Title

MRI Super-Resolution with Partial Diffusion Models

Journal

IEEE Transactions on Medical Imaging, PP(99)

ISSN

0278-0062

Authors

Zhao, Kai
Pang, Kaifeng
Hung, Alex Ling Yu
et al.

Publication Date

2024-10-17

DOI

10.1109/tmi.2024.3483109

Peer reviewed

MRI Super-Resolution with Partial Diffusion Models

Kai Zhao, Kaifeng Pang, Alex Ling Yu Hung, Haoxin Zheng, Ran Yan, Kyunghyun Sung

***Abstract*— Diffusion models have achieved impressive performance on various image generation tasks, including image super-resolution. Despite their impressive performance, diffusion models suffer from high computational costs due to the large number of denoising steps. In this paper, we proposed a novel accelerated diffusion model, termed Partial Diffusion Models (PDMs), for magnetic resonance imaging (MRI) super-resolution. We observed that the latents of diffusing a pair of low- and high-resolution images gradually converge and become indistinguishable after a certain noise level. This inspires us to use certain low-resolution latent to approximate corresponding high-resolution latent. With the approximation, we can skip part of the diffusion and denoising steps, reducing the computation in training and inference. To mitigate the approximation error, we further introduced 'latent alignment' that gradually interpolates and approaches the high-resolution latents from the low-resolution latents. Partial diffusion models, in conjunction with latent alignment, essentially establish a new trajectory where the latents, unlike those in original diffusion models, gradually transition from low-resolution to high-resolution images. Experiments on three MRI datasets demonstrate that partial diffusion models achieve competetive super-resolution quality with significantly fewer denoising steps than original diffusion models. In addition, they can be incorporated with recent accelerated diffusion models to further enhance the efficiency.**

***Index Terms*— Generative models, Diffusion models, Score-matching, MRI, Super-resolution**

I. Introduction

Magnetic resonance imaging (MRI) is essential in clinical diagnosis because it provides structural and functional information without ionizing radiation. High-resolution MRI is generally desired for precise clinical diagnosis and analysis. However, acquiring high-resolution MRI is limited due to various constraints, such as scan time and hardware. MRI super-resolution (SR) is a promising technique that improves the resolution of MRI without increasing acquisition time or decreasing signal-noise ratio (SNR).

In recent years, deep learning has been widely used in image super-resolution and successfully applied to MRI [1], [2]. Deep learning models can be trained to map from low-resolution (LR) to high-resolution (HR) images from massive training data. Deep generative models, such as Generative Adversarial Networks (GANs) [3], have shown impressive results in image generation and have been applied to super-resolution [2], [4]. Though generating realistic images, GANs often suffer from instability in model optimization and model collapse. In addition, the inconsistency between these artificially generated details and their lower-resolution inputs can be detrimental to clinical decision-making. Recently, diffusion models [5]–[7] have shown remarkable performance on image generation tasks and have been applied to the super-resolution for both natural images [8], [9] and MRI [10], [11]. By modeling the reverse process of gradually diffusing the data distribution into Gaussian noise, diffusion models generate new images by iterative denoising from random Gaussian noise. Studies have shown that diffusion-based SR methods exhibit superior performance and enhanced consistency [8].

Unlike unconditional image generation, where models are expected to generate samples from pure noise, image super-resolution involves generating a high-resolution output from a low-resolution input image. The low-resolution input exhibits a similar structure, content, and appearance as a high-resolution image, except for high-frequency details. This unique feature raises an interesting question: *is it necessary to denoise from pure noise when applying diffusion models to image super-resolution?*

To answer this question, we first investigate the diffusion processes of low- and high-resolution image pairs. As shown in Fig. 1, we found that the latents of low- and high-resolution images gradually converge and become almost indistinguishable after a certain noise level. Given the observation that the latents are similar, potentially we can use the low-resolution latent at a certain noise level to approximate the corresponding high-resolution latent. This motivates us to propose a novel diffusion model that only executes part of the denoising steps. As illustrated in Fig. 1, our method uses a low-resolution latent (x_K^{LR}) to approximate the high-resolution latent (x_K^{HR}). This allows us to bypass denoising steps between T to K and only execute the denoising steps from K to 0. Our method, termed partial diffusion models (PDMs), can accelerate diffusion models by reducing the number of denoising steps.

Although the two latents are visually similar, there is still a statistical disparity which will inevitably have a detrimental effect on the quality of generation, especially when the approximation is made at a lower noise level (smaller K values). We introduced 'latent alignment' to mitigate the approximation error and enhance the quality of generation, which progres-

This research was funded in part by the National Institutes of Health under grants R01-CA248506 and R01-CA272702, and by the Integrated Diagnostics Program of the Departments of Radiological Sciences and Pathology in the UCLA David Geffen School of Medicine.

K Zhao, K Pang, A Hung, H Zheng, R Yan, and K Sung are with the Department of Radiological Sciences, David Geffen School of Medicine, University of California, Los Angeles, CA 90045, USA. kz@kaizhao.net, alexhung96@ucla.edu, {kaifengpang, hzheng, ranyan, ksung}@mednet.ucla.edu.

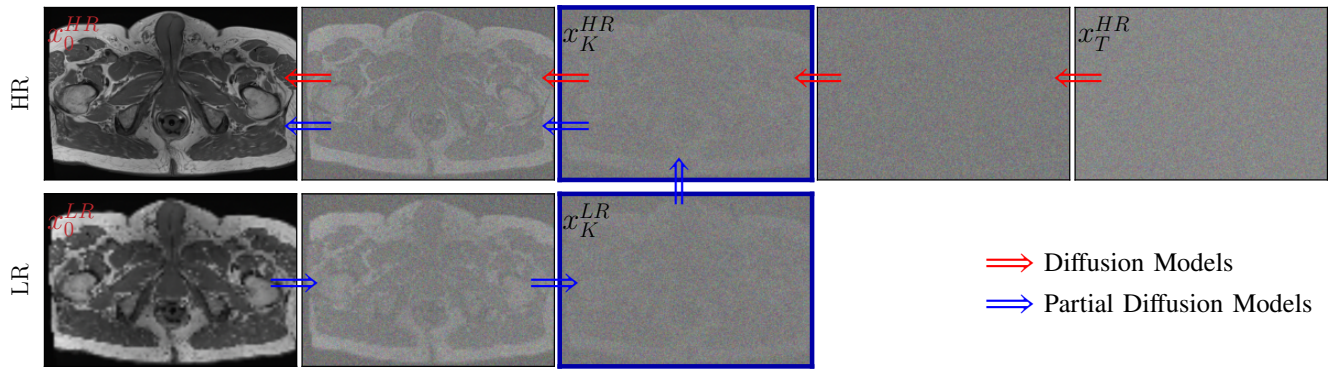


Fig. 1: The low- and high-resolution latents and the denoising trajectories of diffusion (top red) and partial diffusion (bottom blue) models. The low- and high-resolution latents gradually converge and become indistinguishable after certain noise levels (X_K with blue frames), making it possible to use x_K^{LR} to approximate x_K^{HR} to reduce the number of denoising steps.

sively aligns the low- and high-resolution latents. Concretely, we interpolate between the low- and high-resolution latents, gradually transitioning from low resolution to high resolution. Latent alignment essentially establishes a new denoising (diffusion) trajectory that smoothly transitions from low (high) resolution to high (low) resolution images, thus avoiding abrupt discontinuities caused by the approximation.

Extensive experiments on three different MRI datasets demonstrated that: 1) partial diffusion models are able to achieve the same or very similar image quality with few denoising steps. 2) with the same number of denoising steps, partial diffusion models achieve better image quality, and 3) partial diffusion models can incorporate with recent accelerated diffusion models to further improve the efficiency.In summary, the contributions of this paper are in three folds:

- We qualitatively and quantitatively observed that the diffusion processes of low- and high-resolution images gradually converge midway and the latents become indistinguishable.
- With the observation, we are motivated to use the latents of low-resolution images to approximate that of the high-resolution images. This allows us to accelerate diffusion models in training and testing by skipping part of the denoising steps.
- We proposed 'latent alignment' that establishes a new trajectory whose latents gradually transition from low-resolution to high-resolution images to mitigate the approximation error and improve the quality of generation.

The rest of this paper is organized as follows: Sec. II summarizes the related works in image super-resolution and diffusion probabilistic models. Sec. III introduces the background of the diffusion models. Sec. IV elaborates on the proposed partial diffusion models (PDMs) and discusses the key components. Sec. V presents experimental details and reports comparison results. Sec. VI makes a conclusion remark.

II. Related Work

Our method is inspired by recent works in generative models, especially diffusion models [5], [6], [12], for image super-resolution. In this section, we first briefly introduce related works in image super-resolution with a focus on deep learning-based methods and then cover some recent advances in diffusion models for image super-resolution.

A. Image Super-resolution

Early methods for image super-resolution employ different types of priors, *e.g.* edges [13], gradient [14], [15] and sparsity [16], to recover high-frequency image details. With the rapid development of deep learning techniques, a line of deep learning-based super-resolution approaches has been proposed, achieving appealing results. Many of the early explorations directly use convolutional neural networks (CNNs) to regress a high-resolution image [17]–[20] based on a low-resolution input. Many new architectures [18], [19], [21] and loss functions [20], [22] have been proposed to improve the quality of super-resolution. Although being able to generate images close to the ground-truth, regression-based methods tend to produce blurry images that correlate poorly to human perception. Deep generative models, e.g., generative adversarial networks (GANs), have shown impressive performance in generating high-fidelity realistic images and benefited conditional tasks such as image super-resolution [4], [23]. Many studies improve GAN-based image super-resolution in network architecture [4], [23], training strategies [4], and domain-specific priors [24]. Although GANs provide a promising direction, they generally suffer from common failure cases of mode collapse and unstable training [25].

Recently, Saharia et al. [8] and Li et al. [9] adapted diffusion models for natural image super-resolution and have shown their outstanding performance in generating realistic high-resolution images. Hung et al. [10] applied diffusion models to conditional medical image generation tasks, including MRI super-resolution. Chung et. al. [11] used diffusion models to reconstruct high-resolution MR images from low-resolution

measurements.However, one deficiency of diffusion models is its tedious generation process, which includes thousands of denoising steps.

B. Diffusion Probabilistic Models and Acceleration

Diffusion probabilistic models [26] are a class of generative models that match a data distribution by learning to reverse a gradual noising process. Diffusion models have received growing attention in recent years due to their promising results in generating high perceptual images [5], [6]. Diffusion models have also shown impressive results in condition image generation tasks such as image super-resolution [8], [9], [11], [27]. SR3 [8] takes the low-resolution image as an additional input to the denoising network and sets up a conditional denoising framework. SRDiff [9] also uses the low-resolution image as the condition but executes the diffusion process in a lower-dimensional hidden space. MedDiff [10] applied diffusion models to various conditional image generation tasks in medical imaging including MRI and CT image super-resolution. Chung et. al. [11] and Xie et. al. [27] apply diffusion models to the inverse problem of MRI reconstruction from undersampled measurements.These methods have shown impressive super-resolution results in restoring high-fidelity details, especially under large upsampling factors.

While achieving appealing performance in image generation, diffusion models are notoriously slow in inference because generating high-quality samples generally needs hundreds or thousands of sequential denoising steps [6]. A line of studies has been proposed to accelerate diffusion models for image generation, such as the fast diffusion probabilistic model solver [28], Denoising Diffusion Implicit Models (DDIM) [29], and Consistency Models (CM) [30]. SkipDiff [31] selectively skips some denoising steps using reinforcement learning. Another work [32] predicts an intermediate state using the low-resolution input and then starts the denoising process from the intermediate to skip denoising steps. However, this approach requires a separate neural network to predict the intermediate state where denoising begins, which limits its practical utility.Lu et al. [28] propose an exact formulation that analytically computes the linear part of the solution to diffusion ordinary differentiable equations (ODEs). Chen et al. [33] condition denoising models on the continuous noise scales instead of discrete denoising steps t, such that separate noise schedules can be used in training and testing, allowing flexible adjustment of denoising steps in inference. This method requires carefully tuned noise schedules in testing, and the resulting noise schedule is unstable in different datasets. Model distillation was also introduced to reduce the denoising steps of diffusion models [34]. More recently, Zheng et al. [35] propose to add noise not until the data becomes pure random noise, but until they reach a hidden noisy data distribution that can be confidently learned. Consequently, few denoising steps are required to generate data from the hidden noise distribution.

Different from these general-purpose diffusion model acceleration methods, our proposed method is dedicated to image super-resolution. Our model does not require any additional computation, and can reduce a large proportion of the denoising steps while achieving competitive image quality. It can be incorporated with other accelerated diffusion models to further improve efficiency.

III. Background on Diffusion Models

We introduce some background of diffusion models and their applications to super-resolution. We adopt the notation of DDPMs [6] for both unconditional image generation and conditional image generation for image super-resolution.

A. Denoising Diffusion Probabilistic Models

Diffusion models transform data samples x_0 into Gaussian noise x_T through a gradual noising process and generate new data by learning to reverse this process. The transition from data to noise is referred to as the forward process (or diffusion process), and the opposite is called the reverse process (or denoising process).

1) Forward process: The forward process transforms data into Gaussian noise by iteratively adding Gaussian noise to a clean sample x_0. This can be formulated as a Markov process with pre-defined Gaussian transitions:

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}), \tag{1}$$

where

$$q(x_t|x_{t-1}) := \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}) \tag{2}$$

is the forward Gaussian transition with variances β_t. Ideally, with sufficiently large T and well-behaved β_t, x_T is nearly an isotropic Gaussian distribution. As noted by Ho *et al.* [6], the diffusion process defined in Eq. (1) allows us to sample arbitrary steps of the latent step x_t conditioned on the input x_0:

$$\begin{aligned} q(x_t|x_0) &= \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\mathbf{I}) \\ \alpha_t &:= 1-\beta_t, \quad \bar{\alpha}_t = \prod_{\tau=1}^{t} \alpha_\tau. \end{aligned} \tag{3}$$

$\sqrt{\bar{\alpha}_t}$ is also called the 'noise scale' of x_t. Furthermore, following DDPM [6], we can derive the posterior distribution of x_{t-1} given x_T and x_0:

$$q(x_{t-1}|x_t, x_0) = \mathcal{N}\left(x_{t-1}; \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t\mathbf{I}\right), \tag{4}$$

$$\tilde{\mu}_t(x_t, x_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t \tag{5}$$

$$\tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t. \tag{6}$$

The forward posterior in Eq. (4) will be compared with the learned reverse posterior during training.

2) Reverse process: The reverse process (denoising process) learns to recover the original sample x_0 $x_T \sim \mathcal{N}(0, \mathbf{I})$. This process is formulated as a Markov process with learned transitions:

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t), \tag{7}$$

where

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}\big(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)\big) \tag{8}$$

is the reverse Gaussian transition with learned mean $\mu_\theta(x_t, t)$ and variance $\Sigma_\theta(x_t, t)$. Note that the variance $\Sigma_\theta(x_t, t)$ can be either a time-dependent constant or learned by a neural network [6], and the mean $\mu_\theta(x_t, t)$ is parameterized by a neural network. The reverse process transforms the standard Gaussian distribution $x^T \sim \mathcal{N}(0, \mathbf{I})$ into data distribution $p(x_0)$.

With learned transition distribution p_θ, to generate a new image from the reverse process, we first sample x_T from standard Gaussian distribution, and then sample $\hat{x}_{t-1}$ from $p_\theta(x_{t-1}|x_t)$ for $t = T, T-1, ..., 1$. $\hat{x}_0$ is the data generated from DDPMs.

Data generation of DDPMs is extremely time-consuming because it involves hundreds or even thousands of evaluations of the neural network parameterizing transition p_θ. Therefore, It is necessary to accelerate the sampling process of the DDPM for practical utilization.

3) Optimization: Like other latent variable generative models, such as VAE [36], training DDPMs is performed by optimizing the evidence lower bound (ELBO) on negative log-likelihood [6]:

$$\begin{aligned} \mathcal{L} &= \mathbb{E}_q \log \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)} \\ &= L_0 + \sum_{t>1} L_t + L_T, \end{aligned} \tag{9}$$

where

$$\begin{aligned} L_0 &= -\log\big(p_\theta(x_0|x_1)\big) \\ L_t &= D_{KL}\big(q(x_{t-1}|x_t, x_0) \;||\; p_\theta(x_{t-1}|x_t)\big) \\ L_T &= D_{KL}\big(q(x_T|x_0) \;||\; P(x_T)\big). \end{aligned} \tag{10}$$

L_0 can be evaluated with the histogram of image pixel values. L_T is independent of θ and will ideally be zero with adequate diffusion steps N and proper noise schedule $\{\beta_1, \beta_2, ..., \beta_t\}$.

L_{t-1} in Eq. (9) compares the KL-divergence between estimated posterior $p_\theta(x_{t-1}|x_t)$ and forward posterior in Eq. (4), which can be analytically expressed. Ho et al. [6] suggest to fix the variance Σ_θ to a constant value, e.g. $\Sigma_\theta = \beta_t \mathbf{I}$. After ignoring all constant variables independent of θ, L_{t-1} can be simplified as:

$$L_{t-1} = \mathbb{E}_q\Big[\frac{(\tilde{\mu}_t - \mu_\theta)^2}{2\sigma^2}\Big]. \tag{11}$$

Instead of directly predicting μ_θ using the neural network, Ho et al. [6] suggest to predict the noise ϵ and the estimated mean can be derived through:

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\big(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t)\big), \tag{12}$$

where ϵ is the random noise added in the forward process, and $\epsilon_\theta(x_t)$ is the model prediction. Under this parameterization, the objective L_{t-1} becomes

$$L_{t-1} = \mathbb{E}_{\epsilon,t}\big[\|\epsilon_\theta(x_t) - \epsilon\|\big] \tag{13}$$

Ho et al. [6] found that predicting ϵ works the best, and the estimated noise ϵ_θ is the gradient of the data density. This connects DDPM with score-based generative models and Langevin dynamics [5].

After training, the model output $\epsilon_\theta(x_t)$ matches the gradient of the log probabilistic density $\nabla_x \log p(x)$ (or the Stein score function) almost everywhere and to sample $p(x_{t-1}|x_t)$ is to compute

$$x_{t-1} = \frac{1}{\sqrt{\bar{\alpha}_t}}\big(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t)\big) + \beta_t \mathbf{I} \cdot \mathbf{z} \tag{14}$$

where $\beta_t \mathbf{I}$ is the time-dependent constant variation and $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$.

B. Diffusion Models Conditioned on Noise Level

In the original DDPM [6], the noise schedule $\{\beta_1, ..., \beta_T\}$ and the number of diffusion (or denoising) steps N have to be carefully tuned to ensure high-quality data generation. The noise schedule is typically determined by hyper-parameter heuristics, e.g., linear [6]. To generate high-quality images at high resolution, N must also be large enough. For example, Ho et al. [6] use $N = 1,000$ to sample 256×256 images.

Instead of conditioning on discrete step index t, Chen et al. [33] reparameterize the model to condition on continuous noise level $\bar{\alpha}_t$. This allows separate noise schedules $\{\beta_t\}_{t=1}^T$ and the number of iterative steps N in training and testing. The network ϵ is conditioned on noise scale, and the objective in Eq. (13) becomes $\epsilon_\theta(x_t, x_0, \bar{\alpha}_t)$.

C. Conditional DDPMs for Image SR

In image super-resolution, we are given a dataset of pairwise low- and high-resolution images $\mathcal{D} = \{(I_{lr}, I_{hr})_1, ..., (I_{lr}, I_{hr})_N\}$, and are expected to learn the distribution of I_{hr} conditioned on I_{lr}: $p(I_{hr}|I_{lr})$. Based on the distribution, we can sample a super-resolution image I_{sr} conditioned on low-resolution input.

Two recent works SR3 [8] and SRDiff [9] approach this problem by adapting the DDPMs to conditional image generation. The basic idea is to use the low-resolution image as the condition in the DDPM image generation framework. Under this setting, the reverse process of conditional DDPMs is:

$$p_\theta(x_{0:T}|I_{lr}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t, I_{lr}). \tag{15}$$

where the reverse transition $p_\theta(x_{t-1}|x_t, I_{lr})$ conditions not only on denoising step t, but also on the low-resolution image I_{lr}. Similarly, to generate super-resolution image I_{sr} from I_{lr}, we first sample x_T from Gaussian distribution and then iteratively sample from $p_\theta(x_{t-1}|x_t, I_{lr})$ for $t = T, T-1, ..., 1$ until we get $I_{sr} = x_0$.

IV. Partial Diffusion Models for Image SR

In this section, we introduce partial diffusion models (PDMs) for MRI super-resolution. We first compare the diffusion process of low- and high-resolution images in Sec. IV-A, and then introduce the two key components of our proposed method, partial diffusion and latent alignment, in Sec. IV-B and Sec. IV-C, respectively.

A. Diffusing LR and HR Images

We first compare the diffusion process of low- and high-resolution images. Let $p(x^{LR}_{1:T}|x^{LR}_0)$ and $p(x^{HR}_{1:T}|x^{HR}_0)$ be the forward processes of low- and high-resolution image pairs, and x^{LR}_t and x^{HR}_t are the latents. The two processes start from different distributions, i.e., low- and high-resolution images, but end up with the same isotropic Gaussian distribution, i.e. $x^{LR}_T, x^{HR}_T \in \mathcal{N}(0, \mathbf{1})$. We hypothesize that the two processes converge at the midway, and x^{HR}_t and x^{LR}_t become indistinguishable after certain noise level. We verified our hypothesis qualitatively and quantitatively.

First, we visualize the diffusion processes of low- and high-resolution images. As shown in Fig. 1, the diffused images become visually indistinguishable after several diffusion steps.

Second, we quantitatively measured the KL-divergence between x^{LR}_t and x^{HR}_t. Specifically, we first diffused the low-resolution and high-resolution image pairs of ProstateX [37] dataset to get the latents x^{LR}_t and x^{HR}_t. Then, we calculated the histogram of each latent at different time steps t. The number of bins used for histogram calculation was set to 256. We then calculated the average KL-divergence between the histograms of low- and high-resolution latents. We tested two different downsampling factors: ×2 and ×4.

The results in Fig. 2 demonstrate that, as expected, the low- and high-resolution latents gradually converge, with the KL-divergence nearing zero at approximately one-quarter of the denoising steps. The statistics in Fig. 2 suggest that we can roughly approximate x^{HR}_t with x^{LR}_t at a t value greater than one-quarter of the denoising steps. And the approximation becomes more accurate if t is equal to or larger than half of the denoising steps.

B. Partial Diffusion Models

Based on the analysis in Sec. IV-A, we propose the partial diffusion Models (PDM) which execute only part of the diffusion and denoising steps by approximating x^{HR}_K with x^{LR}_K, where $K < T$ is an intermediate step, after which $x_K LR$ and $x_K HR$ become indistinguishable. PDMs accelerate the diffusion models by skipping all steps with $t \geq K$.

In particular, in training, we only train the reverse Gaussian transition $p_\theta(x_{t-1}|x_t)$ for $t = 1, 2, ..., K$ and all steps after K are skipped. During generation, given the low-resolution image $L_{lr} := x^{LR}_0$ as input, we first diffuse it by K steps to derive x^{LR}_K which can be analytically evaluated using Eq. (3), and then use x^{LR}_K as the proxy to x^{HR}_K and start denoising from x^{HR}_K until reach $x^{HR}_0 := I_{sr}$, which is the generated high-resolution image. In other words, PDMs skipped diffusion and denoising steps for $t \geq K$ for both training and testing.

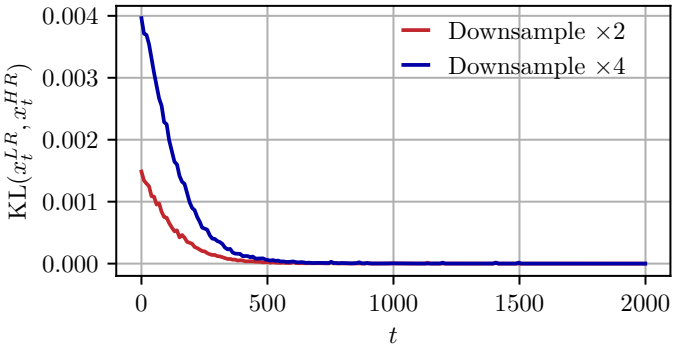


Fig. 2: KL-divergence between low-resolution latents (x^{LR}_t, downsampled by factors of ×2 and ×4) and high-resolution latents (x^{HR}_t) across denoising steps t. The latents gradually converge and the KL-divergence approaches zero at approximately one-quarter of the denoising steps.

It's worth noting that the approximation may lead to suboptimal results due to the disparity between X^{HR}_K and X^{LR}_K. As shown in Fig. 3 (b), during training, the denoising trajectory is $X^{HR}_K \rightarrow X^{HR}_{K-1} ... \rightarrow X^{HR}_{K-2}$. In testing, the trajectory becomes $X^{LR}_K \rightarrow X^{HR}_{K-1} ... \rightarrow X^{HR}_{K-2}$. The approximation error becomes more severe when the upsampling factor is large or when more steps are skipped. In Sec. IV-C, we will elaborate on how we mitigate the approximation error and establish a unified denoising trajectory with latent alignment.

C. Latent Alignment

The partial diffusion introduced in Sec. IV-B can be seamlessly integrated into any pretrained diffusion model. This can be achieved by approximating the high-resolution latent with a low-resolution latent, as shown in Fig. 3 (b). The subtle approximation error between x^{LR}_K and x^{HR}_K may cause a slight degradation in generation quality. The disparity is even more noticeable with larger upsampling factors or when more steps are skipped (smaller K).

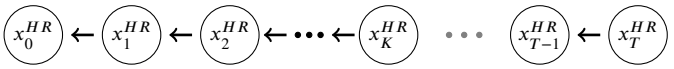
We propose the 'latent alignment' to mitigate the approximation error and unify the trajectory in training and inference. Latent alignment gradually interpolates between the low- and high-resolution latents so that the denoising trajectory gradually starts from X^{LR}_K and gradually approaches X^{HR}_0.

As shown in Fig. 3 (c), latent alignment essentially establishes a new diffusion (denoising) trajectory: between x^{HR}_0 and x^{LR}_K. The denoising model learns to recover high-resolution image x^{HR}_0 from the low-resolution latent x^{LR}_K.

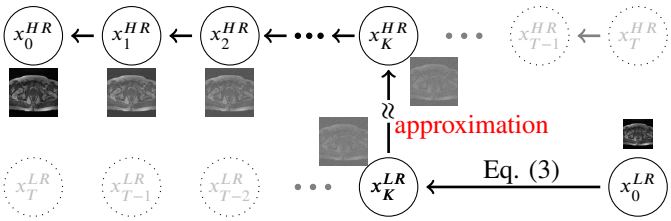
For each training iteration, we first randomly sample a step-index $t \in (0, K]$ and diffuse a pair of low- and high-resolution images to derive x^{LR}_t and x^{HR}_t according to Eq. (3). The latent partial diffusion models with latent alignment are defined as:

$$q(\hat{x}_t|x^{LR}_0, x^{HR}_0) = \mathcal{N}\Big(\hat{x}_t; \sqrt{\bar{\alpha}_t}\big(\lambda_t x^{HR}_0 + (1-\lambda_t)x^{LR}_0\big), (1-\bar{\alpha}_t)\mathbf{I}\Big), \tag{16}$$

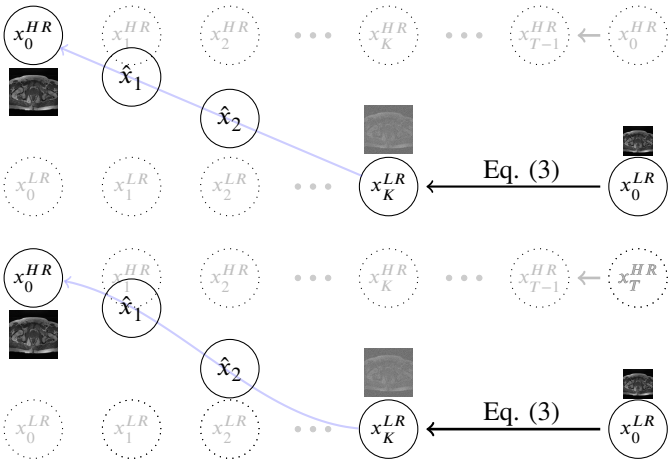
where $\hat{x}_t$ is the latent of the new trajectory, and λ_t is the predefined interpolation weights. λ_t ensures that $\hat{x}_0 = x^{HR}_0$, $\hat{x}_K = x^{LR}_K$, and for $0 < t < K$, $\hat{x}_t$ monotonically approaches



(a) Diffusion models denoise from x_T^{HR} to x_0^{HR} over T steps.



(b) Trajectories of partial diffusion models without latent alignment. During training, the model learns to denoise from x_K^{HR} to x_0^{HR}. During inference, the denoising trajectory follows $x_K^{LR} \rightarrow x_{K-1}^{HR} \rightarrow ... \rightarrow x_0^{HR}$, reduces the number of steps from T to K. However, the disparity between x_K^{HR} and x_K^{LR} leads to suboptimal results.



(c) Partial diffusion with latent alignment using linear (top) and cosine (bottom) schedules. Both training and inference share the same trajectory. The latents $\hat{x}_t$ interpolate between x_t^{HR} and x_t^{LR}, smoothly transitioning from x^{LR} to x^{HR} without abrupt changes.

Fig. 3: The denoising trajectories of diffusion models (a), partial diffusion models with (c) and without (b) latent alignment.

x_0^{HR} from x_K^{LR}. We propose two λ_t schedules, the linear and the cosine schedules, in Eq. (17).

$$\lambda_t = \begin{cases} 1 - \frac{t}{K}, (a) \\ 0.5 \cdot \left(\cos \frac{t}{K}\pi + 1 \right), . \end{cases} \quad t \in \{0, ..., K\} \tag{17}$$

The two schedules are illustrated in Fig. 3 (c). By default, we used the cosine schedule unless stated otherwise, and we compared the two schedules in Tab. VII of Sec. V-F.

Similar to Eq. (4), the posterior of $\hat{x}_{t-1}$ in PDMs becomes:

$$q(\hat{x}_{t-1}|x_t, x_0^{LR}, x_0^{HR}) = \mathcal{N}\left(\hat{x}_{t-1}; \hat{\mu}_t(\hat{x}_t, \hat{x}_0), \hat{\beta}_t \mathbf{I}\right), \tag{18}$$

where

$$\begin{aligned} \hat{\mu}_t(x_t, x_0) &= \lambda_t \tilde{\mu}_t(x_t^{HR}, x_0^{HR}) + (1 - \lambda_t)\tilde{\mu}_t(x_t^{LR}, x_0^{LR}) \\ \hat{\beta}_t &= \tilde{\beta}_t. \end{aligned} \tag{19}$$

and $\tilde{\mu}(\cdot, \cdot)$ and $\hat{\beta}_t$ are the mean and variance of forward posterior defined in Eq. (5) and Eq. (6), respectively. During training, the forward posterior in Eq. (18) is used as the target to guide the denoising model. The loss term L_{t-1} in Eq. (9) becomes

$$L_{t-1} = D_{KL}\Big(q(\hat{x}_{t-1}|x_t, x_0^{LR}, x_0^{HR}) \;||\; p_\theta(x_{t-1}|x_t)\Big).$$

The model learns to denoise along the newly estimated trajectory that approaches X_0^{HR} from X_K^{LR}, as illustrated in Fig. 3 (c).

V. Experiments

In this section, we introduce the implementation details and report the experimental results.

A. Implementation details

We applied partial diffusion to various diffusion model variants. All the models are implemented with the PyTorch [38] framework. All hyper-parameters, unless otherwise specified, are identical to those in the original papers for fair comparisons.

1) Data and Data Preprocessing: We test our method on three multi-slice MRI datasets: i) our in-house prostate MRI dataset, ii) the ProstateX dataset [37], and iii) the Knee MRI from the FastMRI [39] dataset.

We use T2-weighted images from ProstateX and our in-house datasets, and proton density-weighted (PD) images from the FastMRI [39] knee scans. Detailed information, including the number of training/testing patients and images, are summarized in Tab. I. All the images are real-valued and in dicom format.

Let $(h \times w \times d)$ be the shape of a multiple-slice 2D scan where $(h \times w)$ is the in-plane resolution and d(epth) is the number of slices. We perform super-resolution on two different settings: 1) in-plane super-resolution that improves the resolution in the $(h \times w)$ plane and 2) through-plane super-resolution that improves the resolution in the d-axis.

For in-plane super-resolution, we downsample high-resolution images using K-space zero padding (KSZP) to simulate the under-sampled MR images. We used a KSZP-downsampled image as the low-resolution image, and the model estimates high-resolution images from low-resolution images as input.

2) Models and Training Details: We compared our method with three diffusion-based super-resolution methods including SR3 [8], ScoreMRI [12], and MC-DDPM [27]. SR3 is a general image super-resolution method proposed for natural image, and ScoreMRI and MC-DDPM were proposed for MRI reconstruction. MC-DDPM was designed to take under-sampled K-space data as input. We adapted it to accept under-sampled image data as input. For ScoreMRI, we utilized its single-coil real-valued configuration to recover high-resolution images from low-resolution undersampled images. We trained both methods using their original settings for fair comparisons. By default, we used the cosine λ schedule (Eq. (17) bottom) unless otherwise specified.

Our method is compatible with and applicable to several other diffusion model acceleration techniques. We applied partial diffusion to three accelerated diffusion models,

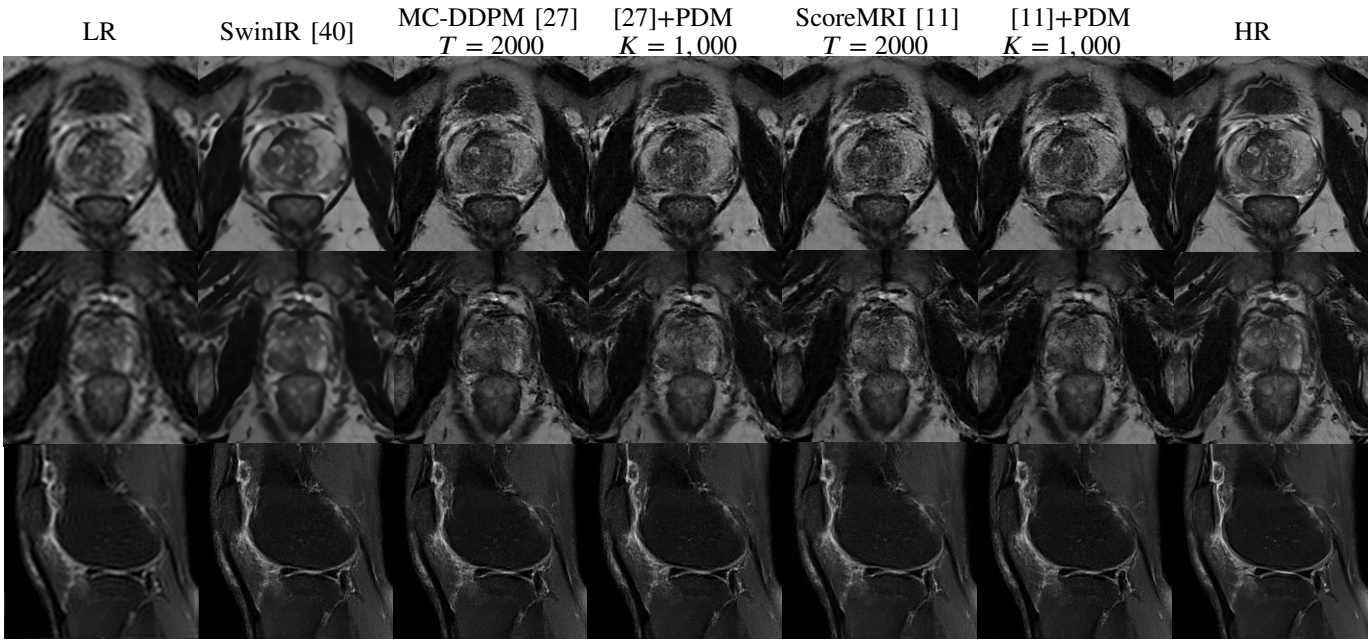


Fig. 4: Super-resolution results (×4) of MC-DDPM, ScoreMRI, and PDMs on the ProstateX (top), our in-house prostate MRI (middle), and the FastMRI knee MRI datasets. PDMs reduce the denoising steps of MC-DDPM and ScoreMRI from $T = 2000$ to $K = 1000$, while achieving competitive results.

TABLE I: Summary of the three datasets used in our experiments.

Datasets	ProstateX [37]	fastMRI [39]	Clinical Prostate
Region	prostate	knee	prostate
Sequence	T2-TSE	T2-PD	T2-TSE
Plane	axial+coronal	sagittal	axial+coronal
#train	206 (8,826)	600 (20,015)	636 (12,542)
#test	142 (12,881)	200 (6,611)	200 (6,271)

DDIM [29], DPM-Solver [28], and Consistency Models [30] to further reduce the number of iterations in both training and testing. We also compared with several non-diffusion-based super-resolution models, including SRGAN [4], EDSR [41], SwinIR [40], and EDSR [41].

All the models were trained on a server with 4 NVIDIA RTX 8000 GPUs. We kept the training recipes of the original papers for fair comparisons except that all the diffusion models were trained for two million iterations and a batch size of eight.

3) Denoising Steps: When experimenting with diffusion-based methods, we strictly follow the recipe of the original papers about the number of diffusion and denoising steps. For MC-DDPM and ScoreMRI, we set $T = 2,000$, while for SR3, we set $T = 100$ by default. We employed PDMs to reduce the denoising steps to one-quarter and one-half of the original methods. For example, with SR3, PDMs reduce the original denoising steps from $T = 100$ to $K = 25$ and $K = 50$. To compare PDMs with the original methods using the same denoising steps, we also train and test original methods using one-quarter and one-half of the denoising steps. For example, we trained ScoreMRI with $T = 2,000$, $T = 1,000$ and $T = 500$.

The denoising steps K in PDMs were determined according to Fig. 2, which shows that the KL-divergence between low- and high-resolution latents becomes nearly zero after one-quarter of the steps and remains stably close to zero after about half of the steps. We also conducted experiments to evaluate the performance with various K in Sec. V-F.2.

B. In-plane MRI Super-resolution

We first report experimental results on in-plane MRI super-resolution. We tested two different super-resolution scales: ×2 ($160 \times 160 \rightarrow 320 \times 320$) and ×4 ($80 \times 80 \rightarrow 320 \times 320$).

1) Qualitative comparisons: Fig. 4 displays some ×4 super-resolution results of SwinIR, MC-DDPM, and ScoreMRI. We applied partial diffusion to MC-DDPM and ScoreMRI to reduce the denoising steps from $T = 2000$ to $K = 1000$. The super-resolution factor is ×4 where images of dimensions 80×80 are enhanced to 320×320. The first column represents the low-resolution input, the last column shows the high-resolution reference, and the middle columns showcase the super-resolution outcomes achieved through various methods.

As shown in Fig. 4, diffusion-based methods, MC-DDPM and ScoreMRI, generate more realistic details than the state-of-the-art non-diffusion SR method, SwinIR. Partial diffusion models achieve very similar results with the original methods with significantly less denoising steps.

2) Quantitative performance: We quantitatively assess the super-resolution image quality of various super-resolution methods, including diffusion-based approaches such as MC-DDPM [27] and ScoreMRI [11], as well as other deep learning-based methods like SRGAN [4], EDSR [41], SwinIR [40], and SMORE [1]. The performance of SR is quantified in terms of three metrics: i) Structural Similarity (SSIM), ii) Peak Signal Noise Ratio (PSNR), and iii) Consistency (Consist). Following the practice of [8], the consistency

TABLE II: PSNR, SSIM, and Consistency on the ProstateX dataset. T denotes the number of steps of original diffusion models, and K is the number of steps in PDMs. In each group, the same color indicates methods with the same number of inference steps.

	Method	T	K	PSNR	SSIM	Consist
×2	Bicubic			30.62	0.8568	38.56
	EDSR [41]			32.59	0.8714	46.48
	SwinIR [40]			32.82	0.8811	46.14
	SRGAN [4]			31.25	0.8579	40.55
	SMORE [1]			32.65	0.8501	39.79
	ScoreMRI [11]	2000		34.12	0.9113	47.09
	ScoreMRI [11]	1000		34.00	0.9102	47.02
	ScoreMRI [11]	500		33.81	0.9037	46.87
	ScoreMRI+PDM	2000	1000	34.12	0.9112	47.09
	ScoreMRI+PDM	2000	500	33.97	0.9092	47.01
	MC-DDPM [27]	2000		34.09	0.9109	47.00
	MC-DDPM [27]	1000		33.92	0.9081	46.93
	MC-DDPM [27]	500		33.76	0.9025	46.19
	MC-DDPM+PDM	2000	1000	34.04	0.9109	47.00
	MC-DDPM+PDM	2000	500	33.92	0.9100	46.91
	SR3 [8]	100		34.02	0.9098	46.86
	SR3 [8]	50		33.81	0.9017	46.83
	SR3 [8]	25		33.19	0.8849	46.11
	SR3+PDM	100	50	34.01	0.9096	46.86
	SR3+PDM	100	25	33.98	0.9079	46.79
×4	Bicubic			25.75	0.6589	36.27
	EDSR [41]			27.45	0.7529	38.57
	SwinIR [40]			27.89	0.7535	38.23
	SRGAN [4]			24.80	0.6756	32.45
	SMORE [1]			25.29	0.6765	33.12
	ScoreMRI [11]	2000		28.48	0.7665	39.81
	ScoreMRI [11]	1000		28.09	0.7654	39.68
	ScoreMRI [11]	500		27.47	0.7608	39.11
	ScoreMRI+PDM	2000	1000	28.46	0.7664	39.80
	ScoreMRI+PDM	2000	500	28.02	0.7649	39.57
	MC-DDPM [27]	2000		28.46	0.7662	39.79
	MC-DDPM [27]	1000		28.01	0.7653	39.65
	MC-DDPM [27]	500		27.39	0.7579	39.07
	MC-DDPM+PDM	2000	1000	28.45	0.7660	39.78
	MC-DDPM+PDM	2000	500	27.89	0.7640	39.36
	SR3 [8]	100		28.12	0.7611	39.48
	SR3 [8]	50		27.92	0.7601	39.31
	SR3 [8]	25		27.69	0.7543	38.92
	SR3+PDM	100	50	28.12	0.7610	39.47
	SR3+PDM	100	25	28.02	0.7598	39.25

is defined as the PSNR between the original LR image and the k-space down-sampled SR result:

$$\text{Consist}(I_{sr}, I_{lr}) = \text{PSNR}\Big(\text{KSZP}(I_{sr}), I_{lr}\Big). \tag{20}$$

The definition in Eq. (20) measures the consistency between the SR results and the original inputs. The inconsistency between the input and output can be detrimental to clinical applications.

Quantitative results on the three datasets are summarized in Tab. II to IV. In the tables, T denotes the number of denoising steps of diffusion models, while K is the denoising steps of partial diffusion models. In each group, we use colors to indicate methods with different inference denoising steps. In Tab. VI, we summarize the per-image execution time of different diffusion models on $80 \times 80 \rightarrow 320 \times 320$ image super-resolution.

These quantitative results clearly demonstrate the superiority of PDMs in accelerating and enhancing the performance of diffusion models. For example, as shown in Tab. II, under the

TABLE III: PSNR, SSIM and Consistency of our in-house MRI dataset.

	Method	T	K	PSNR	SSIM	Consist
×2	Bicubic			36.30	0.9163	42.98
	EDSR [41]			37.21	0.9228	43.90
	SwinIR [40]			38.32	0.9231	44.01
	SRGAN [4]			36.11	0.9128	39.98
	SMORE [1]			36.95	0.9407	40.21
	ScoreMRI [11]	2000		39.74	0.9511	46.23
	ScoreMRI [11]	1000		39.57	0.9502	46.19
	ScoreMRI [11]	500		38.72	0.9379	45.75
	ScoreMRI+PDM	2000	1000	39.73	0.9510	46.23
	ScoreMRI+PDM	2000	500	39.58	0.9501	46.20
	MC-DDPM [27]	2000		39.63	0.9485	46.04
	MC-DDPM [27]	1000		39.45	0.9411	45.93
	MC-DDPM [27]	500		38.93	0.9351	45.67
	MC-DDPM+PDM	2000	1000	39.63	0.9483	46.03
	MC-DDPM+PDM	2000	500	39.39	0.9456	45.82
	SR3 [8]	100		39.37	0.9449	46.19
	SR3 [8]	50		39.37	0.9429	46.07
	SR3 [8]	25		38.72	0.9270	45.49
	PDM	100	50	39.37	0.9448	46.19
	PDM	100	25	39.34	0.9479	46.17
×4	Bicubic			31.00	0.7783	40.81
	EDSR [41]			33.48	0.8461	38.45
	SwinIR [40]			33.45	0.8457	39.32
	SRGAN [4]			31.44	0.7911	31.82
	SMORE [1]			32.29	0.6765	32.39
	ScoreMRI [11]	2000		34.21	0.8523	41.26
	ScoreMRI [11]	1000		34.18	0.8521	41.24
	ScoreMRI [11]	500		34.09	0.8506	41.17
	ScoreMRI+PDM	2000	1000	34.21	0.8522	41.26
	ScoreMRI+PDM	2000	500	34.17	0.8521	41.23
	MC-DDPM [27]	2000		34.19	0.8523	41.25
	MC-DDPM [27]	1000		34.13	0.8514	41.20
	MC-DDPM [27]	500		33.86	0.8479	41.06
	MC-DDPM+PDM	2000	1000	34.17	0.8522	41.23
	MC-DDPM+PDM	2000	500	34.11	0.8510	41.18
	SR3 [8]	100		34.05	0.8511	41.18
	SR3 [8]	50		33.89	0.8502	41.11
	SR3 [8]	25		33.23	0.8381	40.15
	PDM	100	50	34.04	0.8510	41.18
	PDM	100	25	33.89	0.8496	41.07

×2 setting, ScoreMRI [11] achieves a PSNR of 34.12 and an SSIM of 0.9113 using 2,000 denoising steps. Under the same setting, the partial diffusion model achieves a PSNR of 34.12 and an SSIM of 0.9112, which is very close to ScoreMRI but uses only 1,000 steps. Under the ×4 setting, MC-DDPM [27] achieves a PSNR of 28.01 and an SSIM of 0.7653 using 1,000 denoising steps. Under the same setting and with the same number of steps, partial diffusion model achieves a PSNR of 28.45 and an SSIM of 0.7660. In general, the results across the three datasets consistently reveal that:

1) Diffusion models significantly outperform other deep learning-based methods in all metrics, particularly in terms of consistency.
2) PDMs reduce the number of denoising steps while achieving the same or very similar results compared to the original diffusion models.
3) When using the same number of denoising steps, PDMs achieve much better image quality.

In Tab. VI, we summarized the running time of different diffusion-based methods for 80×80 → 80×80 super-resolution. We calculated the average time of generating a single image on the ProstateX dataset. As shown in Tab. VI, the running

TABLE IV: In-plane MRI super-resolution results on FastMRI dataset.

	Method	T	K	PSNR	SSIM	Consist
×2	Bicubic			36.48	0.9059	44.21
	EDSR [41]			38.51	0.9121	46.68
	SwinIR [40]			38.23	0.9138	46.24
	SRGAN [4]			36.49	0.8994	42.23
	SMORE [1]			36.62	0.9025	43.21
	ScoreMRI [11]	2000		39.54	0.9375	49.05
	ScoreMRI [11]	1000		39.37	0.9312	49.01
	ScoreMRI [11]	500		38.72	0.9279	48.55
	ScoreMRI+PDM	2000	1000	39.53	0.9375	49.05
	ScoreMRI+PDM	2000	500	39.41	0.9358	48.89
	MC-DDPM [27]	2000		39.53	0.9375	49.04
	MC-DDPM [27]	1000		39.41	0.9341	48.93
	MC-DDPM [27]	500		38.63	0.9251	48.67
	MC-DDPM+PDM	2000	1000	39.48	0.9373	47.03
	MC-DDPM+PDM	2000	500	39.39	0.9340	46.82
	SR3 [8]	100		39.33	0.9363	48.85
	SR3 [8]	50		39.02	0.9257	48.72
	SR3 [8]	25		38.31	0.9125	48.27
	SR3+PDM	100	50	39.33	0.9361	48.84
	SR3+PDM	100	25	39.14	0.9282	48.67
×4	Bicubic			32.17	0.8454	42.86
	EDSR [41]			33.49	0.8388	44.49
	SwinIR [40]			33.22	0.8312	44.19
	SRGAN [4]			32.13	0.7536	36.36
	SMORE [1]			32.31	0.7571	36.51
	ScoreMRI [11]	2000		35.11	0.8540	47.91
	ScoreMRI [11]	1000		34.99	0.8527	47.71
	ScoreMRI [11]	500		34.39	0.8467	47.00
	ScoreMRI+PDM	2000	1000	35.03	0.8537	47.86
	ScoreMRI+PDM	2000	500	34.87	0.8522	47.64
	MC-DDPM [27]	2000		35.07	0.8535	47.85
	MC-DDPM [27]	1000		34.78	0.8529	47.69
	MC-DDPM [27]	500		34.32	0.8434	46.94
	MC-DDPM+PDM	2000	1000	35.01	0.8533	47.78
	MC-DDPM+PDM	2000	500	34.67	0.8501	47.25
	SR3 [8]	100		34.89	0.8529	47.72
	SR3 [8]	50		33.71	0.8501	47.64
	SR3 [8]	25		33.19	0.8337	46.87
	SR3+PDM	100	50	34.87	0.8528	47.70
	SR3+PDM	100	25	33.69	0.8500	47.61

time scales linearly with the number of denoising steps. The running time on other datasets is similar because it only depends on the image dimensions and hardware.

C. Incorporating with Accelerated Diffusion Models

In this experiment, we incorporate PDMs with several accelerated diffusion models to further improve their efficiency. In particular, we experimented with DDIM [29], Consistency Models (CM) [30], and DPM-Solver [28]. These models allow flexible number of denoising steps. We used different denoising steps, e.g. 5, 10, 15 and 20, to assess the performance under various computation budgets. We used the *consistency distillation* when experimenting with consistency models. We reused the configurations and hyperparameters from the original papers, making only the necessary adjustments to incorporate partial diffusion models.

As demonstrated in Tab. V, PDMs can effectively reduce the number of denoising steps required by accelerated diffusion models. On one hand, PDMs reduce the number of steps while maintaining comparable performance. On the other hand, when employing an equivalent number of denoising steps, PDMs achieve significantly improved image quality. This is

TABLE V: The incorporation of PDMs with accelerated diffusion models for ×4 super-resolution on the ProstateX dataset.

Method	T	K	PSNR	SSIM	Consist
DDIM [29]	20		27.62	0.7601	39.39
DDIM [29]	15		27.51	0.7582	39.32
DDIM [29]	10		27.21	0.7466	38.75
DDIM [29]	5		26.13	0.7161	37.58
DDIM + PDM	20	15	27.62	0.7601	39.39
DDIM + PDM	20	10	27.61	0.7598	39.37
DDIM + PDM	20	5	26.85	0.7428	38.18
DPM-Solver [28]	20		27.67	0.7608	39.41
DPM-Solver [28]	15		27.60	0.7598	39.38
DPM-Solver [28]	10		27.23	0.7467	39.29
DPM-Solver [28]	5		26.49	0.7298	37.71
DPM-Solver + PDM	20	15	27.67	0.7607	39.40
DPM-Solver + PDM	20	10	27.66	0.7604	39.39
DPM-Solver + DPM	20	5	27.02	0.7467	38.37
CM [30]	20		27.89	0.7611	39.45
CM [30]	15		27.78	0.7603	39.36
CM [30]	10		27.69	0.7521	38.85
CM+PDM	5		26.67	0.7317	37.83
CM+PDM	20	15	27.89	0.7610	39.44
CM+PDM	20	10	27.83	0.7607	39.44
CM+PDM	20	5	27.11	0.7481	38.63

TABLE VI: Average execution time of various diffusion models for generating a single image. The time was measured on (80×80 → 320×320) image super-resolution with an NVIDIA RTX 8000 GPU.

Method	# Steps	Time (sec)
SR3 [8]	100	4.8
SR3 + PDM	50	2.4
SR3 + PDM	25	1.2
MC-DDPM [27]	2000	92.7
MC-DDPM + PDM	1000	46.4
MC-DDPM + PDM	500	23.3
ScoreMRI [11]	2000	88.1
ScoreMRI + PDM	1000	44.3
ScoreMRI + PDM	500	22.0

because PDMs decrease the number of denoising iterations by lowering the noise levels to be denoised (from $x_T \rightarrow x_0$ to $x_K \rightarrow x_0$, $K \ll T$), while other acceleration methods, such as DDIM, DPM-Solver, and CM, reduce the number of iterations without changing the noise level ($x_T \rightarrow x_0$). Consequently, PDMs exhibit higher iteration density (i.e., the number of iterations divided by the noise level) than other methods, leading to more accurate gradient estimations and higher-quality generations.

D. Through-plane MRI Super-resolution

Multi-slice 2D MRI images have an anisotropic resolution due to the heterogeneous pixel distances. For example, in our prostate MRI dataset, the in-plane pixel spacing (the physical distance between two pixels) is 0.625 mm, while the slice spacing (the physical distance between two slices) is 3.6 mm. In this experiment, we test our method in improving the through-plane resolution of MRI, where the upsampling scale is set to $6 \approx 3.6/0.625$.

To construct low- and high-resolution training pairs, we use two distinct scans acquired in orthogonal planes, e.g., axial scan and coronal scan, during training and testing. The two orthogonal scans are illustrated in Fig. 6. Let anterior↔posterior

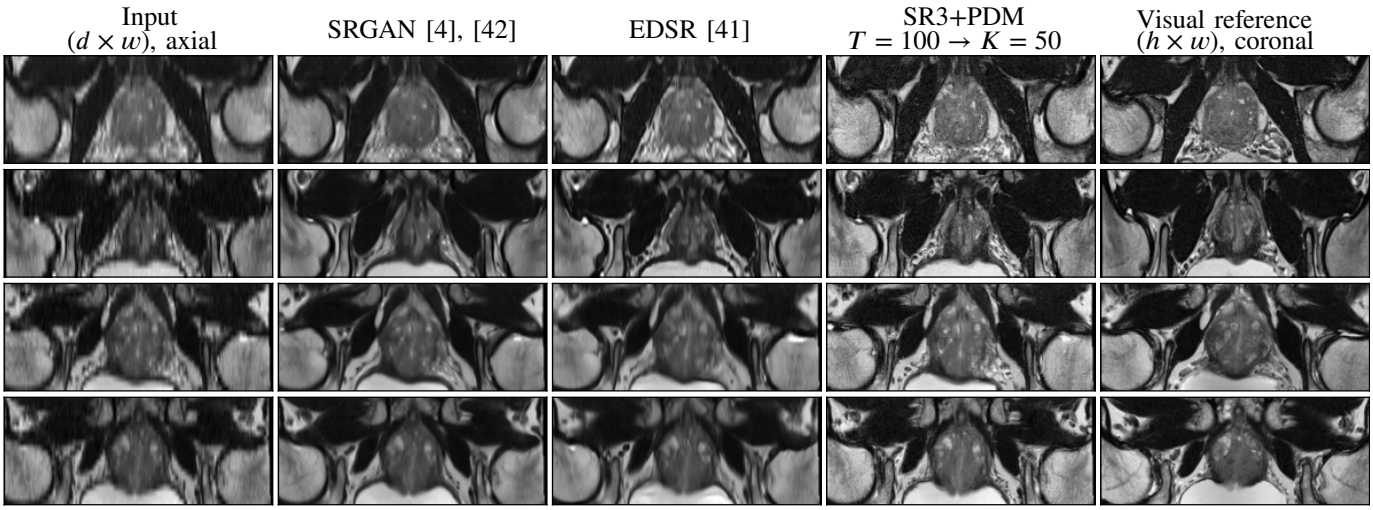


Fig. 5: Example results on through-plane MRI image super-resolution. The model is trained with in-plane slices $(h \times w)$ of coronal scan and the test input is the through-plane $(w \times d)$ images of axial scan. The visual reference is an in-plane slice $(h \times w)$ from coronal scan and is not necessarily aligned with the results. We only visualize PDMs because the results of SR3 and SR3+PDMs are very close.

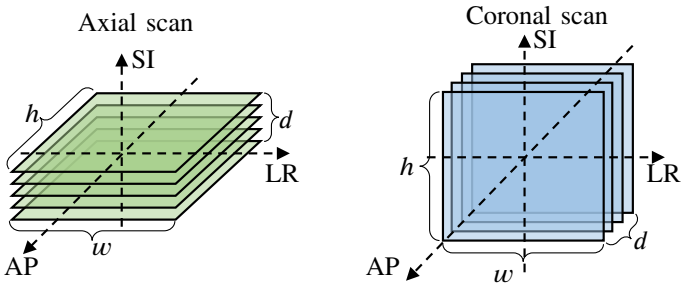


Fig. 6: Axial scan, coronal scan, and the three anatomical directions: anterior↔posterior (AP), left↔right (LR) and superior↔inferior (SI).

(AP) left↔right (LR) and superior↔inferior (SI) be the anatomical directions in 3D space. Axial scan captures 2D slices in the AP and LR planes, and coronal scan captures slices in SI and LR planes. Let $h \times w \times d$ be the shape of MRI data where $h \times w$ is the in-plane image size and d is the number of slices. In our data, $h = w = 320$ and $d = 20$. During training, we collect $h \times w$ in-plane slices from the coronal scan as the high-resolution image, and downsample those slices along the h dimension (SI) to simulate the low through-plane (SI direction) in the axial scan. The models learn to super-resolution along the SI direction by training with the data pairs. During the test, the model takes as input the $w \times d$ through-plane slices of the axial scan and performs super-resolution in the d dimension (SI).

Example through-plane MRI super-resolution results of the axial scan are shown in Fig. 5. There is no such 'ground-truth' in through-plane super-resolution, we use the in-plane slice from a separate acquisition of the coronal scan as the visual reference. The visual references are not necessarily perfectly aligned with the super-resolution results due to potential patient motion between the two scans. Note that we use an average of 6 ($\approx 3.6/0.625$) consecutive through-plane images as the input to super-resolution models to compensate for the differences in pixel spacing (0.625 mm) and slice thickness (3.6 mm). As shown in Fig. 5, SRGAN generates blurry results that lack rich texture details. LIIF and our method generate much better high-frequency details and are generally aligned better with the visual reference.

E. Application to Prostate Zonal Segmentation

Prostate zonal segmentation is an important step in automatic prostate cancer detection, and a suspicious lesion should be analyzed differently in different prostate zones due to variations in image appearance and cancer prevalence [43]. In this experiment, we test the performance of zonal segmentation using images upsampled with different methods. We use a pretrained model [44] which segments the prostate into the peripheral zone (PZ) and the transition zone (TZ).

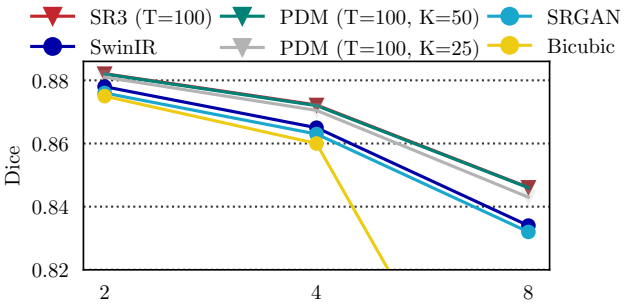


Fig. 7: Zonal segmentation performance under different upsampling factors. The test images are downsampled and then upsampled by various super-resolution models.

Fig. 7 compares the dice coefficients of segmentation results under various upsampling factors. The results demonstrate that our method consistently achieves higher segmentation performance under various upsampling factors, and the performance gap is becoming wider at large upsampling

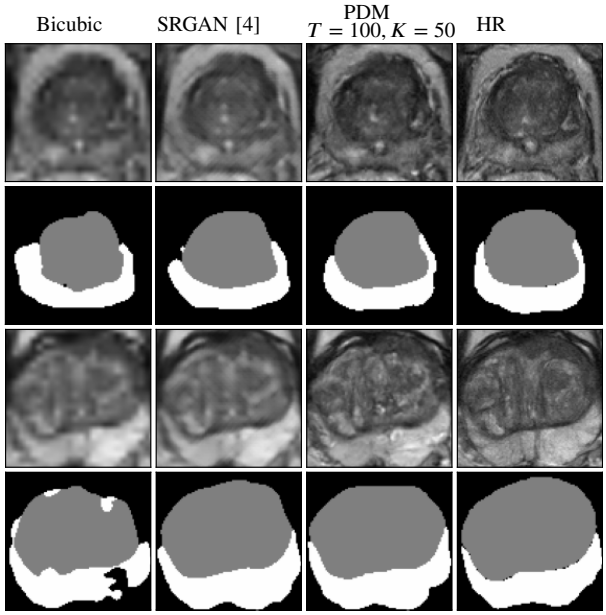


Fig. 8: Example images and zonal segmentation results. The input images are upsampled by ×4 using different methods. $K = 50$ was used for PDMs to reduce the iterations of SR3 with $T = 100$. The results of SR3 are omitted because they are very similar to the results of PDMs and visually indistinguishable.

factors. Interestingly, although our upsampled images are visually much more realistic than SRGAN, the improvements in segmentation are not as noticeable as the visual differences. This reveals that visual realism is not well correlated with segmentation quality. Fig. 8 shows some zonal segmentation results using images upsampled (×4) by different methods.

F. Ablation Study

We did ablation studies to verify the effectiveness of Latent Alignment and to demonstrate how we choose K during training. The experiments were conducted on the ProstateX dataset.

1) Latent Alignment: We compared the super-resolution quality with and without *latent alignment* (Sec. IV-C). Scale is set to ×4 in this experiment. Quantitative results are summarized in Tab. VII and visual comparisons are made in Fig. 9. The results in Tab. VII show that: 1) 'Latent alignment' significantly improves the performance of partial diffusion models, especially when many steps are skipped (i.e., smaller K values). This is because smaller K increases the disparity between X_K^{HR} and X_K^{LR}, and latent alignment mitigates these approximation errors to improve generation quality. 2) The improvements become larger with lower K values, because of the larger gap between low- and high-resolution latents (see Fig. 2). 3) Latent alignment with a cosine schedule performs slightly better than with a linear schedule due to a smoother transition from low-resolution to high-resolution latents.

2) Different K values: We tested the performance with different inference steps K for ×4 super-resolution on the ProstateX dataset. The results in Fig. 10 demonstrate that

TABLE VII: SSIM of ×4 super-resolution with various 'latent alignment' schedules on the ProstateX dataset. We utilize partial diffusion to reduce the denoising steps of SR3 [8] from $T = 100$ to $K = 75, 50$, and $K = 25$.

K	75	50	25
w/o	0.7610	0.7579	0.6704
Linear	0.7611	0.7607	0.7594
Cosine	0.7611	0.7610	0.7598

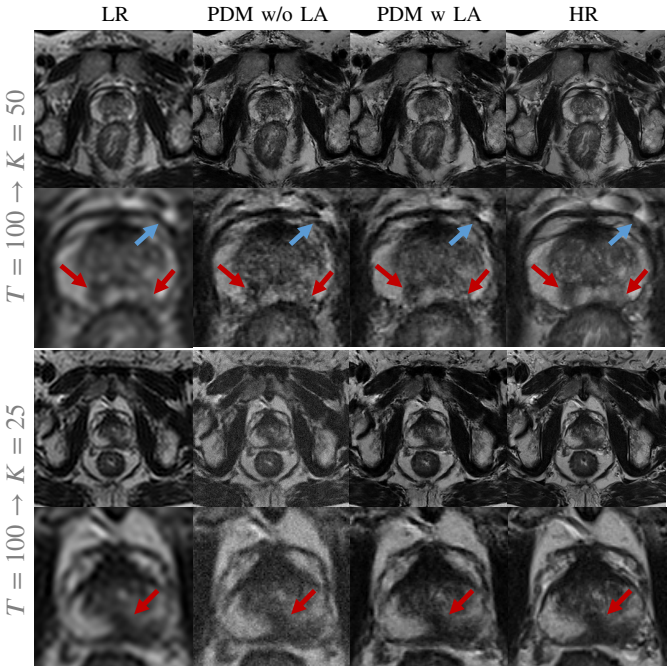


Fig. 9: Super-resolution results (×4) on ProstateX dataset with and without latent alignment (LA). We utilized PDMs to reduce the denoising steps of SR3 [8] from $T = 100$ to $K = 50$ and $K = 25$. The red arrows point out prostate tumors and blue arrows highlight the differences between with and without LA. There is a clear benefit of using LA, especially a larger number of steps are skipped.

partial diffusion models can achieve very similar performance with SR3 with half of denoising steps ($K = 50$), and achieve decent performance with only a quarter of denoising steps ($K = 25$).

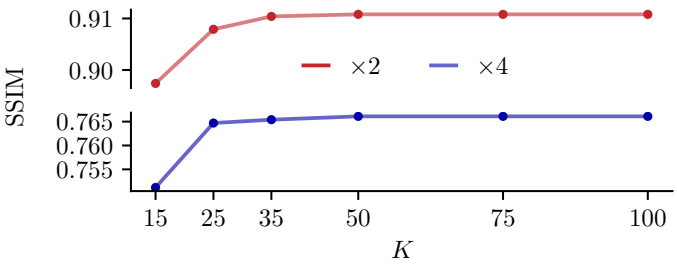


Fig. 10: SSIM of ×4 super-resolution on the ProstateX dataset using different K values.

VI. Conclusions

We introduced the Partial Diffusion Model for MRI super-resolution. Our method accelerates diffusion-based super-resolution methods by skipping part of the diffusion steps. We first observed that the latent of a pair of low- and high-resolution images gradually converge and become indistinguishable after a certain noise level. Based on this observation, we proposed to approximate the high-resolution latents with the corresponding low-resolution latents, allowing us to skip and shortcut some of the denoising steps. To mitigate the approximation error, we further proposed the latent alignment that gradually interpolates between the high- and low-resolution latents. Partial diffusion models with latent alignment essentially establish a new diffusing (denoising) trajectory where the latent directly evolves from the high-resolution (low-resolution) image to the low-resolution (high-resolution) image. Extensive experiments on clinical MRI datasets demonstrated that the proposed method significantly reduces the number of denoising steps without sacrificing the quality of the generation. One limitation is that it applies only to conditional generation tasks where a conditional input can be used for the approximation.

One limitation of our method is the requirement for input that closely resembles the target image in structure to achieve accurate approximation. This restricts its applicability to conditional image generation tasks such as image super-resolution and translation. Furthermore, the evaluation section relies heavily on numerical metrics such as PSNR, SSIM, and Consistency to validate that the proposed method can replicate the output of diffusion models with fewer denoising steps. Human expert evaluation is yet to be introduced to perceptually assess the quality of the generated images.

References

[1] C. Zhao, B. E. Dewey, D. L. Pham, P. A. Calabresi, D. S. Reich, and J. L. Prince, "Smore: a self-supervised anti-aliasing and super-resolution algorithm for mri using deep learning," *IEEE transactions on medical imaging*, vol. 40, no. 3, pp. 805–817, 2020.

[2] M. de Leeuw den Bouter, G. Ippolito, T. O'Reilly, R. Remis, M. van Gijzen, and A. Webb, "Deep learning-based single image super-resolution for low-field mr brain images," *Scientific Reports*, vol. 12, no. 1, p. 6362, 2022.

[3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014.

[4] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4681–4690.

[5] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," *Neural Information Processing Systems*, vol. 32, 2019.

[6] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[7] Y. Song and S. Ermon, "Improved techniques for training score-based generative models," *Neural Information Processing Systems*, vol. 33, pp. 12 438–12 448, 2020.

[8] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Recognition and Machine Intelligence*, pp. 1–14, 2022.

[9] H. Li, Y. Yang, M. Chang, S. Chen, H. Feng, Z. Xu, Q. Li, and Y. Chen, "Srdiff: Single image super-resolution with diffusion probabilistic models," *Neurocomputing*, vol. 479, pp. 47–59, 2022.

[10] A. L. Y. Hung, K. Zhao, H. Zheng, R. Yan, S. S. Raman, D. Terzopoulos, and K. Sung, "Med-cdiff: Conditional medical image generation with diffusion models," *Bioengineering*, vol. 10, no. 11, p. 1258, 2023.

[11] H. Chung and J. C. Ye, "Score-based diffusion models for accelerated mri," *Medical image analysis*, vol. 80, p. 102479, 2022.

[12] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations*, 2021.

[13] R. Fattal, "Image upsampling via imposed edge statistics," in *ACM SIGGRAPH 2007 papers*, 2007, pp. 95–es.

[14] J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[15] Q. Shan, Z. Li, J. Jia, and C.-K. Tang, "Fast image/video upsampling," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 5, pp. 1–7, 2008.

[16] K. I. Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *IEEE Transactions on Pattern Recognition and Machine Intelligence*, vol. 32, no. 6, pp. 1127–1133, 2010.

[17] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Recognition and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2015.

[18] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *IEEE/CVF International Conference on Computer Vision*, 2015, pp. 370–378.

[19] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1874–1883.

[20] M. S. Sajjadi, B. Scholkopf, and M. Hirsch, "Enhancenet: Single image super-resolution through automated texture synthesis," in *IEEE/CVF International Conference on Computer Vision*, 2017, pp. 4491–4500.

[21] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, and W. Wu, "Feedback network for image super-resolution," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3867–3876.

[22] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *ECCV*. Springer, 2016, pp. 694–711.

[23] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *International Conference on Learning Representations*, 2019.

[24] Y. Chen, Y. Tai, X. Liu, C. Shen, and J. Yang, "Fsrnet: End-to-end learning face super-resolution with facial priors," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2492–2501.

[25] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," 2017.

[26] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *ICML*. PMLR, 2015, pp. 2256–2265.

[27] Y. Xie and Q. Li, "Measurement-conditioned denoising diffusion probabilistic model for under-sampled medical image reconstruction," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2022, pp. 655–664.

[28] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, "Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps," in *Neural Information Processing Systems*, 2022.

[29] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=St1giarCHLP

[30] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever, "Consistency models," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23. JMLR.org, 2023.

[31] X. Luo, Y. Xie, Y. Qu, and Y. Fu, "Skipdiff: Adaptive skip diffusion model for high-fidelity perceptual image super-resolution," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 5, 2024, pp. 4017–4025.

[32] H. Chung, B. Sim, and J. C. Ye, "Come-closer-diffuse-faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 413–12 422.

[33] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan, "Wavegrad: Estimating gradients for waveform generation," in *International Conference on Learning Representations*, 2021.

[34] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," in *International Conference on Learning Representations*, 2022.
[35] H. Zheng, P. He, W. Chen, and M. Zhou, "Truncated diffusion probabilistic models and diffusion-based adversarial auto-encoders," in *International Conference on Learning Representations*, 2023.
[36] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *International Conference on Learning Representations*, 2014.
[37] G. Litjens, O. Debats, J. Barentsz, N. Karssemeijer, and H. Huisman, "Prostatex challenge data," *Cancer Imaging Arch*, vol. 10, p. K9TCIA, 2017.
[38] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Neural Information Processing Systems*, vol. 32, 2019.
[39] F. Knoll, J. Zbontar, A. Sriram, M. J. Muckley, M. Bruno, A. Defazio, M. Parente, K. J. Geras, J. Katsnelson, H. Chandarana *et al.*, "fastmri: A publicly available raw k-space and dicom dataset of knee images for accelerated mr image reconstruction using machine learning," *Radiology: Artificial Intelligence*, vol. 2, no. 1, p. e190007, 2020.
[40] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "Swinir: Image restoration using swin transformer," in *IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1833–1844.
[41] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *CVPRW*, 2017, pp. 136–144.
[42] R. Sood and M. Rusu, "Anisotropic super resolution in prostate mri using super resolution generative adversarial networks," in *ISBI*. IEEE, 2019, pp. 1688–1691.
[43] B. Israel, M. van der Leest, M. Sedelaar, A. R. Padhani, P. Zamecnik, and J. O. Barentsz, "Multiparametric magnetic resonance imaging for the detection of clinically significant prostate cancer: what urologists need to know. part 2: interpretation," *European urology*, vol. 77, no. 4, pp. 469–480, 2020.
[44] A. L. Y. Hung, H. Zheng, Q. Miao, S. S. Raman, D. Terzopoulos, and K. Sung, "Cat-net: A cross-slice attention transformer model for prostate zonal segmentation in mri," *IEEE TMI*, vol. 42, no. 1, pp. 291–303, 2022.