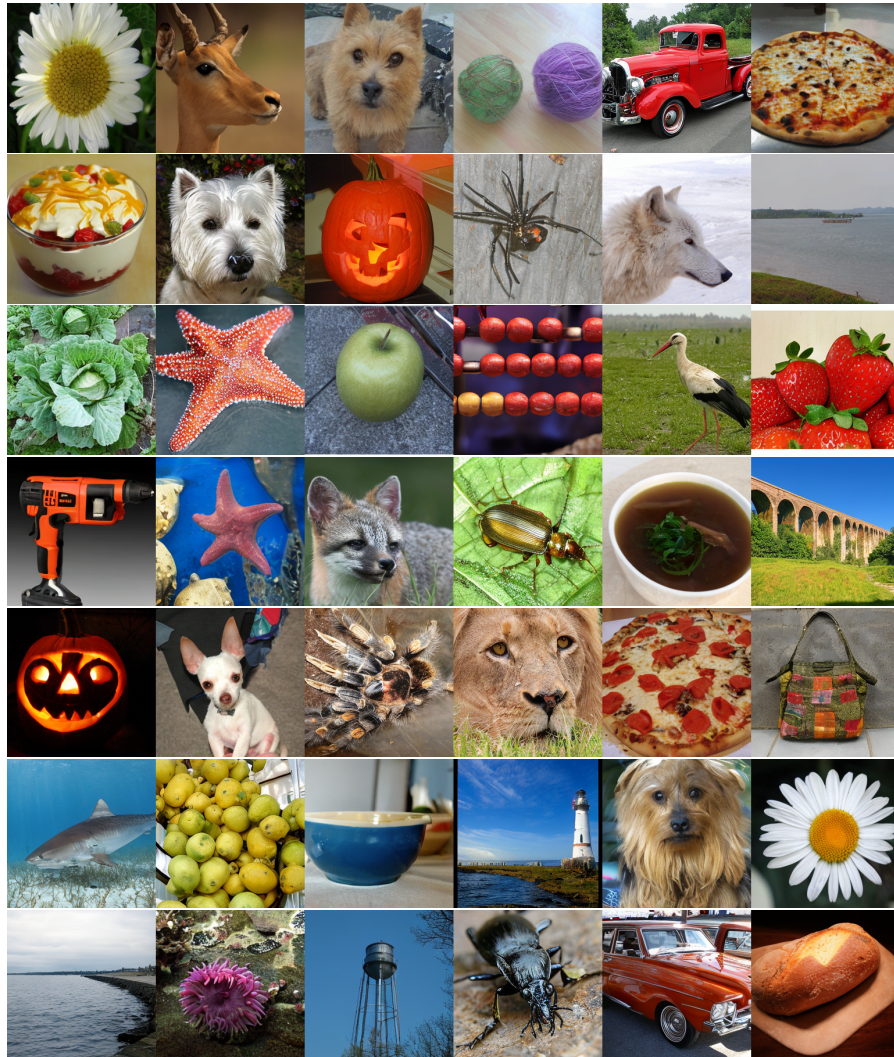
Fig. S.9 – Visualization of uncurated generated 256×256 images on ImageNet [15] dataset by latent DiffiT model. Images are randomly sampled. Best viewed in color.
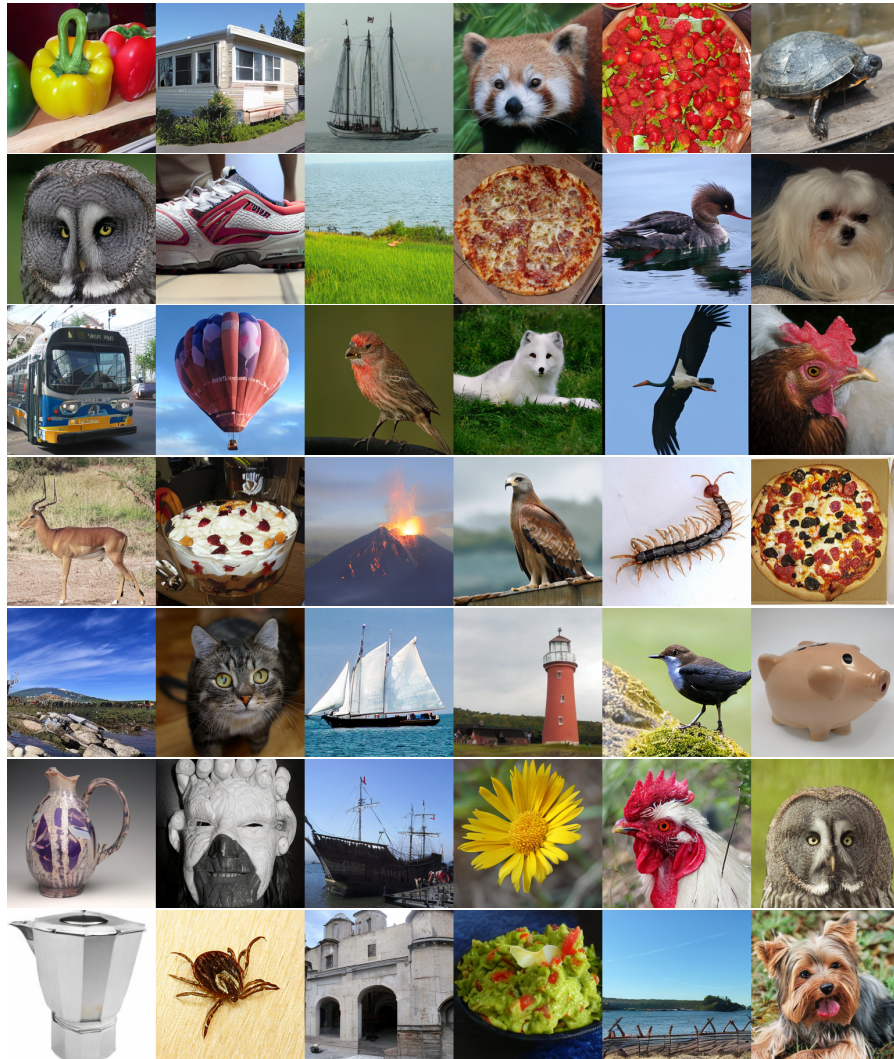
Fig. S.10 – Visualization of uncurated generated 256×256 images on ImageNet [15] dataset by latent DiffiT model. Images are randomly sampled. Best viewed in color.