Boosting Diffusion Models with Moving Average Sampling in Frequency Domain*

Yurui Qian†, Qi Cai‡, Yingwei Pan‡, Yehao Li‡, Ting Yao‡, Qibin Sun†, and Tao Mei‡
†University of Science and Technology of China ‡HiDream.ai Inc.
qyr123@mail.ustc.edu.cn, {cqcaiqi, pandy, liyehao, tiyao}@hidream.ai,
qibinsun@ustc.edu.cn, tmei@hidream.ai

Abstract

Diffusion models have recently brought a powerful revolution in image generation. Despite showing impressive generative capabilities, most of these models rely on the current sample to denoise the next one, possibly resulting in denoising instability. In this paper, we reinterpret the iterative denoising process as model optimization and leverage a moving average mechanism to ensemble all the prior samples. Instead of simply applying moving average to the denoised samples at different timesteps, we first map the denoised samples to data space and then perform moving average to avoid distribution shift across timesteps. In view that diffusion models evolve the recovery from low-frequency components to high-frequency details, we further decompose the samples into different frequency components and execute moving average separately on each component. We name the complete approach "Moving Average Sampling in Frequency domain (MASF)". MASF could be seamlessly integrated into mainstream pre-trained diffusion models and sampling schedules. Extensive experiments on both unconditional and conditional diffusion models demonstrate that our MASF leads to superior performances compared to the baselines, with almost negligible additional complexity cost.

1. Introduction

Diffusion model is an increasingly appealing direction to improve the state-of-the-art innovations for generative tasks. In the regime of computer vision, multiple milestones under diffusion models have been established on unconditional and conditional image synthesis [5, 18, 37, 41], image restoration [10, 32, 48], inpainting [1, 23, 30], captioning [31], video generation [9, 19, 52], 3D/audio synthesis [6, 22, 40, 49]. In general, the diffusion model consists of a forward process that progressively introduces Gaussian noise into an image, and a denoising network that approximates the reverse of the forward process to produce an image from noise. Compared to generative adversarial networks (GAN) [13, 21, 34], diffusion models are shown capable of better training stability and less sensitivity to hyperparameters, leading to high-quality and coherent samples and alleviating mode collapse. The representative works are DDPM [18] and DDIM [41], which generate samples from white noise by a Markov chain and a "short" generative Markov chain corresponding to non-Markovian forward process, respectively. Despite accelerating sampling several orders of magnitude by DDIM, both DDPM and DDIM capitalize only on the current sample to produce the next one, introducing discretization errors [29]. The stochastic nature of these errors brings instability into denoising. As illustrated in Figure 1, the denoised sample thus tends to oscillate around the ground truth value and the denoising process necessitates lengthy timesteps to converge. To mitigate this issue, some high-order solvers including DPM-Solver [28, 29] and UniPC [60], exploit the resulting samples of the previous K (usually 1 to 3) timesteps to refine the prediction of the current sample. Nevertheless, the

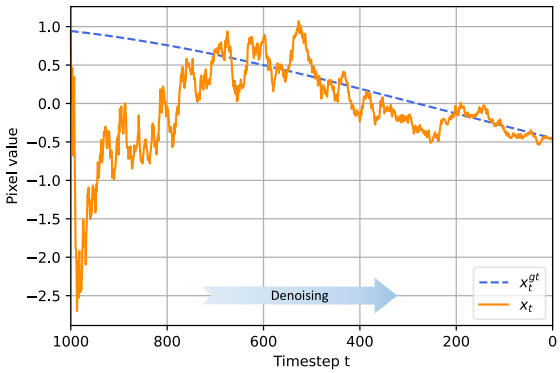


Figure 1. We utilize a diffusion model from ADM [8] pre-trained on ImageNet-64 to sample from white noise and capture the intermediate output as denoised sample $\boldsymbol{x}_t$. Subsequently, we plot the pixel value of $\boldsymbol{x}_t$ with respect to generative timesteps. The starting point and ending point of the trajectory of $\boldsymbol{x}_t$ are regarded as the ground truth for the noisy image and clean image, respectively. With that we calculate the ground truth of $\boldsymbol{x}_t$ at any timestep t using the formulation defined in forward process.

*This work was performed at HiDream.ai.

prior information in the generative process is still not yet fully leveraged for denoising.

Another observation in diffusion models is the frequency principle that diffusion models denoise the low-frequency signals first, and gradually add high-frequency details into the sample. Most existing diffusion approaches, however, seldom explore this principle in generative process and treat the sampling consistent as the denoising proceeds, regardless of the connection between frequency evolution and generative timesteps. In contrast, Spectral Diffusion [53] presents dynamic feature extraction at each generative timestep to adjust frequency characteristics. The requirements of a specialized network design and the re-training of the denoising network make it difficult to apply to existing diffusion models.

In response to the above issues, we propose a training-free approach, namely Moving Average Sampling in Frequency domain (MASF), to enhance the stability of generative process. Technically, MASF reframes the iterative denoising process as model optimization and delves into the moving average mechanism to utilize all the prior samples at each timestep. Note that the diffusion models denoise samples from $\boldsymbol{x}_T$ to $\boldsymbol{x}_0$, and each sample inherently lies in distinct distribution between white noise and the initial data distribution. This case deviates from the assumption in model optimization that the parameters should come from a constant distribution. As such, it is inappropriate to directly apply moving average to $\boldsymbol{x}_t$. Instead, we map sample $\boldsymbol{x}_t$ to the data space $\boldsymbol{x}_0$ and execute moving average on $\boldsymbol{x}_0$ to reduce distribution shift across sampling at different timesteps. Furthermore, MASF decomposes the sample into frequency components and performs separate moving average on each component to dynamically evolve different components along the denoising process. Specifically, we devise a weighting scheme to prioritize low-frequency components denoising in the early timesteps and progressively contribute more weights to high-frequency components in the later timesteps. To ensure compatibility with existing diffusion networks that only accept the complete sample, we reconstruct the sample from all frequency components before feeding it into the denoising network at each timestep. The conversion between sample and frequency components only introduces insignificant extra overheads.

In summary, we have made the following contributions: 1) The proposed MASF is shown capable of leveraging all the prior samples in frequency domain to better denoise the current sample in generative process. 2) The exquisitely designed MASF is shown able to be seamlessly integrated into existing diffusion models. 3) MASF has been properly analyzed and verified through extensive experiments on both unconditional and conditional diffusion models to validate its efficacy.

2. Related Work

Sampling Methods in Diffusion Models. Diffusion models have made tremendous progress in generating high-fidelity images [8, 36, 38, 39, 57], wherein sampling methods play a crucial role in unleashing their power for image generation with minimal computational cost. [42] first formulates the diffusion sampling process as the solving of stochastic differential equations (SDEs) [2, 12, 20, 59] and ordinary differential equations (ODEs) [27, 29, 41, 60]. Specifically, some works [12, 28, 29, 58] build on approximating exponential integrators to reduce the truncation error, while the others [25, 27] follows traditional numeric methods [4, 45] to solve ODEs. In addition, [8, 17] imposes advanced guidance during sampling process to facilitate generation. In this work, we propose an orthogonal design to the aforementioned sampling methods from the perspective of improving denoising stability.

Sampling Process Stabilization. In an effort to alleviate the denoising instability issue, some prior works explore momentum to stabilize sampling process. For instance, [46] examines the divergence artifacts in scenarios with limited sampling steps and incorporates the momentum technique into existing sampling methods. [47] associates the diffusion process with stochastic optimization procedure and draws inspiration from momentum SGD to design momentum-based forward process to accelerate training convergence. Similarly, inspired by Adam optimizer, [44] proposes a new sampler that follows the convention in Adam optimizer to define their momentum and update velocity. In contrast, our proposed MASF executes moving average in frequency domain to novelly excavate the frequency dynamics for stabilizing sampling process along frequency evolution.

Frequency Modeling in Diffusion Models. Wavelet decomposition [14, 33] has been widely adopted in conventional generative methods (e.g., GANs [11, 50, 51, 56]) to exploit additional frequency-aware information in frequency domain. Recently, several advances start to integrate diffusion models with wavelet information. In particular, [24] employs score-based models in wavelet spectrum to promote image colorization, while [15] accelerates score-based generative models by factorizing data distribution into multiscale conditional probabilities of wavelet coefficients. Additionally, [35] and [55] design frequency-aware architectures to process data in frequency domain, pursuing faster processing and higher image quality, respectively. Spectral Diffusion [53] also studies the frequency evolution in denoising procedure, and utilizes wavelet gating to trigger spectrum-aware distillation, leading to reduced computation cost. Nevertheless, Spectral Diffusion requires additional re-training of a specialized denoising network, and thus fails to be directly applied to different diffusion models. Instead, our approach seeks a training-free solution

that exploits all the prior samples in frequency domain to strengthen the stability of denoising process, which can be seamlessly integrated into any diffusion models.

3. Preliminaries

Here we first briefly review the typical Denoising Diffusion Probabilistic Models (DDPM) [18] and Denoising Diffusion Implicit Model (DDIM) [41] for sampling.

Denoising Diffusion Probabilistic Models. DDPM consists of a forward process and a denoising process. For the forward process, DDPM transitions from intractable data distribution, denoted as $\boldsymbol{x}_0 \sim q_0(\boldsymbol{x}_0)$, to Gaussian distribution $q_T(\boldsymbol{x}_T) \sim \mathcal{N}(\boldsymbol{x}_T; \mathbf{0}, \boldsymbol{I})$. This is achieved by progressively adding Gaussian noise to the original image $\boldsymbol{x}_0$, and the transition distribution at timestep t is thus defined as:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_t; \sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}), \quad (1)$$

where $\alpha_1, \ldots, \alpha_T$ are predefined variance schedules. Following the properties of chained Gaussian processes, DDPM defines $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$ and the value of $\boldsymbol{x}_t$ is thus calculated in a single step:

$$\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}). \quad (2)$$

The denoising process employs a neural network to approximate the conditional distribution $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$. The optimization objective can be derived from the variational lower bound, which is expressed as:

$$L = D_{KL}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)||p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)). \quad (3)$$

The analytical form of $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$ is defined by $\mathcal{N}(\boldsymbol{x}_{t-1}; \tilde{\boldsymbol{\mu}}_{t-1}, \sigma_{t-1}^2\boldsymbol{I})$, while $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ takes the form $\mathcal{N}(\boldsymbol{x}_{t-1}; \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \sigma_{t-1}^2\boldsymbol{I})$. The specific formulations of $\tilde{\boldsymbol{\mu}}_{t-1}$ and $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t)$ are measured as follows:

$$\tilde{\boldsymbol{\mu}}_{t-1} = \frac{1}{\sqrt{\alpha_t}}(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}), \quad (4)$$

$$\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)). \quad (5)$$

In this context, $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$, generated by the diffusion model, serves to approximate the added noise $\boldsymbol{\epsilon}$, based on the noised image $\boldsymbol{x}_t$ and the specific timestep t. After eliminating constant scaling factors in this loss function, DDPM allows for a simplification of the optimization objective:

$$L_{t-1} = \mathbb{E}_{\boldsymbol{x}_0, \boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\|^2\right]. \quad (6)$$

Denoising Diffusion Implicit Model. DDIM upgrades DDPM framework by integrating a non-Markovian process, which effectively decouples $\boldsymbol{x}_{t-1}$ from $\boldsymbol{x}_t$, enabling the skipping of timesteps to accelerate the sampling process.

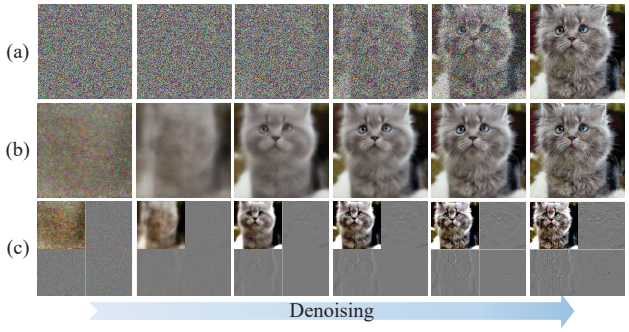


Figure 2. The evolution of (a) denoised sample $\boldsymbol{x}_t$, (b) estimated sample in data space $\boldsymbol{x}_0^t$ and (c) four subbands in frequency domain of $\boldsymbol{x}_0^t$ along denoising process. Here each group of subbands is achieved via wavelet decomposition, yielding four different frequency components: ll (↖), lh (↗), hl (↙), and hh (↘).

In this way, DDIM redefines the denoising distribution as:

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\sqrt{\bar{\alpha}_{t-1}}\boldsymbol{x}_0^t + \sqrt{1-\bar{\alpha}_t-\eta_t^2}\frac{\boldsymbol{x}_t - \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0^t}{\sqrt{1-\bar{\alpha}_t}}, \eta_t^2\boldsymbol{I}). \quad (7)$$

Here $\boldsymbol{x}_0^t$ denotes the estimated original sample from the perturbed sample $\boldsymbol{x}_t$, which is calculated as:

$$\boldsymbol{x}_0^t = (\boldsymbol{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t))/\sqrt{\bar{\alpha}_t}. \quad (8)$$

The model's behavior is contingent on the value of η_t. Specifically, when η_t is set as $\sqrt{(1-\bar{\alpha}_{t-1})/(1-\bar{\alpha}_t)}\sqrt{1-\alpha_t}$, it aligns with the sampling process of DDPM. Conversely, setting η_t to 0 completely eliminates stochasticity during sampling, thereby turning into the sampling process of DDIM.

4. Our Approach

Now we proceed to present our central proposal, Moving Average Sampling in Frequency domain (MASF), aiming to enhance the stability of the denoising process. This section starts by introducing the moving average sampling within the data space. After that, we novelly capitalize on Discrete Wavelet Transformation (DWT) to extend such moving average strategy into the frequency domain. Furthermore, a new dynamic weighting scheme is designed to dynamically perform moving average over different frequency components, pursuing harmonized stabilization along with frequency evolution at denoising process. Figure 3 depicts an overview of our MASF framework.

4.1. Moving Average in Data Space

Recall that during typical denoising process, the estimated denoised sample $\boldsymbol{x}_t$ always oscillates around its ground truth value due to the stochastic nature of discretization errors [29]. The local errors committed at each timestep accumulate into the global error, which can potentially disrupt

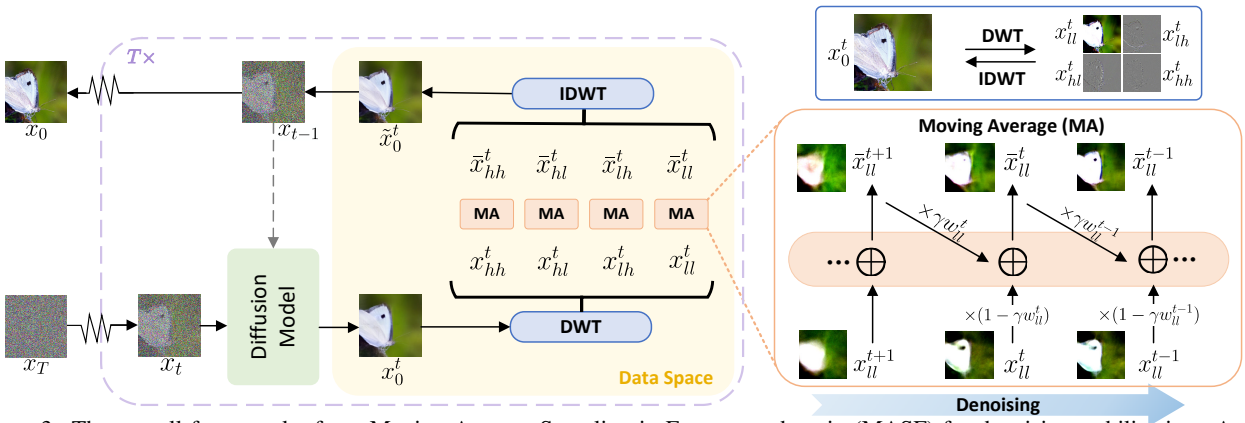


Figure 3. The overall framework of our Moving Average Sampling in Frequency domain (MASF) for denoising stabilization. At each denoising timestep t, MASF first maps the denoised sample $\boldsymbol{x}_t$ into data space, leading to the estimated sample $\boldsymbol{x}_0^t$. We then perform frequency decomposition of $\boldsymbol{x}_0^t$ via Discrete Wavelet Transformation (DWT) and achieve four subbands ($\boldsymbol{x}_{\{ll,lh,hl,hh\}}^t$). After that, MASF updates each frequency component (e.g., the low-frequency component $\boldsymbol{x}_{ll}^t$) through moving average over prior samples, pursuing harmonized stabilization along with frequency evolution. The refined subbands $\bar{\boldsymbol{x}}_{\{ll,lh,hl,hh\}}^t$ are finally converted back to image domain via Inverse DWT (IDWT) to trigger the subsequent denoising process.

the sampling process and result in denoising instability. Intuitively, as shown in Figure 2 (a), the denoising process resembles model optimization, where the denoised sample is iteratively refined via the learnt diffusion model, akin to parameter (denoised sample) optimization (denoising) in model training. This observation motivates us to explore the commonly adopted moving average technique in conventional model optimization, to stabilize the optimization trajectory of the denoised sample $\boldsymbol{x}_t$ at inference. Nevertheless, considering that denoised samples $\boldsymbol{x}_t$ at different timesteps are perturbed at various noise scales, simply applying moving average over primary $\boldsymbol{x}_t$ derived from distinct distributions might inject harmful distortions into optimization trajectory. As an alternative, we propose to map the denoised sample $\boldsymbol{x}_t$ back to initial data space, leading to estimated sample $\boldsymbol{x}_0^t$ consistently predicting clean sample (see Figure 2 (b)). After that, we perform moving average over prior estimated samples $\boldsymbol{x}_0^t$ to stabilize the denoising process. It is worthy to note that this design of moving average in data space is readily pluggable to any sampling solver. We next discuss how to integrate moving average into DDIM framework [41] and other solvers.

Moving Average in DDIM Solver. Since DDIM explicitly computes estimated sample $\boldsymbol{x}_0^t$ during sampling, our moving average design can be directly applied over the measured $\boldsymbol{x}_0^t$. Formally, in order to stabilize the optimization trajectory of $\boldsymbol{x}_0^t$, we maintain a global moving average $\bar{\boldsymbol{x}}_0^t$ that aggregates all prior estimated samples. At each timestep t, we update $\bar{\boldsymbol{x}}_0^t$ by additionally augmenting the observed $\boldsymbol{x}_0^t$ ($0 \leq t < T$):

$$\bar{\boldsymbol{x}}_0^t = (1-\gamma)\boldsymbol{x}_0^t + \gamma \bar{\boldsymbol{x}}_0^{t+1}, \tag{9}$$

where γ is a balancing hyperparameter that controls the degree of dependence on the previous moving average versus the current sample. A larger γ reflects more reliance on previous $\bar{\boldsymbol{x}}_0^{t+1}$ but less focus on $\boldsymbol{x}_0^t$, leading to a smoother trajectory. Based on $\bar{\boldsymbol{x}}_0^t$, we can estimate more stable and accurate denoised sample $\boldsymbol{x}_{t-1}$ by simply replacing $\boldsymbol{x}_0^t$ with the global moving average $\bar{\boldsymbol{x}}_0^t$ in Eq. (7) of DDIM.

In addition, we observe that the different spatial locations in an image evolve in different rates during denoising process. For example, as shown in Figure 2 (b), the cat's face in $\boldsymbol{x}_0^t$ evolves more sharply than left bottom corner of background. Motivated by this, we introduce an adaptive weight $\boldsymbol{w}_t$ to modify γ for different spatial locations. Here $\boldsymbol{w}_t$ is measured as the discrepancy between $\boldsymbol{x}_0^t$ and $\bar{\boldsymbol{x}}_0^{t+1}$. In this way, when the spatial location evolves sharply (i.e., large discrepancy between $\boldsymbol{x}_0^t$ and $\bar{\boldsymbol{x}}_0^{t+1}$), we will amplify the reliance on the global moving average $\bar{\boldsymbol{x}}_0^{t+1}$ to pursue more stable denoising process. Eventually, the update function of $\bar{\boldsymbol{x}}_0^t$ is operated as:

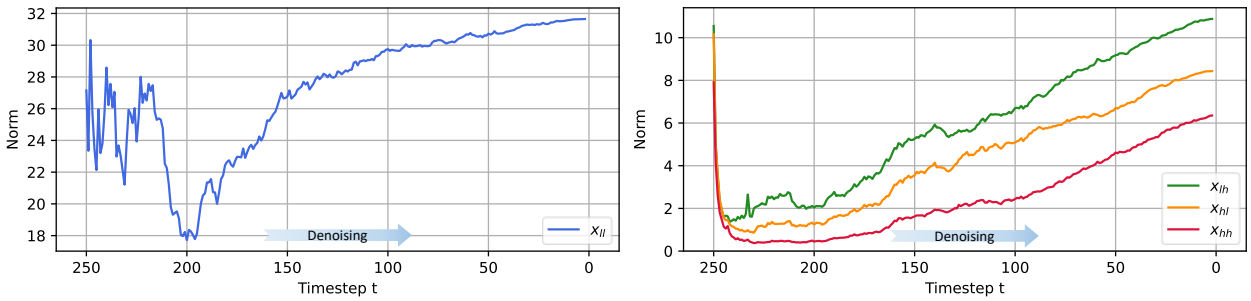
$$\bar{\boldsymbol{x}}_0^t = (\mathbf{1} - \gamma \boldsymbol{w}_t) \circ \boldsymbol{x}_0^t + \gamma \boldsymbol{w}_t \circ \bar{\boldsymbol{x}}_0^{t+1}, \tag{10}$$

where $\circ$ denotes element-wise multiplication.

Moving Average in Other Solvers. For solvers [28, 41] that explicitly define $\boldsymbol{x}_0$ in their formulations, we can directly apply Eq. (10) to integrate the moving average technique. For other solvers [18, 27] which leverage $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$ instead of $\boldsymbol{x}_0$ in updating function, we first calculate $\boldsymbol{x}_0$ using Eq. (8) and then apply moving average as in Eq. (10). Subsequently, $\boldsymbol{x}_0^t$ is replaced with $\bar{\boldsymbol{x}}_0^t$ to achieve a refined $\bar{\boldsymbol{\epsilon}}_\theta(\boldsymbol{x}_t, t) = (\boldsymbol{x}_t - \sqrt{\bar{\alpha}_t}\bar{\boldsymbol{x}}_0^t)/\sqrt{1-\bar{\alpha}_t}$, which can be seamlessly integrated into those solvers.

4.2. Moving Average in Frequency Domain

A well-known evolution law of diffusion models at denoising process is first focusing on the recovery of low-frequency component in the earlier timesteps and gradu-



(a) Low-frequency Component. (b) High-frequency Components.

Figure 4. Evolution (l_2 norm) of different frequency subbands during denoising process. (a) The low-frequency subband oscillates sharply at the beginning and stabilizes after a certain timestep. (b) In contrast, the l_2 norm of high-frequency subbands drops rapidly into a small value and then increases steadily along with the denoising process. (We use the model from ADM [8] pre-trained on ImageNet-64)

ally turning to recover high-frequency details in the later timesteps. Taking the decomposed frequency components in Figure 2 (c) as an example, the low-frequency component (ll) only evolves sharply in the earlier timesteps, while the high-frequency components (lh, hl, hh) start to evolve significantly in the later timesteps. As such, we further extend moving average sampling into frequency domain by executing moving average separately on different frequency components, thereby encouraging more harmonized stabilization along with frequency evolution.

Technically, MASF employs Discrete Wavelet Transformation (DWT) [14] to decompose the estimated sample $\boldsymbol{x}_0^t \in \mathbb{R}^{H\times W}$ into four wavelet subbands: $\boldsymbol{x}_{ll}^t$, $\boldsymbol{x}_{lh}^t$, $\boldsymbol{x}_{hl}^t$, $\boldsymbol{x}_{hh}^t$. The dimension of each subband is $\mathbb{R}^{H/2\times W/2}$. Note that here we implement DWT as the classical Haar wavelet [43] for simplicity. Among the four wavelet subbands, $\boldsymbol{x}_{ll}^t$ refers to the low-frequency component that reflects the basic object structure (resembling a downsampled image), while $\boldsymbol{x}_{\{lh,hl,hh\}}^t$ represent high-frequency components that capture texture details. After that, we separately apply moving average over each kind of frequency subband as in Eq. (10):

$$\bar{\boldsymbol{x}}_f^t = (\mathbf{1} - \gamma \boldsymbol{w}_f^t) \circ \boldsymbol{x}_f^t + \gamma \boldsymbol{w}_f^t \circ \bar{\boldsymbol{x}}_f^{t+1}, \tag{11}$$

where $f \in \{ll, lh, hl, hh\}$. As such, each frequency component is independently augmented with the corresponding moving average (denoted as $\bar{\boldsymbol{x}}_f^t$). This ensures that the trajectories of these frequency components remain non-interfering. Considering that the denoising network is fed with samples in image space, we convert the moving average in frequency domain ($\bar{\boldsymbol{x}}_f^t$) back to image domain via Inverse DWT (IDWT) at each timestep.

4.3. Frequency Weighting Scheme

The aforementioned extension of moving average from data space to frequency domain elegantly triggers the interaction between typical denoising process and frequency evolution. Nevertheless, such extension leaves the inherent different priorities for each frequency component along denoising process under-exploited, resulting in the sub-optimal denoising stabilization. In particular, we conduct a comprehensive analysis of frequency evolution in Figure 4 by visualizing the l_2 norm of each frequency component at denoising process. As shown in this figure, the low-frequency subband $\boldsymbol{x}_{ll}^t$ exhibits sharp oscillation in earlier timesteps and gradually stabilizes after a certain timestep (approximately at 100 steps). Instead, the evolution of high-frequency components $\boldsymbol{x}_f^t$ ($f \in \{ll, lh, hl, hh\}$) reflects a different trend. They drop dramatically into a relatively small value, but subsequently show a steady increase as the denoising process progresses. Such observation reveals that the denoising process often prioritizes reconstructing low-frequency component in the earlier stage, and then focuses on the recovery of high-frequency details later.

Based on these observations, we further upgrade the moving average in frequency domain with a new dynamic weighting scheme to better align with the evolution dynamics of different frequency components. This dynamic weighting scheme prioritizes low-frequency components in the early timesteps and gradually amplifies the weights of high-frequency components when converting $\boldsymbol{x}_f^t$ ($f \in \{ll, lh, hl, hh\}$) back to image domain. The detailed operation of dynamic weighting scheme is defined as follows:

$$\hat{\boldsymbol{x}}^t = \text{IDWT}(\beta_f(t)\boldsymbol{x}_f^t | f = ll, lh, hl, hh), \tag{12}$$

where $\beta_{ll}(t)$ decreases linearly as denoising progresses (as timestep evolves from $t+1$ to t) and $\beta_{\{lh,hl,hh\}}(t)$ increase linearly. By integrating previous moving average operation (Eq. (11)) with this dynamic weighting scheme, we can achieve the refined version of $\boldsymbol{x}_0^t$:

$$\tilde{\boldsymbol{x}}_0^t = \text{IDWT}\big(\beta_f(t)\big((\mathbf{1} - \gamma \boldsymbol{w}_f^t) \circ \boldsymbol{x}_f^t + \gamma \boldsymbol{w}_f^t \circ \bar{\boldsymbol{x}}_f^{t+1}\big)\big). \tag{13}$$

Finally, the refined estimated sample $\tilde{\boldsymbol{x}}_0^t$ is fed into denoising network to enable a stabilized denoising process.

5. Experiments

We empirically verify the merit of MASF for image generation using diffusion models. The first experiment validates MASF on both conditional and unconditional models across different datasets. The second experiment integrates MASF into recent advances of sampling techniques to examine its impact when combining with state-of-the-art models. The third experiment analyzes how each design in MASF influences the overall performance.

Table 1. FID performances of 50K samples for class-conditional generation on ImageNet with different resolutions and NFEs.

Method	Resolution	NFE			
		10	15	20	25
DDIM	64	52.19	24.56	15.18	11.04
+MASF	64	**22.63**	**11.23**	**7.64**	**6.32**
DDIM	128	20.36	14.87	12.63	11.48
+MASF	128	**17.19**	**12.22**	**10.16**	**9.37**
DDIM	256	25.68	19.49	17.23	16.56
+MASF	256	**22.64**	**17.67**	**16.08**	**15.51**

Table 2. FID performances of 30K samples for text-conditional generation on MS-COCO with different solvers and NFEs.

Method	NFE			
	10	15	20	25
DDIM	35.10	31.05	29.18	28.51
+MASF	**25.67**	**23.04**	**22.00**	**21.80**
DPM-Solver++ (3Fast)	6.35	6.03	6.03	5.76
+MASF	**6.20**	**6.01**	**5.97**	**5.69**

5.1. Results on Conditional/Unconditional Models

Conditional Models. Conditional generation leverages control signals into image generation process. The typical conditional models are briefly grouped into two categories: class-conditional and text-conditional. For class-conditional sampling, we utilize pixel-space pre-trained models by the ADM framework [8] to sample 50K images on different resolutions in the range of 64, 128, and 256, and conduct the experiments on ImageNet [7] which contains 1,000 distinct classes. We also execute the evaluations with respect to the number of function evaluations (NFE). Table 1 summarizes the Fréchet inception distance (FID) [16] performances of applying MASF to DDIM [41] with different NFEs on ImageNet. Overall, using MASF consistently exhibits better FID scores across four NFEs on ImageNet with three image resolutions. The performance gain is larger at small NFE where the instability issue is more severe, demonstrating the advantage of MASF to stabilize the sampling process through moving average in the frequency domain. Notably, MASF brings only 0.97% extra computational cost to the entire sampling process, making the overhead of the deployment of MASF to diffusion models negligible.

For text-conditional generation, we exploit the pre-learnt diffusion model of U-ViT [3] on MS-COCO [26] dataset to produce image samples with the resolution of 256×256. Following the U-ViT protocol, we randomly select 30K prompts from MS-COCO validation set for FID evaluation. Table 2 lists FID comparisons for text-conditional generation with different solvers and NFEs. As indicated by the results, applying MASF to DDIM reduces the FID score from 35.10 to 25.67 at the NFE of 10, making an absolute improvement of 9.43. Notably, the FID scores on MS-COCO are generally higher than that on ImageNet. We speculate that this may be the result of more complex structures in MS-COCO images. Furthermore, we employ a stronger solver DPM-Solver++ [28] instead of DDIM, significantly lowering the FID scores. Impressively, MASF still exhibits its superiority over DPM-Solver++ across all NFEs and MASF leads FID score by 0.07 when sampling 25 steps. To qualitatively validate our MASF, we showcase four image examples generated by DPM-Solver++ and DPM-Solver++ plus MASF in Figure 5. The images clearly show that MASF by involving the utilization of moving average in the frequency domain generates higher quality images with less distortions.

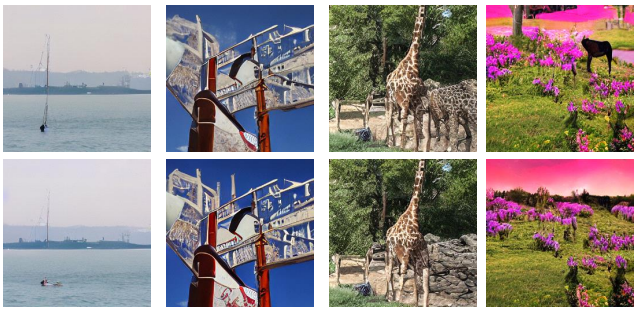


Figure 5. The generated images on MS-COCO using DPM-Solver++ (top) and DPM-Solver++ plus MASF (bottom).

Unconditional Models. Different from conditional generation which leverages some specific control signals, unconditional sampling produces images for a particular class completely from pure Gaussian noise. We assess impact of MASF by sampling 50K samples on two widely-adopted datasets: LSUN [54] and FFHQ [21]. For the LSUN dataset, we exploit three pre-trained pixel-space models by the ADM framework [8] for generating 256×256 images of Horse, Bedroom and Cat, respectively. We run these models with 25 NFE. Table 3 details per-class FID performances of unconditional generation on LSUN dataset. Again, DDIM plus MASF constantly improves the FID scores across all three categories. For the extreme case of Bedroom class, DDIM already achieves a very competitive FID score of 4.11, but our MASF still manages to decrease the score to 3.76. For the FFHQ face dataset, we use a latent-space DPM [36] to sample images for FID measure. As shown in Table 4, MASF contributes a FID decrease of 1.82, 0.89, 0.81 and 0.5 with NFE of 10, 15, 20 and 25, respectively, demonstrating the benefit of MASF.

5.2. Integration with Other Sampling Techniques

To further verify the generalizability and effectiveness of MASF on recent advances of diffusion models, we integrate

Table 3. Per-class FID performances of 50K samples for unconditional generation on LSUN dataset.

Method	Horse	Bedroom	Cat
DDIM	18.41	4.11	12.42
+MASF	**16.07**	**3.76**	**11.65**

Table 4. FID comparisons of 50K samples for unconditional generation on FFHQ dataset with different NFEs.

Method	NFE			
	10	15	20	25
DDIM	11.92	6.92	5.85	5.67
+MASF	**10.10**	**6.03**	**5.04**	**5.17**

Table 5. FID comparisons of 50K samples for conditional generation on ImageNet 128×128 with different guidance scales.

Method	Scale	NFE			
		10	15	20	25
DDIM	0.0	20.36	14.87	12.63	11.48
+MASF	0.0	**17.19**	**12.22**	**10.16**	**9.37**
DDIM	0.5	13.61	9.60	8.01	7.16
+MASF	0.5	**11.29**	**7.63**	**6.17**	**5.49**
DDIM	1.0	11.18	8.07	6.86	6.14
+MASF	1.0	**9.39**	**6.53**	**5.32**	**4.73**
DDIM	2.0	10.35	8.11	7.19	6.66
+MASF	2.0	**8.95**	**6.82**	**5.92**	**5.42**
DDIM	4.0	11.61	10.01	9.26	8.94
+MASF	4.0	**10.29**	**8.73**	**8.06**	**7.72**

MASF into different sampling techniques including Classifier Guidance [8] and high-order solvers [28, 60]. All the performances here are computed on 50K sampled images with the resolution of 128×128.

Classifier Guidance. As introduced in ADM [8], Classifier Guidance enhances generation via modeling the conditional probability of images given a class by a pre-learnt classifier. The diffusion model can take the gradients of the classifier as the condition and a scale is used to adjust the magnitude of the gradients. Table 5 lists the FID comparisons with respect to different levels of scales and NFEs. MASF always leads to an FID decrease across all the scales and NFEs. As expected, the scale 0 implies that Classifier Guidance is not involved in this case, yielding inferior performances. The gains of FID scores are between 1.32 and 2.32 at the NFE of 10, when the scale is set in the range of 0.5, 1.0, 2.0 and 4.0. In particular, DDIM plus MASF with guidance scale of 1.0 and NFE of 25 attains the best FID score of 4.73. The results basically demonstrate the effectiveness of our MASF on the guided diffusion models.

High-order Solvers. Next, we extend the evaluation of MASF from on the basic solvers of DDPM and DDIM to high-order solvers of DPM Solver++ [28], UniPC [60], and F-PNDM [27]. The evaluation protocol follows Classifier Guidance [8] with a fixed scale of 1.0. Table 6 shows the performance comparisons across different solvers. In general, high-order solvers are superior to basic ones, particularly at small NFE. Similar to the observations in Classifier Guidance, integrating MASF into these high-order solvers further improves FID scores, boosting generation quality. The results further validate the design of moving average in frequency domain in our MASF.

Table 6. FID comparisons of 50K samples with different solvers on ImageNet 128×128. * F-PNDM needs at least 12 NFE and is not applicable when NFE=10.

Method	NFE			
	10	15	20	25
DDPM	25.24	14.74	10.38	8.37
+MASF	**19.03**	**10.60**	**7.33**	**5.87**
DDIM	11.18	8.07	6.86	6.14
+MASF	**9.39**	**6.53**	**5.32**	**4.73**
DPM-Solver++(2M)	5.45	4.54	4.18	3.98
+MASF	**5.25**	**4.30**	**3.96**	**3.79**
UniPC	6.61	4.39	4.12	3.95
+MASF	**6.26**	**3.94**	**3.68**	**3.50**
F-PNDM	*	5.98	4.56	3.30
+MASF	*	**5.60**	**4.13**	**3.26**

5.3. Studies of MASF Designs

We perform ablation studies to examine each component's role in MASF. Moreover, we evaluate different γ values in moving average, various formulations of spatial weighting $\boldsymbol{w}_t$ and frequency weighting $\beta_f(t)$. In view that sampling 50K images is computationally expensive, we generate 10K ImageNet samples of resolution 128×128 with pre-trained model by ADM to do more ablations here.

Effect of Each Component. We first study how each particular design in MASF influences the overall performance for image generation. We degrade MASF by removing the frequency domain transformation, termed "+MA". Table 7 summarizes the FID improvements by considering one more component at each stage. The "+MA" variant applies moving average in pixel space to stabilize the denoising process to some extend, thereby outperforming the base model of DDIM. When further connecting generative sampling and frequency evolution, MASF manifests an apparent FID boost. The results basically prove the complementarity between moving average and frequency domain transformation.

Effect of γ in Moving Average. Next, we test the effect of γ in Eq. (10), which balances the contributions of the

Table 7. FID comparisons of 10K samples with different components on ImageNet 128×128.

Method	NFE			
	10	15	20	25
DDIM	23.00	17.97	15.73	14.70
+MA	21.88	17.48	15.42	14.58
+MASF	**21.76**	**16.11**	**13.70**	**12.80**

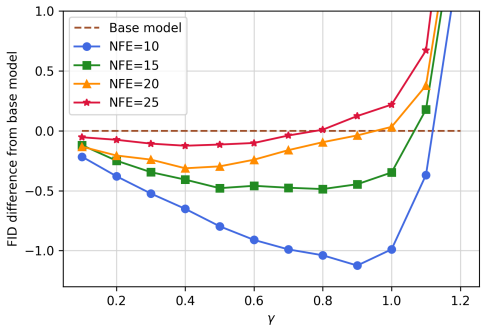


Figure 6. FID comparisons of 10K samples with different γ on ImageNet 128×128. The y-axis quantifies the FID improvements over the baseline model. Each curve corresponds to a distinct NFE.

moving average term and the current predicted sample. In general, a higher value of γ emphasizes more on the moving average term, resulting in a smoother trajectory. In contrast, a lower γ value prioritizes the current predicted sample, with $\gamma = 0$ downgrading to the base model. Figure 6 depicts the FID improvements over the base model through pixel space moving average. An observation is that when the values of γ range from 0.1 to 0.8, utilizing moving average consistently yields superior FID scores compared to the base model. Such trend verifies the model's robustness to the variations of γ values. Taking a closer look at the curves on different NFEs, a smaller NFE favors a larger γ, indicating a greater reliance on the moving average term. This observation elegantly corroborates our hypothesis that shorter sampling processes are more vulnerable to instability issues and benefit more from moving average.

Effect of $\boldsymbol{w}_t$ for Adaptive Weighting.

In order to analyze the impact of adaptive weight $\boldsymbol{w}_t$ in Eq. (10), we further compare different formulations of $\boldsymbol{w}_f^t$: Constant weight $\boldsymbol{w}_f^t = \mathbf{1}$, Linear weight $\boldsymbol{w}_f^t = |\boldsymbol{x}_f^t - \bar{\boldsymbol{x}}_f^{t+1}|$ and Quadratic weight $\boldsymbol{w}_f^t = |\boldsymbol{x}_f^t - \bar{\boldsymbol{x}}_f^{t+1}|^2$. The results in Table 8 indicate that employing Linear weight is superior to Constant weight. The observation supports the spirit behind that when the discrepancy between $\boldsymbol{x}_f^t$ and $\bar{\boldsymbol{x}}_f^{t+1}$ is large, relying more on moving average makes the denoising process more stable, therefore leading to lower FID. Given the fact that $|\boldsymbol{x}_f^t - \bar{\boldsymbol{x}}_f^{t+1}|$ is generally smaller than 1, Quadratic weight is smaller than Linear weight, and obtains inferior performances. We speculate that this might be the result of over-reliance on the current predicted sample, making the denoising process less stable.

Table 8. FID comparisons of 10K samples with different adaptive weight $\boldsymbol{w}_t$ on ImageNet 128×128.

Method	NFE			
	10	15	20	25
$\boldsymbol{w}_t = \mathbf{1}$	22.05	16.24	13.84	12.85
$\boldsymbol{w}_t = \lvert\boldsymbol{x}_0^t - \bar{\boldsymbol{x}}_0^{t+1}\rvert$	**21.76**	**16.11**	**13.70**	**12.80**
$\boldsymbol{w}_t = \lvert\boldsymbol{x}_0^t - \bar{\boldsymbol{x}}_0^{t+1}\rvert^2$	22.06	16.21	13.83	12.84

Table 9. FID comparisons of 10K samples with different frequency weight $\beta_f(t)$ on ImageNet 128×128.

Method	NFE			
	10	15	20	25
Low ↗	25.49	21.08	19.27	18.34
Low ↘	22.80	17.00	14.42	13.42
High ↘	27.18	20.43	20.43	19.63
High ↗	21.96	16.15	13.74	12.84
High ↗ + Low ↘	**21.76**	**16.11**	**13.70**	**12.80**

Effect of $\beta_f(t)$ for Frequency Weighting. To verify how the frequency weighting scheme $\beta_f(t)$ influences the denoising process, we detail the FID metric with different $\beta_f(t)$ variants. Based on the analysis in Section 4.3, frequency component weighting can follow two main trends: linear increase (↗) or decrease (↘) as denoising progresses. The performances are summarized in Table 9. We have $\beta_{ll}(t_{start}) = 1.03, \beta_{ll}(t_{end}) = 1, \beta_{h*}(t_{start}) = 1, \beta_{h*}(t_{end}) = 1.13$ for the last row in Table 9. Either decreasing the weight of low-frequency components (the second row) or increasing the weight of high-frequency components (the fourth row) in denoising process leads to notable FID improvements. Combining them together (the fifth row) yields the most favorable results. This aligns with our analysis of prioritizing low-frequency components in the early timesteps and gradually shifting focus to high-frequency components later in the process.

6. Conclusion

We have presented the Moving Average Sampling in Frequency domain (MASF), a new technique to enhance the stability of the diffusion process. MASF capitalizes on the moving average mechanism, effectively harnessing all previous samples. Moreover, MASF decomposes the sample into distinct frequency components, allowing for the dynamic evolution of each component during the denoising process. Extensive experiments validate that MASF significantly improves performance across various datasets, models, and sampling techniques. More remarkably, MASF introduces negligible computational overhead and can be readily integrated into existing diffusion models.

References

[1] Titas Anciukevičius, Zexiang Xu, Matthew Fisher, Paul Henderson, Hakan Bilen, Niloy J Mitra, and Paul Guerrero. Renderdiffusion: Image Diffusion for 3D Reconstruction, inpainting and generation. In *CVPR*, 2023. 1

[2] Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-DPM: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. In *ICLR*, 2022. 2

[3] 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 *CVPR*, 2023. 6

[4] John Charles Butcher. A history of Runge-Kutta methods. *Applied numerical mathematics*, 20(3):247–260, 1996. 2

[5] Jingwen Chen, Yingwei Pan, Ting Yao, and Tao Mei. Controlstyle: Text-driven stylized image generation using diffusion priors. In *ACM Multimedia*, 2023. 1

[6] Yang Chen, Yingwei Pan, Yehao Li, Ting Yao, and Tao Mei. Control3d: Towards controllable text-to-3d generation. In *ACM Multimedia*, 2023. 1

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 6

[8] Prafulla Dhariwal and Alexander Nichol. Diffusion Models Beat GANs on Image Synthesis. In *NeurIPS*, 2021. 1, 2, 5, 6, 7

[9] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and Content-Guided Video Synthesis with Diffusion Models. In *ICCV*, 2023. 1

[10] Ben Fei, Zhaoyang Lyu, Liang Pan, Junzhe Zhang, Weidong Yang, Tianyue Luo, Bo Zhang, and Bo Dai. Generative Diffusion Prior for Unified Image Restoration and Enhancement. In *CVPR*, 2023. 1

[11] Rinon Gal, Dana Cohen Hochberg, Amit Bermano, and Daniel Cohen-Or. SWAGAN: A style-based wavelet-driven generative model. *ACM Transactions on Graphics (TOG)*, 40(4):1–11, 2021. 2

[12] Martin Gonzalez, Nelson Fernandez, Thuy Tran, Elies Gherbi, Hatem Hajri, and Nader Masmoudi. SEEDS: Exponential SDE Solvers for Fast High-Quality Sampling from Diffusion Models. In *NeurIPS*, 2023. 2

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *NeurIPS*, 2014. 1

[14] Amara Graps. An Introduction to Wavelets. *IEEE computational science and engineering*, 1995. 2, 5

[15] Florentin Guth, Simon Coste, Valentin De Bortoli, and Stephane Mallat. Wavelet Score-Based Generative Modeling. In *NeurIPS*, 2022. 2

[16] 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. In *NeurIPS*, 2017. 6

[17] Jonathan Ho and Tim Salimans. Classifier-Free Diffusion Guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 2

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. In *NeurIPS*, 2020. 1, 3, 4

[19] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen Video: High Definition Video Generation with Diffusion Models. *arXiv preprint arXiv:2210.02303*, 2022. 1

[20] Alexia Jolicoeur-Martineau, Ke Li, Rémi Piché-Taillefer, Tal Kachman, and Ioannis Mitliagkas. Gotta Go Fast When Generating Data with Score-Based Models. *arXiv preprint arXiv:2105.14080*, 2021. 2

[21] Tero Karras, Samuli Laine, and Timo Aila. A Style-Based Generator Architecture for Generative Adversarial Networks. In *CVPR*, 2019. 1, 6

[22] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A Versatile Diffusion Model for Audio Synthesis. In *ICLR*, 2021. 1

[23] Haodong Li, Weiqi Luo, and Jiwu Huang. Localization of Diffusion-Based Inpainting in Digital Images. *IEEE transactions on information forensics and security*, 12(12):3050–3064, 2017. 1

[24] Jin Li, Wanyun Li, Zichen Xu, Yuhao Wang, and Qiegen Liu. Wavelet Transform-Assisted Adaptive Generative Modeling for Colorization. *IEEE Transactions on Multimedia*, 25:4547–4562, 2023. 2

[25] Shengmeng Li, Luping Liu, Zenghao Chai, Runnan Li, and Xu Tan. ERA-Solver: Error-Robust Adams Solver for Fast Sampling of Diffusion Probabilistic Models. *arXiv preprint arXiv:2301.12935*, 2023. 2

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 6

[27] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo Numerical Methods for Diffusion Models on Manifolds. In *ICLR*, 2022. 2, 4, 7

[28] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver++: Fast Solver for Guided Sampling of Diffusion Probabilistic Models. *arXiv preprint arXiv:2211.01095*, 2022. 1, 2, 4, 6, 7

[29] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver: A Fast ODE Solver for Diffusion Probabilistic Model Sampling in Around 10 Steps. In *NeurIPS*, 2022. 1, 2, 3

[30] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *CVPR*, 2022. 1

[31] Jianjie Luo, Yehao Li, Yingwei Pan, Ting Yao, Jianlin Feng, Hongyang Chao, and Tao Mei. Semantic-conditional diffusion networks for image captioning. In *CVPR*, 2023. 1

[32] Ziwei Luo, Fredrik K Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B Schön. Refusion: Enabling large-size realistic image restoration with latent-space diffusion models. In *CVPR*, 2023. 1

[33] S.G. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, 11(7):674–693, 1989. 2

[34] Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. To create what you tell: Generating videos from captions. In *ACM Multimedia*, 2017. 1

[35] Hao Phung, Quan Dao, and Anh Tran. Wavelet Diffusion Models Are Fast and Scalable Image Generators. In *CVPR*, 2023. 2

[36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 6

[37] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis With Latent Diffusion Models. In *CVPR*, 2022. 1

[38] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 2023. 2

[39] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. In *NeurIPS*, 2022. 2

[40] Flavio Schneider. Archisound: Audio Generation with Diffusion. *arXiv preprint arXiv:2301.13267*, 2023. 1

[41] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising Diffusion Implicit Models. In *ICLR*, 2021. 1, 2, 3, 4, 6

[42] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations. In *ICLR*, 2021. 2

[43] Radomir S Stankovic and Bogdan J Falkowski. The Haar wavelet transform: its status and achievements. *Computers & Electrical Engineering*, 29(1):25–44, 2003. 5

[44] Xiyu Wang, Anh-Dung Dinh, Daochang Liu, and Chang Xu. Boosting diffusion models with an adaptive momentum sampler. *arXiv preprint arXiv:2308.11941*, 2023. 2

[45] Daniel Raymond Wells. *Multirate linear multistep methods for the solution of systems of ordinary differential equations*. University of Illinois at Urbana-Champaign, 1982. 2

[46] Suttisak Wizadwongsa, Worameth Chinchuthakun, Pramook Khungurn, Amit Raj, and Supasorn Suwajanakorn. Diffusion Sampling with Momentum for Mitigating Divergence Artifacts. In *ICLR*, 2023. 2

[47] Zike Wu, Pan Zhou, Kenji Kawaguchi, and Hanwang Zhang. Fast Diffusion Model. *arXiv preprint arXiv:2306.06991*, 2023. 2

[48] Bin Xia, Yulun Zhang, Shiyin Wang, Yitong Wang, Xinglong Wu, Yapeng Tian, Wenming Yang, and Luc Van Gool. Diffir: Efficient diffusion model for image restoration. In *ICLR*, 2023. 1

[49] Haibo Yang, Yang Chen, Yingwei Pan, Ting Yao, Zhineng Chen, and Tao Mei. 3dstyle-diffusion: Pursuing fine-grained text-driven 3d stylization with 2d diffusion models. In *ACM Multimedia*, 2023. 1

[50] Mengping Yang, Zhe Wang, Ziqiu Chi, and Wenyi Feng. WaveGAN: An Frequency-aware GAN for High-Fidelity Few-shot Image Generation. In *ECCV*, 2022. 2

[51] Mengping Yang, Zhe Wang, Ziqiu Chi, and Yanbing Zhang. FreGAN: Exploiting Frequency Components for Training GANs under Limited Data. In *NeurIPS*, 2022. 2

[52] Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion Probabilistic Modeling for Video Generation. *Entropy*, 25(10):1469, 2023. 1

[53] Xingyi Yang, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Diffusion Probabilistic Model Made Slim. In *CVPR*, 2023. 2

[54] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. *arXiv preprint arXiv:1506.03365*, 2015. 6

[55] Xin Yuan, Linjie Li, Jianfeng Wang, Zhengyuan Yang, Kevin Lin, Zicheng Liu, and Lijuan Wang. Spatial-Frequency U-Net for Denoising Diffusion Probabilistic Models. *arXiv preprint arXiv:2307.14648*, 2023. 2

[56] Bowen Zhang, Shuyang Gu, Bo Zhang, Jianmin Bao, Dong Chen, Fang Wen, Yong Wang, and Baining Guo. StyleSwin: Transformer-Based GAN for High-Resolution Image Generation. In *CVPR*, 2022. 2

[57] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. 2

[58] Qinsheng Zhang and Yongxin Chen. Fast Sampling of Diffusion Models with Exponential Integrator. *arXiv preprint arXiv:2204.13902*, 2022. 2

[59] Qinsheng Zhang, Molei Tao, and Yongxin Chen. gDDIM: Generalized denoising diffusion implicit models. In *ICLR*, 2023. 2

[60] Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. UniPC: A Unified Predictor-Corrector Framework for Fast Sampling of Diffusion Models. In *NeurIPS*, 2023. 1, 2, 7