LaMamba-Diff: Linear-Time High-Fidelity Diffusion Models Based on Local Attention and Mamba

Yunxiang Fu[1]
yunxiang@connect.hku.hk
Chaoqi Chen[2]
cqchen1994@gmail.com
Yizhou Yu[1]
yizhouy@acm.org

[1] School of Computing and Data Science, The University of Hong Kong
[2] College of Computer Science and Software Engineering, Shenzhen University

Abstract

Recent Transformer-based diffusion models have shown remarkable performance, largely attributed to the ability of the self-attention mechanism to accurately capture both global and local contexts by computing all-pair interactions among input tokens. However, their quadratic complexity poses significant computational challenges for long-sequence inputs. Conversely, a recent state space model called Mamba offers linear complexity by compressing a filtered global context into a hidden state. Despite its efficiency, compression inevitably leads to information loss of fine-grained local dependencies among tokens, which are crucial for effective visual generative modeling. Motivated by these observations, we introduce Local Attentional Mamba (LaMamba) blocks that combine the strengths of self-attention and Mamba, capturing both global contexts and local details with linear complexity. Leveraging the efficient U-Net architecture, our model exhibits exceptional scalability and surpasses the performance of DiT across various model scales on ImageNet at 256x256 resolution, all while utilizing substantially fewer GFLOPs and a comparable number of parameters. Compared to state-of-the-art diffusion models on ImageNet 256x256 and 512x512, our largest model presents notable advantages, such as a reduction of up to 62% GFLOPs compared to DiT-XL/2, while achieving competitive performance with comparable or fewer parameters.

1 Introduction

Diffusion models have made significant strides in the field of generative modeling [13, 38], especially for images [5, 25, 29, 30, 31, 34], often surpassing the previously popular generative adversarial networks (GANs) [8]. The advancements of diffusion models have been driven by many orthogonal factors, including sampling approaches [12, 13, 17, 37], latent space modeling [31], and backbone architecture designs [27, 42]. Despite recent attempts to devise innovative backbones, state-of-the-art backbones for diffusion models [16, 27, 31] largely rely on self-attention [41] for high-fidelity generation. The effectiveness of self-attention stems from its capacity to accurately capture global contexts and fine-grained local

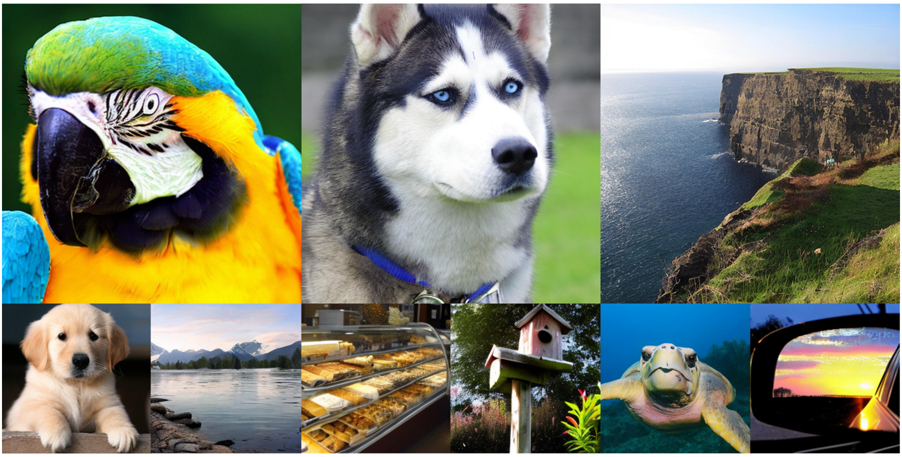

Figure 1: Sample images generated by our model trained on ImageNet at 512×512 and 256×256 resolutions.

dependencies from inputs by explicitly computing all-pair interactions among input tokens. However, their quadratic computational complexity with respect to the input sequence length poses computational challenges when dealing with long sequences, such as high-resolution images or sequences formed with a small patch size.

To address the quadratic complexity of self-attention, a recent state space model (SSM) named Mamba [9] has been proposed. Using dynamic weights, Mamba captures global contexts with linear time by compressing contextual information of input tokens into a hidden state. Demonstrating a strong potential in long-sequence modeling, Mamba has been applied across diverse domains, including medical data [19, 33], point clouds [20], and vision tasks [21, 45]. However, unlike self-attention, the compression and selective process of Mamba does not explicitly compute pairwise interactions. Consequently, part of fine-grained local information is lost, leading to sub-optimal performance. The significance of local details for generative modeling is demonstrated by the detrimental decline in the performance of DiT [27] as the patch size increases, attributed to the loss of fine-grained information within each patch [10].

In light of these observations, we introduce a novel Local Attentional Mamba (LaMamba) block that combines the strengths of Transformers and Mamba to accurately model global contexts and local details with linear complexity. LaMamba captures global contexts efficiently using Mamba while accurately preserving fine-grained local dependencies using local self-attention. Our local self-attention has linear complexity by computing pairwise interactions within a context window with a fixed size. LaMamba brings the best of two worlds for visual generative modeling, offering favorable properties like scalability, robustness, and efficiency.

Based on LaMamba blocks, we design LaMamba-Diff, a novel backbone for diffusion models that adopts a U-Net architecture [32]. LaMamba-Diff naturally constructs multi-scale hierarchical features through down and up-sampling and exhibits efficiency by compressing spatial dimensions during the downsampling phase. The highly efficient design of LaMamba and LaMamba-Diff allows us to utilize 1×1 patches, which enables more accurate modeling of fine-grained spatial dependencies by preventing the loss of local details within each

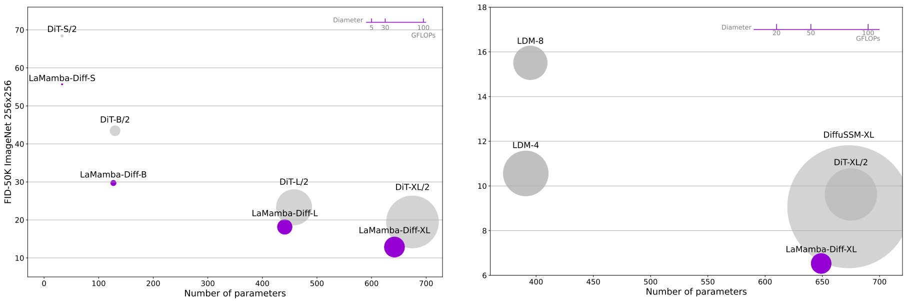

Figure 2: Unconditional image generation quality on ImageNet 256x256. The area of bubbles denote GFLOPs. *Left*: FID-50K of LaMamba-Diff models trained for 400k iterations. Performance improves with the number of parameters and GFLOPs. *Right*: Our largest model outperforms state-of-the-art diffusion models with substantially fewer GFLOPs.

patch [10]. In particular, with a comparable number of parameters, LaMamba-Diff with a patch size of 1 uses significantly fewer GLFOPs compared to DiT [27] with a patch size of 2, despite that DiT has a 4x shorter input sequence length.

We evaluate the performance of LaMamba-Diff on the widely-used ImageNet dataset for image generation. As illustrated in Fig. 2 (right), LaMamba-Diff achieves state-of-the-art Fréchet Inception Distance (FID) with comparable or fewer parameters and significantly less GFLOPs on ImageNet at 256×256 resolution. Moreover, as depicted in Fig. 2 (left), LaMamba-Diff exhibits excellent scalability with FID scores consistently decreasing with the number of parameters and GFLOPs. In the class-conditional ImageNet 256x256 image generation benchmark, LaMamba-Diff-XL achieves a state-of-the-art FID of 2.04 using 57.6% fewer GFLOPs compared to DiT-XL/2. For class-conditioned ImageNet 512x512 image generation, LaMamba-Diff-XL achieves an FID of 3.01 using 61.6% fewer GFLOPs compared to DiT-XL/2. These results demonstrate the effectiveness and efficiency of LaMamba-Diff for high-resolution image generation tasks.

To summarise, our contributions in this paper are three-fold:

- We design a novel LaMamba block that combines the strengths of Transformers and Mamba, accurately capturing global contexts and local details with linear complexity.
- We propose LaMamba-Diff, a highly efficient backbone for diffusion models. It allows a patch size of 1 without substantial computational overhead, which was previously unavailable for diffusion backbones.
- Experimental results demonstrate that LaMamba-Diff achieves competitive FID using significantly fewer GFLOPs. Additionally, LaMamba-Diff demonstrates excellent scalability.

2 Related works

Diffusion model backbones. Diffusion models (DM) belong to a class of probabilistic generative models that iteratively corrupt data by introducing noise through a forward process, and subsequently learn to reverse this process for sample generation [13, 36, 39]. Recently,

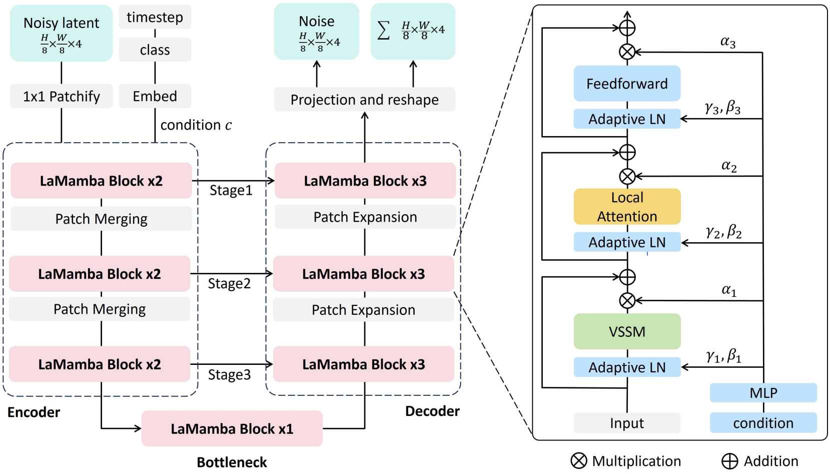

Figure 3: Network architecture of LaMamba-Diff. *Left:* Architecture of LaMamba-Diff-S. *Right:* Local attentional Mamba block.

DM has become the de facto choice for image generation, due to its capability to generate photo-realistic images and stable training property [3, 29, 29, 31, 31, 34, 44], which was unavailable for the previous state-of-the-art GANs [8]. The backbones for state-of-the-art DM are UNet-based [13, 31] or Transformer-based [27] and rely heavily on attention [41]. However, the quadratic complexity of attention hinders the application of diffusion models to long sequential data. Recently, to achieve linear complexity, state space models have been leveraged to construct backbones for DM [7, 14, 42] by replacing the self-attention module in existing backbones like DiT [27]. However, they fail to outperform DiT using a comparable number of parameters and GFLOPs. In contrast, we propose a hybrid Mamba backbone using local attention, achieving linear complexity without sacrificing image fidelity and outperforming DiT using fewer GFLOPs and parameters.

State space models. Dynamic state space models (Mamba) [9] has gained popularity in sequence modeling renowned for its linear complexity, dynamic weights, and global receptive field. Mamba has been widely explored in various domains, including vision backbones [21, 45], medical imaging [19, 33], 3D point clouds [20], tabular data [1], and image/video generation [7, 14, 26, 42]. In this paper, we investigate the use of Mamba for image generation, distinguishing ourselves from previous works by utilizing local attention to address the loss of fine-grained local details when replacing attention with Mamba.

3 Methodology

We present LaMamba-Diff, a linear-time hybrid U-Net architecture combining dynamic state space models (Mamba) and local self-attention through novel Local Attentional Mamba (LaMamba) blocks. As shown in Fig.3, LaMamba blocks synergistically integrate global context modeling via efficient state space transitions (O(L) complexity) with precise local dependency capture through constrained self-attention.

3.1 LaMamba Blocks

Overview. Transformers have excelled in latent diffusion models [27, 31] due to their self-attention mechanism, which captures detailed contextual information from all input token pairs without compression. However, the quadratic complexity of self-attention with respect to the length of the input sequence poses scalability challenges for long input sequences arising from high-resolution scenarios or small patch sizes. The selective SSM in Mamba [9] addresses this by compressing contextual information into a single hidden state, achieving linear-time complexity. Nevertheless, compression inevitably leads to the loss of fine-grained local information, which is crucial for diffusion models. When increasing the patch size in DiT architectures, the Fréchet Inception Distance (FID) increases significantly [27], directly attributed to the loss of high-frequency spatial information within enlarged patches. This limitation is evident since SSM-based diffusion models do not outperform DiT using a comparable number of parameters and GFLOPs [7, 27, 42]. To combine the strengths of self-attention and selective SSM, we introduce Local Attentional Mamba (LaMamba) blocks, which capture global contexts and local details in linear time.

The proposed LaMamba block, depicted in Fig.3 (right), comprises three key components: a Visual State Space Module (VSSM), a local attention module (LA), and a feedforward network (FFN). Each component includes a residual connection and integrates conditioning information via adaptive layer normalization (AdaLN) with zero initialization [5, 27]. Specifically, $\mathrm{AdaLN}(h,c) = \gamma \mathrm{LN}(h) + \beta$, where γ and β are computed by passing the conditioning embedding c to an MLP. At the start of training, the MLP parameters are initialized to zero, making α a zero vector and LaMamba blocks as identity functions since the output of each component is equal to its residual connection. VSSM incorporates the selective SSM introduced in Mamba [9] to efficiently capture global contexts. We take advantage of existing vision variants of the selective SSMs designed for 2D visual data, such as 2D Selective Scan (SS2D) in VMamba [21]. Local attention explicitly computes local self-attention among input tokens, resulting in fine-grained local representations. The standard FFN with a hidden dimension expansion ratio of 4 is added at the end of the block [6].

Main components. The selective SSM in Mamba, originally designed for sequence modeling in natural language processing, processes 1D input sequences in a causal manner. Directly applying SSM to non-causal 2D visual signals can lead to suboptimal results, as it does not account for the 2D spatial information critical for vision tasks [21, 45]. Hence, we utilize a visual state space module (VSSM) that explicitly models 2D spatial information. We compare four variants of VSSM: Bidirectional SSM in ViM [45], SSM in LocalVMamba [15], Efficient Scan 2D in EfficientVMamba [28], and the SS2D block in VMamba [21]. A comparison of different VSSMs can be found in the Supplementary Material Table 5. We adopt SS2D blocks [21], and modify the scanning trajectories in SS2D to continuous spatial scans [43]. Unless otherwise stated, all models in this work use our modified SS2D block as the VSSM owing to its superior performance.

The compression and selection mechanism in VSSM serves to filter out irrelevant information for improved efficiency [9] but may lose information about fine-grained local details and dependencies. To address this, we utilize local attention to recover local details. In practice, we utilize window-based self-attention [22], which partitions the input into non-overlapping square windows and applies self-attention within each window This approach has linear complexity as attention is applied to fixed-size windows.

Summary. The novelty of LaMamba lies in its hybrid composition. To the best of our knowledge, this is the first attempt to incorporate two complementary components, selective

SSM and local self-attention, within the building block of a diffusion model backbone, resulting in the capability to capture both global contexts and fine-grained local details with linear time complexity. LaMamba naturally inherits the strengths of both Mamba and self-attention, including scalability and robustness. This is demonstrated through its state-of-the-art performance and efficiency, as shown in Table 1 and Figure 2, respectively. Moreover, ablation studies in Table 3 empirically confirm the effectiveness of local attention in learning fine-grained local representations.

3.2 Network Architecture

While LaMamba blocks can be applied to isotropic architectures like DiT [27], we adopt the U-Net architecture [32] to leverage hierarchical features and improve efficiency. Figure 3 (left) illustrates the architecture of LaMamba-Diff-S. We follow LDM [31] to train our model in the latent space of the pre-trained VAE with a down-sampling factor of 8.

Input tokens. The inputs consists of timestep t, class label l, and noisy latent $z_t \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times 4}$, which has the same dimensions as the latent of the pre-trained VAE. By viewing z_t as a $\frac{H}{8} \times \frac{W}{8}$ grid of features, we treat every position on the grid as a patch (1x1 patches) and linearly embed every patch without flattening. This encodes z_t into a latent representation of size $\frac{H}{8} \times \frac{W}{8} \times D$, where D is the hidden dimension. Meanwhile, t and l are encoded into continuous representations using an MLP and a learnable embedding dictionary, respectively; the sum of these two encoding results becomes the conditioning embedding c.

4 Experiments

4.1 Setting

Latent Diffusion Model. LaMamba-Diff operates in the latent space of a pre-trained VAE [1] that has a downsampling factor of 8. For instance, an RGB image x_0 of size $256 \times 256 \times 3$ would be encoded into a compressed latent $z_0 = \mathcal{E}(x_0)$ of size $32 \times 32 \times 4$. We train our model to learn the reverse diffusion process in the latent space of the VAE using diffusion hyperparameters from ADM [5]. Specifically, we use a linear variance scheduler and ADM's parameterization of covariance Σ_θ. Specifically, we follow their embedding approach to encode timestep and class label conditions, and use a linear variance scheduler and ADM's parameterization of covariance Σ_θ. Images are generated by sampling novel latents z and passing them to the VAE decoder $x = \mathcal{D}(z)$.

Training Details. We follow the training and hyperprameter settings in [27] to train variants of LaMamba-Diff on the ImageNet-1k dataset [4] at 256×256 and 512×512 resolutions using classifier free guidance [12]. Specifically, we employ the AdamW optimizer [23] with a constant learning rate of 1e-4, a global batch size of 256, and no weight decay. We follow the setting in [21] for the VSSM hyperparameters. All experiments were conducted on 8 NVIDIA H800 GPUs.

Evaluation. We measure performance with Fréchet Inception Distance (FID) [11], a standard metric for assessing the quality of generated images. For a fair comparison with prior works, we use 250 DDPM sampling steps to sample 50K images to compute FID [27] using the ADM's TensorFlow implementation [5]. We also report sFID [24], Inception Score [35], and Precision, and Recall [18] as secondary metrics.

[1] https://huggingface.co/stabilityai/sd-vae-ft-mse

Model	Parameters(M)	GFLOPs	Training Steps(M)	FID↓	sFID↓	IS↑	Precision↑	Recall↑
ImageNet 256×256								
				Unconditional				
ADM	554	1120	2.0	10.94	6.02	100.98	0.69	0.63
ADM-U	608	742	2.0	7.49	5.13	127.49	0.72	0.63
LDM-8	395	79	1.2	15.51	-	79.03	0.65	0.63
LDM-4	391	103	0.8	10.56	-	103.49	**0.71**	0.62
DiT-XL/2	675	118	7.0	9.62	6.85	121.50	0.67	**0.67**
DiffuSSM-XL	673	280	2.6	9.07	**5.52**	118.32	0.69	0.64
LaMamba-Diff-XL	656	50	2.0	**6.12**	5.58	**149.13**	**0.71**	0.64
				Classifier-free guidance				
ADM-G, ADM-U	673	761	2.0	3.94	6.14	215.84	0.83	0.53
LDM-8-G	395	79	1.2	7.76	-	209.52	0.84	0.35
LDM-4-G	391	103	0.8	3.60	-	247.67	**0.87**	0.48
U-ViT-H/2-G	501	133	2.0	2.29	-	247.67	**0.87**	0.48
DiffuSSM-XL-G	673	280	2.6	2.28	4.49	259.13	0.86	0.56
DiT-XL/2-G	675	118	7.0	2.27	4.60	278.24	0.83	**0.57**
DiS-H/2-G [7]	900	217	2.0	2.10	4.55	271.32	0.82	0.58
DiM-Huge	860	210	1.9	2.21	-	-	-	-
LaMamba-Diff-XL-G	656	50	2.0	**2.04**	**4.52**	**296.07**	0.84	0.56
ImageNet 512×512								
				Unconditional				
DiT-XL/2	675	524	3.0	12.03	7.12	105.25	0.75	**0.64**
LaMamba-Diff-XL	656	201	3.0	**7.76**	**6.89**	**122.41**	**0.81**	0.60
				Classifier-free guidance				
ADM-G, ADM-U	774	2834	1.9	3.85	5.86	221.72	0.84	0.53
U-ViT-H/4-G	501	133	2.0	4.05	8.44	261.13	0.84	0.48
DiffuSSM-XL-G	673	1066	1.2	3.41	5.84	255.06	0.85	0.49
DiT-XL/2-G	675	524	3.0	3.04	**5.02**	240.82	0.84	**0.54**
DiS-H/2-G [7]	900		4.0	2.88	4.74	272.33	0.84	0.56
DiM-Huge	860	708	-	3.78	-	-	-	-
LaMamba-Diff-XL-G	656	201	3.0	**3.01**	5.15	**277.19**	0.85	0.52

Table 1: **Quantitative comparison of image generation quality on ImageNet 256x256 and 512x512.** We denote classifier-free guidance by appending "-G" to model names. LaMamba-Diff-XL achieves competitive performance with substantially fewer GFLOPs. The training steps denote the number of iterations trained using a batch size of 256.

4.2 Main Results

ImageNet 256x256. We compare our largest model, LaMamba-Diff-XL,with state-of-the-art (SOTA) diffusion models on ImageNet at 256x256 resolution. Table 1 reports quantitative results, while qualitative examples are provided in the Supplementary Material Figures 4-9. In both unconditional and class-conditional settings, LaMamba-Diff-XL outperforms attention-only (DiT [27],U-ViT [2]) and SSM-only (DiffuSSM [42], DiS-H/2 [7], DiM [40]) methods, achieving SOTA FID scores while requiring substantially fewer GFLOPs. In unconditional image generation, LaMamba-Diff-XL achieves a SOTA FID of 6.12, improving the previous SOTA FID (9.07) by 2.95 (33%) with 82% fewer GFLOPs and 17M fewer parameters. In conditional image generation with classifier-free guidance (CFG) [12], LaMamba-Diff-XL achieves a SOTA FID of 2.04 using 57.6% fewer GFLOPs and 19M fewer parameters compared to DiT-XL/2. Moreover, LaMamba-Diff-XL achieves the highest Inception Score (IS) of 296.07.

ImageNet 512x512. We evaluate the image generation quality of LaMamba-Diff-XL on ImageNet at 512x512 resolution. Here, LaMamba-Diff-XL processes an input latent of size 64x64x4. Table 1 shows comparisons with SOTA methods, while Supplementary Material Figures 10-15 present qualitative examples. LaMamba-Diff-XL achieves a SOTA FID compared to all prior diffusion models of similar sizes and uses 61.6% less GFLOPs compared to DiT-XL/2. In the unconditional setting, LaMamba-Diff-XL substantially improves the FID of DiT-XL/2 from 12.03 to 7.76. When using CFG, LaMamba-Diff-XL achieves an FID of

3.01 and the highest IS of 277.19. Note that DiS-H/2 achieves a slightly lower FID (2.88) but uses 37% more parameters and 258% more GFLOPs.

Model	Parameters (M)	GFLOPs	FID
LaMamba-Diff-S	32	3.19	55.68
LaMamba-Diff-B	127	12.32	29.71
LaMamba-Diff-L	449	33.39	18.15
LaMamba-Diff-XL	656	49.90	12.86
DiT-S/2	33	6.06	68.40
DiT-B/2	130	23.01	43.47
DiT-L/2	458	80.71	23.33
DiT-XL/2	675	118.64	19.47

Table 2: Comparison of LaMamba-Diff and DiT scaling properties based on unconditional FID and models trained on ImageNet 256x256 for 400k steps using a batch size 256.

Scaling model complexity. We evaluate the scaling properties of LaMamba-Diff by training models presented in Supplementary Material Table 6 on ImageNet 256x256 for 400k steps. Figure 2 (left) and Table 2 present our results. We observe significant FID improvements when increasing parameters. LaMamba-Diff-XL (656M) improves the FID score by 42.82 compared to LaMamba-Diff-S (32M). Additionally, LaMamba-Diff consistently outperforms DiT models with similar parameter counts, delivering superior performance with significantly fewer GFLOPs. These findings highlight the superior scaling capability of LaMamba-Diff models.

Method	GFLOPs	FID
LaMamba-Diff	3.19	55.68
w/o attention	2.45	69.78
w/o shifting	3.19	58.81
global attention	4.24	53.74

Table 3: Ablation study of attention in LaMamba blocks.

4.3 Ablation Studies

This section investigates the effectiveness of local attention in LaMamba blocks and the network architecture design. Ablations on the visual state space module (VSSM) can be found in the Supplementary Material. Unless otherwise stated, we report FID-50K for variants of LaMamba-Diff-S trained on ImageNet 256x256 for 400K steps with a batch size of 256.

Local Attention in LaMamba blocks are crucial to success. They complement VSSM by capturing detailed local contextual information. In Table 3, we study the effect of local attention and different attention strategies. Specifically, we removed the local attention module entirely (w/o attention), disabled the alternating window shift (w/o shifting), and replaced local attention with global attention.

Table 3 presents our results, indicating that removing local attention leads to a substantial increase in FID by 14.1, confirming the importance of modeling detailed local contexts. Additionally, removing the shifting scheme for windowed attention leads to a 3.13 increase in FID, suggesting that the propagation of fine-grained information across window boundaries is useful. Global attention marginally improves FID by 3.5 despite bringing a significant 32.5% computational overhead, indicating that the combination of local attention and VSSM is as powerful as global attention plus VSSM in capturing both global and local contexts while incurring much less computation.

Ablation Type	Method	Initial dim	Param (M)	FLOP (G)	FID
	LaMamba-Diff	96	32	3.19	55.68
Downsampling	1x	144	30	6.99	53.49
	3x	48	33	1.93	62.23
Block Number	Decoder -1	104	32	3.08	54.93
	All +1	80	33	3.51	63.54
	All -1	112	35	2.93	58.17
Isotropic architecture	Patch size 1	384	34	24.96	52.01
	Patch size 2	384	34	6.19	91.89
	Patch size 4	384	34	1.96	129.92
	Patch size 8	384	34	0.65	172.10
U-Net Shortcut	Concatenate	64	33	3.02	89.10

Table 4: Ablation studies on network architectural design. The initial dim represents the hidden dimension of stage 1 of the LaMamba-Diff Encoder. Param denotes the number of parameters.

Network Architecture. In this section, we conduct a comparative analysis of various architectural design choices for LaMamba-Diff. Specifically, we evaluate four architectural variations: (1) number of downsampling blocks; (2) number of blocks in each U-Net stage; (3) isotropic architecture instead of U-Net; (4) channel concatenation instead of addition in U-Net shortcuts. The hidden dimension is adjusted to ensure that all variants have a similar number of parameters. The results of these ablations are presented in Table 4.

In LaMamba-Diff, we downsample twice in stages 1 and 2 of the encoder, resulting in a bottleneck resolution of 8x8 for 256x256 input images. Here, we investigate the impact of downsampling only once ($1\times$) and three times ($3\times$), corresponding to bottleneck resolutions of 16x16 and 4x4, respectively. Our observations reveal that $3\times$ downsampling leads to a decline in performance, while $1\times$ downsampling marginally improves FID by 3.9, albeit with an 118% increase in GFLOPs.

Next, we analyze the impact of varying the number of LaMamba blocks in each U-Net stage. LaMamba-Diff-S originally includes 2 blocks per encoder stage and 3 blocks per decoder stage, bearing a large decoder design akin to StableDiffusion [31]. We explore configurations with 2 blocks in every decoder stage (Decoder -1), adding one block to all stages (All +1), and removing one block from all stages (All -1). Our results show that increasing blocks while reducing hidden dimensions degrades performance. Conversely, using fewer decoder blocks (Decoder -1) achieves similar FID and GLFOPs, suggesting an alternative design for LaMamba-Diff.

We compare LaMamba-Diff with an isotropic architecture, essentially a DiT [27] incorporating LaMamba blocks. An isotropic design with a patch size of 1 improves FID by 3.67 but requires 7.8 times more GFLOPs, while larger patch sizes perform worse. Our U-Net architecture offers a far better trade-off between FID and GFLOPs.

LaMamba-Diff achieves significantly lower FID compared to channel concatenation in U-Net shortcuts. Overall, ablation studies highlight that the hidden dimension size is crucial to LaMamba-Diff's effectiveness.

Resolution	Model	GFLOPs	Image/s
256×256	DiT-XL/2	118.64	8.9
	LaMamba-Diff-XL	50.46	6.8
512×512	DiT-XL/2	524.60	1.9
	LaMamba-Diff-XL	201.20	2.0
1024×1024	DiT-XL/2	2910.30	0.3
	LaMamba-Diff-XL	804.18	0.5

Table 5: Comparison of GFLOPs and Images generated per second between LaMamba-Diff-XL and DiT-XL/2. Image/s is computed using 8 H800 GPUs and 250 sampling steps.

5 Computational Complexity

Theoretical analysis. The majority of FLOPs in VSSM are due to the the 2D Selective Scan (SS2D) mechanism, which computes 4 SSM processes with a hidden dimension expansion ratio of 2 [21]. Thus, we approximate the complexity of VSSM with SS2D.

Given a noisy diffusion latent $z_t \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times D}$, where D is the hidden dimension, the computational complexity of SS2D, windowed multi-head self-attention (W-MSA), and feedforward net (FFN) are given below:

$$\Omega(\text{SS2D}) \approx 4\Omega(\text{SSM}) = 4(3L(2D)N + L(2D)N), \tag{1}$$

$$\Omega(\text{W-MSA}) = 4HWD^2 + 2M^2HWD, \tag{2}$$

$$\Omega(\text{FFN}) = 4LD^2, \tag{3}$$

where $L = H \times W$ denotes the sequence length, N is the SSM state dimension, and $M = 8$ is the fixed window size (matching the 8×8 bottleneck resolution for 256×256 images) SS2D approximates $4\Omega(\text{SSM})$ complexity due to four scan paths. SSM complexity has two terms: $3L(2D)N$ from computing vectors $\bar{B}, C$, and scalar $\bar{D}$; and $L(2D)N$ from computing matrix $\bar{A}$. Thus, VSSM is linear in L. With fixed M, W-MSA is also linear in L. Consequently, all LaMamba components (VSSM, local attention, and FFN) exhibit linear complexity, making LaMamba-Diff inherently linear. Negligible-GFLOPs exclusions for theoretical analysis: SS2D 4-output reshaping/merging and LaMamba block AdaLN/α-scaling.

Empirical analysis. We empirically evaluate the computational complexity and latency of the largest LaMamba-Diff and DiT [27] models in Table 5. With similar parameter counts, LaMamba-Diff-XL reduces GFLOPs by 57.5%and 72.4%at 256×256 and 1024×1024 resolutions, respectively. It achieves 1.7× faster inference on 1024×1024 resolution than DiT/XL-2, but is slower on 256×256 resolution due to limited SSM operator parallelization on GPU.

6 Conclusion

In this paper, we have proposed LaMamba-Diff, a novel linear-time backbone network for diffusion models. It efficiently captures both global contexts and local dependencies from input tokens. Our experiments demonstrate that LaMamba-Diff, with comparable number of parameters and significantly fewer GLFOPs, achieves very competitive performance against state-of-the-art diffusion backbones. Furthermore, our LaMamba block demonstrates excellent scalability, and like DiT, can be taken as a generic building block for various types of diffusion models, including both text-to-image and text-to-video diffusion models.

Acknowledgement

This work was supported in part by Hong Kong Research Grants Council under NSFC/RGC Collaborative Research Scheme (Grant CRS_HKU703/24).

References

[1] Md Atik Ahamed and Qiang Cheng. Mambatab: A simple yet effective approach for handling tabular data. *arXiv preprint arXiv:2401.08867*, 2024.

[2] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22669–22679, 2023.

[3] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[7] Zhengcong Fei, Mingyuan Fan, Changqian Yu, and Junshi Huang. Scalable diffusion models with state space backbone. *arXiv preprint arXiv:2402.05608*, 2024.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[9] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[10] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *Advances in neural information processing systems*, 34:15908–15919, 2021.

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[12] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[13] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[14] Vincent Tao Hu, Stefan Andreas Baumann, Ming Gui, Olga Grebenkova, Pingchuan Ma, Johannes Fischer, and Bjorn Ommer. Zigma: Zigzag mamba diffusion model. *arXiv preprint arXiv:2403.13802*, 2024.

[15] Tao Huang, Xiaohuan Pei, Shan You, Fei Wang, Chen Qian, and Chang Xu. Localmamba: Visual state space model with windowed selective scan. *arXiv preprint arXiv:2403.09338*, 2024.

[16] Allan Jabri, David Fleet, and Ting Chen. Scalable adaptive computation for iterative generation. *arXiv preprint arXiv:2212.11972*, 2022.

[17] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:26565–26577, 2022.

[18] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in neural information processing systems*, 32, 2019.

[19] Jiarun Liu, Hao Yang, Hong-Yu Zhou, Yan Xi, Lequan Yu, Yizhou Yu, Yong Liang, Guangming Shi, Shaoting Zhang, Hairong Zheng, et al. Swin-umamba: Mamba-based unet with imagenet-based pretraining. *arXiv preprint arXiv:2402.03302*, 2024.

[20] Jiuming Liu, Ruiji Yu, Yian Wang, Yu Zheng, Tianchen Deng, Weicai Ye, and Hesheng Wang. Point mamba: A novel point cloud backbone based on state space model with octree-based ordering strategy. *arXiv preprint arXiv:2403.06467*, 2024.

[21] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024.

[22] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[24] Charlie Nash, Jacob Menick, Sander Dieleman, and Peter W Battaglia. Generating images with sparse representations. *arXiv preprint arXiv:2103.03841*, 2021.

[25] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

[26] Yuta Oshima, Shohei Taniguchi, Masahiro Suzuki, and Yutaka Matsuo. Ssm meets video diffusion models: Efficient video generation with structured state spaces. *arXiv preprint arXiv:2403.07711*, 2024.

[27] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[28] Xiaohuan Pei, Tao Huang, and Chang Xu. Efficientvmamba: Atrous selective scan for light weight visual mamba. *arXiv preprint arXiv:2403.09977*, 2024.

[29] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[30] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[31] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[33] Jiacheng Ruan and Suncheng Xiang. Vm-unet: Vision mamba unet for medical image segmentation. *arXiv preprint arXiv:2402.02491*, 2024.

[34] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[35] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[36] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

[37] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[38] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[39] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[40] Yao Teng, Yue Wu, Han Shi, Xuefei Ning, Guohao Dai, Yu Wang, Zhenguo Li, and Xihui Liu. Dim: Diffusion mamba for efficient high-resolution image synthesis. *arXiv preprint arXiv:2405.14224*, 2024.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[42] Jing Nathan Yan, Jiatao Gu, and Alexander M Rush. Diffusion models without attention. *arXiv preprint arXiv:2311.18257*, 2023.

[43] Chenhongyi Yang, Zehui Chen, Miguel Espinosa, Linus Ericsson, Zhenyu Wang, Jiaming Liu, and Elliot J Crowley. Plainmamba: Improving non-hierarchical mamba in visual recognition. *arXiv preprint arXiv:2403.17695*, 2024.

[44] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 2022.

[45] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024.