Improving Diffusion Models for Inverse Problems using Manifold Constraints

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recently, diffusion models have been used to solve various inverse problems in an unsupervised manner with appropriate modifications to the sampling process. However, the current solvers, which recursively apply a reverse diffusion step followed by a measurement consistency step, often produce sub-optimal results. By studying the generative sampling path, here we show that current solvers throw the sample path off the data manifold, and hence the error accumulates. To address this, we propose an additional correction term inspired by the manifold constraint, which can be used synergistically with the previous solvers to make the iterations close to the manifold. The proposed manifold constraint is straightforward to implement within a few lines of code, yet boosts the performance by a surprisingly large margin. With extensive experiments, we show that our method is superior to the previous methods both theoretically and empirically, producing promising results in many applications such as image inpainting, colorization, and sparse-view computed tomography.

1 Introduction

Diffusion models have shown impressive performance both as generative models themselves [35, 10], and also as unsupervised inverse problem solvers [35, 6, 7, 21] that do not require problem-specific training. Specifically, given a pre-trained unconditional score function (i.e. denoiser), solving the reverse stochastic differential equation (SDE) numerically would amount to sampling from the data generating distribution [35]. For many different inverse problems (e.g. super-resolution [6, 7], inpainting [35, 7], compressed-sensing MRI (CS-MRI) [34, 7], sparse view CT (SV-CT) [34], etc.), it was shown that simple incorporation of the measurement process produces satisfactory conditional samples, even when the model was not trained for the specific problem.

Nevertheless, for certain problems (e.g. inpainting), currently used algorithms often produce unsatisfactory results when implemented naively (e.g. boundary artifacts, as shown in fig. 1 (b)). The authors in [26] showed that in order to produce high quality reconstructions, one needs to iterate back and forth between the noising and the denoising step at least > 10 times *per iteration*. These iterations are computationally demanding and should be avoided, considering that diffusion models are slow to sample from even without such iterations.

Recently, another type of score-based approach, called Noise2Score [23], was proposed for image denoising without clean references. In contrast to the diffusion models, Noise2Score is computationally efficient as it is a deterministic single-step approach that does not involve any stochastic sampling. Unfortunately, the performance of Noise2Score is inferior to the diffusion models that rely on iterative denoising. Furthermore, Noise2Score has never been applied to other inverse problem applications beyond the image denoising.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

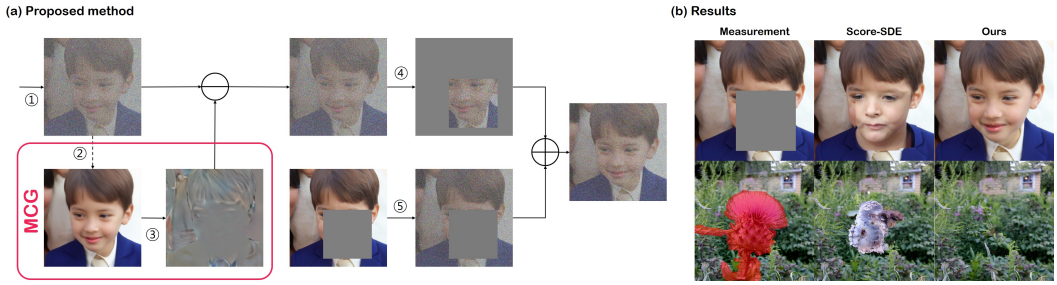

Figure 1: Visual schematic of the MCG correction step. (a) ① Unconditional reverse diffusion generates $\boldsymbol{x}_i$; ② Q_i maps the noisy $\boldsymbol{x}_i$ to generate $\hat{\boldsymbol{x}}_0$; ③ Manifold Constrained Gradient (MCG) $\frac{\partial}{\partial \boldsymbol{x}_i}\|\boldsymbol{W}(\boldsymbol{y}-\boldsymbol{H}\hat{\boldsymbol{x}}_0)\|_2^2$ is applied to fix the iteration on manifold; ④ Takes the orthogonal complement; ⑤ Samples from $p(\boldsymbol{y}_i|\boldsymbol{y})$, then combines $\boldsymbol{A}\boldsymbol{x}'_{i-1}$ and $\boldsymbol{y}_i$. (b) Representative results of inpainting, compared with score-SDE [35]. Reconstructions with score-SDE produce incoherent results, while our method produces high fidelity solutions.

Given the two types of seemingly different approaches that rely on the score function, one may wonder what is the relation between the two, and whether there is a way to synergistically combine the two. Accordingly, one of the main contributions of this work is to first demonstrate that the key idea of Noise2Score leads to what we call the *manifold constraint* that can be combined in a complementary fashion with the existing diffusion models to significantly improve the performance of reconstruction across various problems despite the simplicity in the implementation. Moreover, we theoretically prove that the correction term from the manifold constraint enforces the sample path to stay on the plane tangent to the data manifold[1], so by combining with the reverse diffusion step, the solution becomes more stable and accurate.

2 Related Works

2.1 Diffusion Models

Continuous Form For a continuous diffusion process $\boldsymbol{x}(t) \in \mathbb{R}^n$, $t \in [0,1]$, we set $\boldsymbol{x}(0) \sim p_0(\boldsymbol{x}) = p_{data}$, where p_{data} represents the data distribution of interest, and $\boldsymbol{x}(1) \sim p_1(\boldsymbol{x})$, with $p_1(\boldsymbol{x})$ approximating spherical Gaussian distribution, containing no information of data. Here, the forward noising process is defined with the following Itô stochastic differential equation (SDE) [35]:

$$d\boldsymbol{x} = \bar{\boldsymbol{f}}(\boldsymbol{x},t)dt + \bar{g}(t)d\boldsymbol{w}, \tag{1}$$

with $\bar{\boldsymbol{f}} : \mathbb{R}^d \mapsto \mathbb{R}^d$ defining the linear drift function, $\bar{g}(t) : \mathbb{R} \mapsto \mathbb{R}$ defining a scalar diffusion coefficient, and $\boldsymbol{w} \in \mathbb{R}^n$ denoting the standard $n-$dimensional Wiener process. Choosing $\bar{\boldsymbol{f}} = 0$, $\bar{g}(t) = \sqrt{d[\sigma^2(t)/dt]}$ leads to the Brownian motion, called the variance exploding SDE (VE-SDE), and choosing $\bar{\boldsymbol{f}}(\boldsymbol{x},t) = -\beta(t)\boldsymbol{x}/2$, $\bar{g}(t) = \sqrt{\beta(t)}$ leads to the Ornstein-Uhlenbeck process [31] where the mean decays to 0 as $t \to 1$, coined variance preserving SDE (VP-SDE). The forward SDE in (1) is coupled with the following reverse SDE by Anderson's theorem [1, 35]

$$d\boldsymbol{x} = [\bar{\boldsymbol{f}}(\boldsymbol{x},t) - \bar{g}(t)^2\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})]dt + \bar{g}(t)d\bar{\boldsymbol{w}}, \tag{2}$$

with dt denoting the infinitesimal negative time step, and $\bar{\boldsymbol{w}}$ defining the standard Wiener process running backward in time. Note that the reverse SDE defines the generative process through the score function $\nabla_{\boldsymbol{x}} p_t(\boldsymbol{x})$, which is typically trained by minimizing the following score-matching objective

$$\min_{\theta} \mathbb{E}_{t\sim U(\varepsilon,1),\boldsymbol{x}(t)\sim p(\boldsymbol{x}(t))}\left[\|\boldsymbol{s}_\theta(\boldsymbol{x}_t,t) - \nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}(t))\|_2^2\right]. \tag{3}$$

Once the parameter θ^* for the score function is estimated, one can replace $\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}(t))$ with $s_{\theta^*}(\boldsymbol{x}_t,t)$ to solve the reverse SDE.

[1] We coin our method **M**anifold **C**onstrained **G**radient (MCG).

Discrete Form Due to the linearity of $\bar{\boldsymbol{f}}$ and $\bar{g}$, the forward diffusion step can be implemented with a simple reparameterization trick [24]. Namely, the general form of the forward diffusion is

$$\boldsymbol{x}_i = a_i \boldsymbol{x}_0 + b_i \boldsymbol{z}, \quad \boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I}), \tag{4}$$

where we have replaced the continuous index $t \in [0, 1]$ with the discrete index $i \in \mathbb{N}$. On the other hand, the discrete reverse diffusion step is implemented as

$$\boldsymbol{x}_{i-1} = \boldsymbol{f}(\boldsymbol{x}_i, \boldsymbol{s}_\theta) + g(\boldsymbol{x}_i)\boldsymbol{z}, \quad \boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I}), \tag{5}$$

where we have replaced the ground truth score function with the trained one. We detail the choice of $a_i, b_i, \boldsymbol{f}, g$ in appendix B.

2.2 Conditional Diffusion for Inverse problems

The main problem of our interest in this paper is the inverse problem, retrieving the unknown $\boldsymbol{x} \in \mathbb{R}^n$ from a measurement $\boldsymbol{y}$:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{\epsilon}, \quad \boldsymbol{y} \in \mathbb{R}^m, \boldsymbol{H} \in \mathbb{R}^{m \times n}. \tag{6}$$

where $\boldsymbol{\epsilon} \in \mathbb{R}^n$ is the noise in the measurement. Accordingly, for the case of the inverse problems, our goal is to generate samples from a conditional distribution with respect to the measurement $\boldsymbol{y}$, i.e. $p(\boldsymbol{x}|\boldsymbol{y})$. Accordingly, the score function $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})$ in (2) should be replaced by the conditional score $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}|\boldsymbol{y})$. Unfortunately, this strictly restricts the generalization capability of the neural network since the conditional score should be retrained whenever the conditions change. To address this, recent conditional diffusion models [18, 35, 6, 7] utilizes the unconditional score function $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})$ but relies on the measurement constraint to impose the conditions. Specifically, one can apply the following:

$$\boldsymbol{x}'_{i-1} = \boldsymbol{f}(\boldsymbol{x}_i, s_\theta) + g(\boldsymbol{x}_i)\boldsymbol{z}, \quad \boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I}), \tag{7}$$

$$\boldsymbol{x}_{i-1} = \boldsymbol{A}\boldsymbol{x}'_{i-1} + \boldsymbol{b}_i, \tag{8}$$

where $\boldsymbol{A}, \boldsymbol{b}_i$ are function of $\boldsymbol{H}, \boldsymbol{y}_0$, and $\boldsymbol{x}_0$. Note that (7) is identical to the unconditional reverse diffusion step in (5), whereas (8) effectively imposes the condition. It was shown in [7] that any general contraction mapping (e.g. projection onto convex sets, gradient step) may be utilized as (8) to impose the constraint.

2.3 Noise2Score

In the score matching problem (3), $\nabla_{\boldsymbol{x}_i} \log p(\boldsymbol{x}_i)$ is often replaced by $\nabla_{\boldsymbol{x}_i} \log p(\boldsymbol{x}_i|\boldsymbol{x}_0)$, in which the procedure is often called the denoising score matching (DSM) [35, 15]:

$$\min_\theta \mathbb{E}_{i,\boldsymbol{x}_0,\boldsymbol{x}_i} \left[\|\boldsymbol{s}_\theta(\boldsymbol{x}_i, i) + (\boldsymbol{x}_i - a_i\boldsymbol{x}_0)/b_i^2\|_2^2 \right] \tag{9}$$

where we use (4) to provide a closed-form Gaussian distribution. For the estimated parameter θ^* and the fixed sample $\boldsymbol{x}_i$, it is straightforward to see that (9) leads to the one-step estimate of the clean image $\boldsymbol{x}_0$:

$$\hat{\boldsymbol{x}}_0 = (\boldsymbol{x}_i + b_i^2 \boldsymbol{s}_\theta(\boldsymbol{x}_i, i))/a_i, \tag{10}$$

which is known as the denoising autoencoder [15, 33]. In fact, the estimate in (10) is shown Bayes optimal when the noise measurement $\boldsymbol{x}_i$ is given by (4) [36]. Furthermore, this formula can be generalized to other noise models such as Poisson, Gamma, etc. using Tweedie's formula [11, 30]. The resulting denoising formula can all be represented in terms of the score function so that the algorithm is called the Noise2Score [23].

3 Conditional Diffusion using Manifold Constraints

3.1 Manifold Constrained Score

Although our original motivation of using the measurement constraint step in (8) was to utilize the unconditionally trained score function in the reverse diffusion step in (7), there is room for imposing additional constraints while still using the unconditionally trained score function.

Specifically, for a general conditioning $\mathcal{C}$, the Bayes rule $p(\boldsymbol{x}|\mathcal{C}) = p(\mathcal{C}|\boldsymbol{x})p(\boldsymbol{x})/p(\mathcal{C})$ gives us

$$\log p(\boldsymbol{x}|\mathcal{C}) = \log p(\mathcal{C}|\boldsymbol{x}) + \log p(\boldsymbol{x}) - \log p(\mathcal{C}) \tag{11}$$

which leads to

$$\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}|\mathcal{C}) = \nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}) + \nabla_{\boldsymbol{x}} \log p(\mathcal{C}|\boldsymbol{x}), \tag{12}$$

when $\mathcal{C}$ is not a function of $\boldsymbol{x}$. Hence, the score function in the reverse SDE in (2) can be replaced by (12). Furthermore, the new conditioning $\mathcal{C}$ is different from the measurement constrain in (8), so we can still use the measurement consistency step in (8) in addition to the diffusion step in (7) with the modified score in (12).

One of the important contributions of this paper is to reveal that the Bayes optimal denoising step in (10) from Noise2Score leads to a preferred condition both empirically and theoretically. Specifically, we define the set and the constraint

$$\mathcal{C} = \{(\boldsymbol{x}, \boldsymbol{y}) | \boldsymbol{x} \in \mathcal{X}_0\}, \quad \text{where} \quad \mathcal{X}_0 = \{\boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{x} = (\boldsymbol{x} + b_t^2(t)\boldsymbol{s}_\theta(\boldsymbol{x}, t))/a_t(t)\} \tag{13}$$

which we call the *manifold constraint (MCG)*. Under the manifold constraint, if the noise $\boldsymbol{\epsilon}$ in (6) is Gaussian, we have

$$\log p(\mathcal{C}|\boldsymbol{x}) = -\alpha \|\boldsymbol{W}(\boldsymbol{y} - \boldsymbol{H}\hat{\boldsymbol{x}}_0)\|_2^2, \quad \hat{\boldsymbol{x}}_0 := (\boldsymbol{x} + b_t^2(t)\boldsymbol{s}_\theta(\boldsymbol{x}, t))/a_t(t) \tag{14}$$

and α and $\boldsymbol{W}$ depend on the noise covariance. Accordingly, the discrete reverse diffusion under the additional manifold constraint can be represented by

$$\boldsymbol{x}'_{i-1} = \boldsymbol{f}(\boldsymbol{x}_i, \boldsymbol{s}_\theta) - \alpha \frac{\partial}{\partial \boldsymbol{x}_i} \|\boldsymbol{W}(\boldsymbol{y}_0 - \boldsymbol{H}\hat{\boldsymbol{x}}_0)\|_2^2 + g(\boldsymbol{x}_i)\boldsymbol{z}, \quad \boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I}), \tag{15}$$

$$\boldsymbol{x}_{i-1} = \boldsymbol{A}\boldsymbol{x}'_{i-1} + \boldsymbol{b}. \tag{16}$$

In the following, we articulate how the proposed MCG can be incorporated into each application, and study the theoretical properties of the method. Further algorithmic details are presented in appendix C.

We note that the authors of [16] proposed a similar gradient method for the application of temporal imputation and super-resolution. When combining (15) with (16), one can arrive at a similar gradient method proposed in [16], and hence our method can be seen as a generalization to arbitrary linear inverse problems.

3.2 Specific Applications

Inpainting The forward model for inpainting is given as

$$\boldsymbol{y} = \boldsymbol{P}\boldsymbol{x} + \boldsymbol{\epsilon}, \quad \boldsymbol{P} \in \mathbb{R}^{m \times n}, \tag{17}$$

where $\boldsymbol{P} \in \{0, 1\}$ is the matrix consisting of the columns with standard coordinate vectors indicating the indices of measurement. For the steps in (15), (16), we choose the following

$$\boldsymbol{W} = \boldsymbol{I}, \quad \boldsymbol{A} = \boldsymbol{I} - \boldsymbol{P}^T\boldsymbol{P}, \quad \boldsymbol{b}_i = \boldsymbol{P}^T\boldsymbol{y}_i, \quad \boldsymbol{y}_i \sim q(\boldsymbol{y}_i|\boldsymbol{y}) := \mathcal{N}(\boldsymbol{y}_i | a_i\boldsymbol{y}, b_i^2\boldsymbol{I}). \tag{18}$$

Specifically, $\boldsymbol{A}$ takes the orthogonal complement of $\boldsymbol{x}'_{i-1}$, meaning that the measurement subspace is corrected by $\boldsymbol{y}_i$, while the orthogonal components are updated from $\boldsymbol{x}'_{i-1}$. Note that we use $\boldsymbol{y}_i$ sampled from $\boldsymbol{y}$ to match the noise level of the current estimate.

Colorization The forward model for colorization is specified as

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{\epsilon} := \boldsymbol{P}\boldsymbol{M}\boldsymbol{x} + \boldsymbol{\epsilon}, \quad \boldsymbol{P} \in \mathbb{R}^{m \times n}, \quad \boldsymbol{M} \in \mathbb{R}^{n \times n}, \tag{19}$$

where $\boldsymbol{P}$ is the matrix that was used in inpainting, and $\boldsymbol{C}$ is an orthogonal matrix that couples the RGB colormaps. $\boldsymbol{C}^T$ is a matrix that de-couples the channels back to the original space. In other words, one can view colorization as performing imputation in some spectral space. Subsequently, for our colorization method we choose

$$\boldsymbol{W} = \boldsymbol{C}^T, \quad \boldsymbol{A} = \boldsymbol{I} - \boldsymbol{C}^T\boldsymbol{C}, \quad \boldsymbol{b}_i = \boldsymbol{C}^T\boldsymbol{y}_i, \quad \boldsymbol{y}_i \sim q(\boldsymbol{y}_i|\boldsymbol{y}). \tag{20}$$

Again, our forward measurement matrix is orthogonal, and we choose $\boldsymbol{A}$ such that we only affect the orthogonal complement of the measurement subspace.

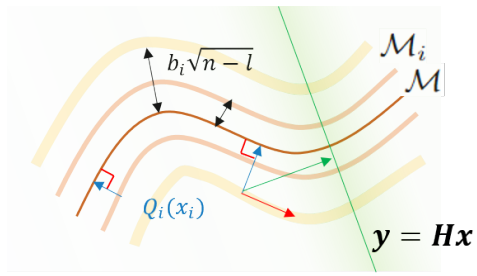

Figure 2: Geometric view of the MCG correction. Each curve represents a manifold of (noisy) data, when $a_i = 1$. By Proposition 1, noisy data are concentrated on the manifold that is away from the data manifold by $b_i\sqrt{n-1}$. In this viewpoint, the backward (or forward) step can be considered as a transition from $\mathcal{M}_i$ to $\mathcal{M}_{i-1}$ (or $\mathcal{M}_{i-1}$ to $\mathcal{M}_i$, respectively). Arrows suggest directions of conventional projection onto convex sets (POCS) step (green arrow) and MCG step (red arrow) which can be predicted by Theorem 1. Since conventional POCS step may leave the manifold, we cannot guarantee the accuracy of the sampling while MCG step maintains the samples on the manifold.

CT Reconstruction For the case of CT reconstruction, the forward model reads

$$\boldsymbol{y} = \boldsymbol{R}\boldsymbol{x} + \boldsymbol{\epsilon}, \quad \boldsymbol{R} \in \mathbb{R}^{m\times n}, \tag{21}$$

where $\boldsymbol{R}$ is the discretized Radon transfrom [5] that measures the projection images from different angles. Note that for CT applications, $\boldsymbol{R}^T$ corresponds to performing backprojection (BP), and $\boldsymbol{R}^\dagger$ corresponds to performing filtered backprojection (FBP). We choose

$$\boldsymbol{W} = \boldsymbol{R}^\dagger, \quad \boldsymbol{A} = \boldsymbol{I} - \boldsymbol{R}^T(\boldsymbol{R}\boldsymbol{R}^T)^\dagger\boldsymbol{R}, \quad \boldsymbol{b}_i = \boldsymbol{R}^T(\boldsymbol{R}\boldsymbol{R}^T)^\dagger\boldsymbol{y}_i, \quad \boldsymbol{y}_i \sim q(\boldsymbol{y}_i|\boldsymbol{y}), \tag{22}$$

where the choice of $\boldsymbol{A}$ reflects that the Radon transform is not orthogonal, and we need the term $(\boldsymbol{R}\boldsymbol{R}^T)^\dagger$ as a term analogous to the filtering step. Indeed, this form of update is known as the algebraic reconstruction technique (ART), a classic technique in the context of CT [13]. We note that this choice is different from what was proposed in [34], where the authors repeatedly apply projection/FBP by explicitly replacing the sinogram in the measured locations. From our experiments, we find that repeated application of FBP is highly numerically unstable, often leading to overflow. This is especially the case when we have limited resources for training data (we use 4k, whereas [34] use 50k), as we further show in section 5.

4 Theoretical Findings

In this section, we theoretically support the effectiveness of the proposed algorithm by showing the problematic behavior of the earlier algorithm and how the proposed algorithm resolves the problem. To see this, we borrow a geometrical viewpoint of the data manifold.

Notation For a scalar a, points $\boldsymbol{x}, \boldsymbol{y}$ and a set A, we use following notations. $aA := \{a\boldsymbol{x} : \boldsymbol{x} \in A\}$; $d(\boldsymbol{x}, A) := \inf_{\boldsymbol{y}\in A} ||\boldsymbol{x} - \boldsymbol{y}||_2$; $B_r(A) := \{\boldsymbol{x} : d(\boldsymbol{x}, A) < r\}$; $T_{\boldsymbol{x}}\mathcal{M}$: the tangent space of to a manifold $\mathcal{M}$ at $\boldsymbol{x}$; $\boldsymbol{J}_f$: the jacobian matrix of a vector valued function f.

To develop the theory, we need an assumption on the data distribution, called the manifold assumption, which is widely exploited in machine learning literature.

Assumption 1 (Manifold assumption)**.** *Suppose $\mathcal{M} \subset \mathbb{R}^n$ is the set of all data points, here we call the data manifold. Then, the manifold coincides with the tangent space with dimension $l \ll n$.*

$$\mathcal{M} \cap B_R(\boldsymbol{x}_0) = T_{x_0}\mathcal{M} \cap B_R(x_0) \text{ and } T_{x_0}\mathcal{M} \cong \mathbb{R}^l.$$

Moreover, the data distribution p_0 is the uniform distribution on the data manifold $\mathcal{M}$.

Under the assumption, the following proposition shows how the data perturbed by noise lies in the ambient space. Fig. 2 illustrates the proposition pictorially.

Proposition 1 (Concentration of noisy data)**.** *Consider the distribution of noisy data $p_i(\boldsymbol{x}_i) = \int p(\boldsymbol{x}_i|\boldsymbol{x})p_0(\boldsymbol{x})d\boldsymbol{x}, p(\boldsymbol{x}_i|\boldsymbol{x}) \sim \mathcal{N}(a_i\boldsymbol{x}, b_i^2\boldsymbol{I})$. Then $p_i(\boldsymbol{x}_i)$ is concentrated on $(n-1)$-dim manifold $\mathcal{M}_i := \{\boldsymbol{y} \in \mathbb{R}^n : d(\boldsymbol{y}, a_i\mathcal{M}) = r_i := b_i\sqrt{n-l}\}$. Rigorously, $p(B_{\epsilon r_i}(\mathcal{M}_i)) > 1 - \delta$.*

Remark 1. *We can infer from the proposition that the score functions are trained only with the data points concentrated on the noisy data manifolds. Therefore, inaccurate inference might be caused by application of a score function on points away from the noisy data manifold.*

Proposition 2 (score function)**.** *Suppose s_θ is the minimizer of* (9)*. Let Q_i be the function that maps $\boldsymbol{x}_i$ to $\hat{\boldsymbol{x}}_0$ for each i,*

$$Q_i : \mathbb{R}^d \to \mathbb{R}^d, \boldsymbol{x}_i \mapsto \frac{1}{a_i}(\boldsymbol{x}_i + b_i^2 s_\theta(\boldsymbol{x}_i, i)).$$

Then, $Q_i(\boldsymbol{x}_i) \in \mathcal{M}$ and $\boldsymbol{J}_{Q_i}^2 = \boldsymbol{J}_{Q_i} = \boldsymbol{J}_{Q_i}^T : \mathbb{R}^d \to T_{Q_i(\boldsymbol{x}_i)}\mathcal{M}$. Intuitively, Q_i is locally an orthogonal projection onto $\mathcal{M}$.

According to the proposition, the score function only concerns the normal direction of the data manifold. In other words, the score function cannot discriminate two data points whose difference is tangent to the manifold. In solving inverse problems, however, we desire to discriminate data points to reconstruct the original signal, and the discrimination is achievable by measurement fidelity. In order to achieve the original signal, the measurement plays a role in correcting the tangent component near the data manifold. Furthermore, with regard to remark 1, diffusion model-based inverse problem solvers should follow the tangent component. The following theorem shows how existing algorithms and the proposed method are different in this regard.

Theorem 1 (Manifold constrained gradient)**.** *A correction by the manifold constrained gradient does not leave the data manifold. Formally,*

$$\frac{\partial}{\partial \boldsymbol{x}_i}\|\boldsymbol{W}(\boldsymbol{y} - \boldsymbol{H}\hat{\boldsymbol{x}}_0)\|_2^2 = -2\boldsymbol{J}_{Q_i}^T\boldsymbol{H}^T\boldsymbol{W}^T\boldsymbol{W}(\boldsymbol{y} - \boldsymbol{H}\hat{\boldsymbol{x}}_0) \in T_{\hat{\boldsymbol{x}}_0}\mathcal{M},$$

the gradient is the projection of data fidelity term onto $T_{\hat{\boldsymbol{x}}_0}\mathcal{M}$,

This theorem suggests that in diffusion models, the naive measurement fidelity step (without considering the data manifold) pushes the inference path out of the manifolds and might lead to inaccurate reconstruction. On the other hand, our correction term from the manifold constraint guides the diffusion to lie on the data manifold, leading to better reconstruction. Such geometric views are illustrated in Fig. 2.

5 Experiments

For all tasks, we aim to verify the superiority of our method against other diffusion model-based approaches, and also against strong supervised learning-based baselines. Further details can be found in appendix D.

Datasets and Implementation For inpainting, we use FFHQ 256×256 [20], and ImageNet 256×256 [9] to validate our method. We utilize pre-trained models from the open sourced repository based on the implementation of ADM (VP-SDE) [10]. We validate the performance on 1000 held-out validation set images for both FFHQ and ImageNet dataset. For the colorization task, we use FFHQ, and LSUN-bedroom 256×256 [42]. We use pre-trained score functions from score-SDE [35] based on VE-SDE. We use 300 validation images for testing the performance with respect to the LSUN-bedroom dataset. For experiments with CT, we train our model based on `ncsnpp` as a VE-SDE from score-SDE [35], on the 2016 American Association of Physicists in Medicine (AAPM) grand challenge dataset, and we process the data as in [19]. Specifically, the dataset contains 3839 training images resized to 256×256 resolution. We simulate the CT measurement process with parallel beam geometry with evenly-spaced 180 degrees. Evaluation is performed on 421 held-out validation images from the AAPM challenge.

Inpainting Score-SDE [35], REPAINT [26], DDRM [21] were chosen as baseline diffusion models to compare against the proposed method. For a fair comparison, we use the same score function for all methods including MCG, and only differentiate the inference method that is used. We also include comparisons against supervised learning based baselines: LaMa [37], AOT-GAN [43], ICT [40], and DSI [28]. We use various forms of inpainting masks: box (128 × 128 sized square region is missing), extreme (only the box region is existent), random (90-95% of pixels are missing), and LaMa-thick.

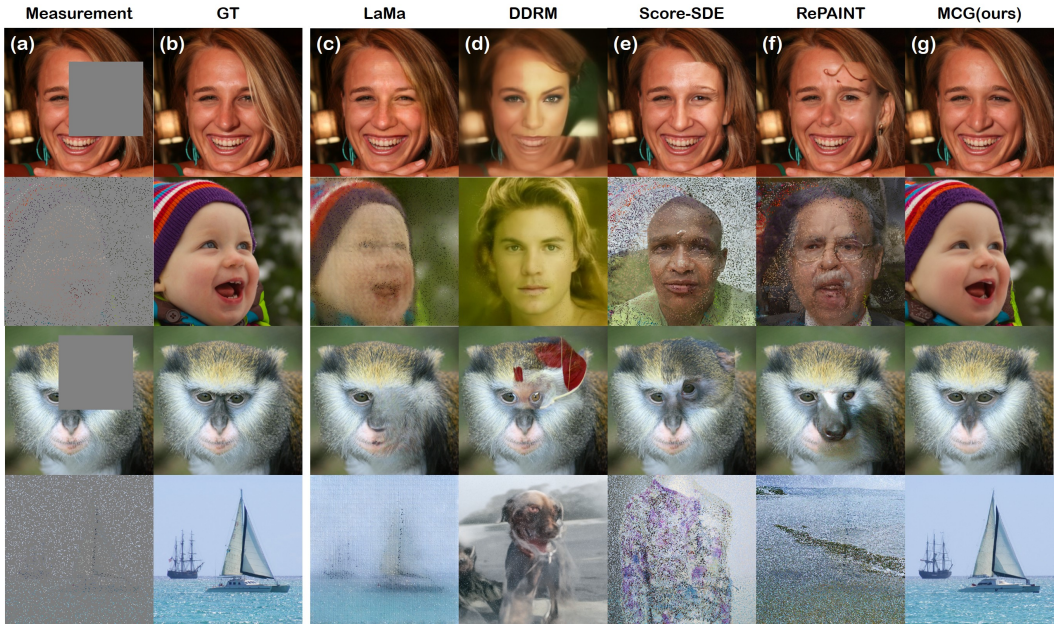

Figure 3: Inpainting results on FFHQ (1st, 2nd row) and ImageNet (3rd, 4th row). (a) Measurement, (b) Ground truth, (c) LaMa [37], (d) DDRM [21], (e) Score-SDE [35], (f) RePAINT [26], (g) MCG (Ours). Out of 256×256 image, the 1st and the 3rd row is masked with size 128×128 box. 92% of pixels (all RGB channels) from the images in the 2nd and 4th row are blocked.

	FFHQ (256 × 256)						**ImageNet (256 × 256)**					
	Box		**Random**		**Extreme**		**Box**		**Random**		**Wide masks**	
Method	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓
MCG (ours)	**23.7**	0.089	**21.4**	**0.186**	**30.6**	**0.366**	**25.4**	0.157	**34.8**	**0.308**	21.9	0.148
Score-SDE [35]	30.3	0.135	109.3	0.674	48.6	0.488	43.5	0.199	143.5	0.758	39.6	0.200
RePAINT* [26]	30.5	0.133	110.6	0.665	48.5	0.487	43.2	0.203	139.7	0.756	37.0	0.205
DDRM [21]	28.4	0.109	111.6	0.774	48.1	0.532	88.8	0.386	99.6	0.767	80.6	0.398
LaMa [37]	27.7	**0.086**	188.7	0.648	61.7	0.492	26.8	**0.139**	134.1	0.567	**20.4**	**0.140**
AOT-GAN [43]	29.2	0.108	97.2	0.514	69.5	0.452	35.3	0.163	119.6	0.583	29.8	0.161
ICT [40]	27.3	0.103	91.3	0.445	56.7	0.425	31.9	0.148	131.4	0.584	25.4	0.148
DSI [28]	27.9	0.096	126.4	0.601	77.5	0.463	34.5	0.155	132.9	0.549	24.3	0.154

Table 1: Quantitative evaluation (FID, LPIPS) of inpainting task on FFHQ and ImageNet. *: Re-implemented with our score function. MCG, Score-SDE, RePAINT, and DDRM all share the same score function and differ only in the inference method. **Bold**: Best, under: second best.

Quantitative evaluation is performed with two metrics - Frechet Inception Distance (FID)-1k [14], and Learned Perceptual Image Patch Similarity (LPIPS) [44].

Our method outperforms the diffusion model baselines [35, 26, 21] by a large margin. Moreover, our method is also competitive with, or even better than the best-in-class fully supervised methods, as can be seen in table 1. In fig. 3, we depict representative results that show the superiority of the method. For box-type inpainting, other diffusion model-based approaches all fail at reconstructing a feasible human face, clearly showing discrepancies inside and outside of the masked region. It is widely known that solving inverse problems on ImageNet is a much harder task due to the variability of the data, and for that matter even the SOTA supervised learning baselines fail to produce coherent reconstructions in the 3rd row. The random masking strategy that we adopt in the 2nd and the 4th row heavily limits the information available about the ground truth image, but our method can still faithfully recover the original. In contrast, the performance of the supervised learning baselines largely deteriorate due to the bias towards masks that were used during training. Diffusion model baselines also perform poorly due to the extremely limited measurement.

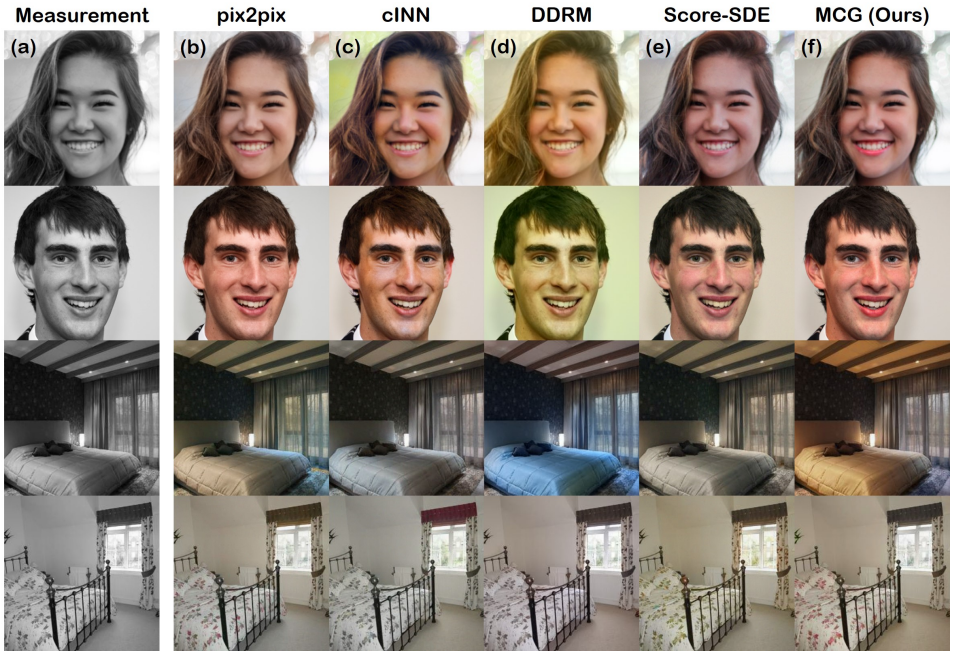

Figure 4: Colorization results on FFHQ and LSUN-bedroom. (a) Measurement, (b) pix2pix [17] (c) cINN [2] (d) DDRM [21], (e) Score-SDE [35], (f) MCG (ours).

Colorization We choose score-SDE [35], and DDRM [21] as diffusion-model based comparison methods, and also compare against cINN [2], and pix2pix [17]. Two metrics were used for evaluation: structural similarity index (SSIM), and LPIPS. Consistent with the findings from inpainting, we achieve much improved performance than score-SDE, and also is favorable against state-of-the-art (SOTA) superivsed learning based methods. We use the pre-trained model checkpoints whenever available, and train the model from scratch using the code from the official github repository when it is not. As can be seen in fig. 4, MCG tends to generate vibrant color throughout the structure, whereas other methods are either biased towards a single tone, or fails to capture the diversity. For example, when comparing the colorization results of the FFHQ dataset, ours is the only method that colors the lips red. In table 2, we see that the proposed method outperforms all other methods in terms of both PSNR/LPIPS in LSUN-bedroom, and also achieves strong performance in the colorization of FFHQ dataset.

CT reconstruction To the best of our knowledge, [34] is the only method that tackles CT reconstruction directly with diffusion models. We compare our method against [34], which we refer to as score-CT henceforth. We also compare with the best-in-class supervised learning methods, cGAN [12] and SIN-4c-PRN [41]. As a compressed sensing baseline, FISTA-TV [3] was included, along with the analytical reconstruction method, FBP. We use two standard metrics - peak-signal-to-noise-ratio (PSNR), and SSIM for quantitative evaluation.

Data	**FFHQ(256×256)**		**LSUN(256×256)**	
Method	SSIM ↑	LPIPS ↓	SSIM ↓	LPIPS ↓
MCG (ours)	<u>0.951</u>	**0.146**	**0.959**	**0.160**
Score-SDE [35]	0.936	0.180	0.945	0.199
DDRM [21]	0.918	0.326	<u>0.957</u>	<u>0.182</u>
cINN [2]	**0.952**	<u>0.166</u>	0.952	0.180
pix2pix [17]	0.935	0.184	0.947	0.174

Table 2: Quantitative evaluation (SSIM, LPIPS) of colorization task. **Bold**: best, <u>under</u>: second best.

From table 3, we see that the newly proposed MCG method outperforms the previous score-CT [34] by a large margin. We can observe the superiority of MCG over other methods more clearly in fig. 5, where MCG reconstructs the measurement with high fidelity, closely mimicking the ground truth. All other methods including the fully supervised baselines fall behind the proposed method.

Properties of MCG Our proposed method is fully unsupervised and is not trained on solving a specific inverse problem. For example, our box masks and random masks have very different forms of

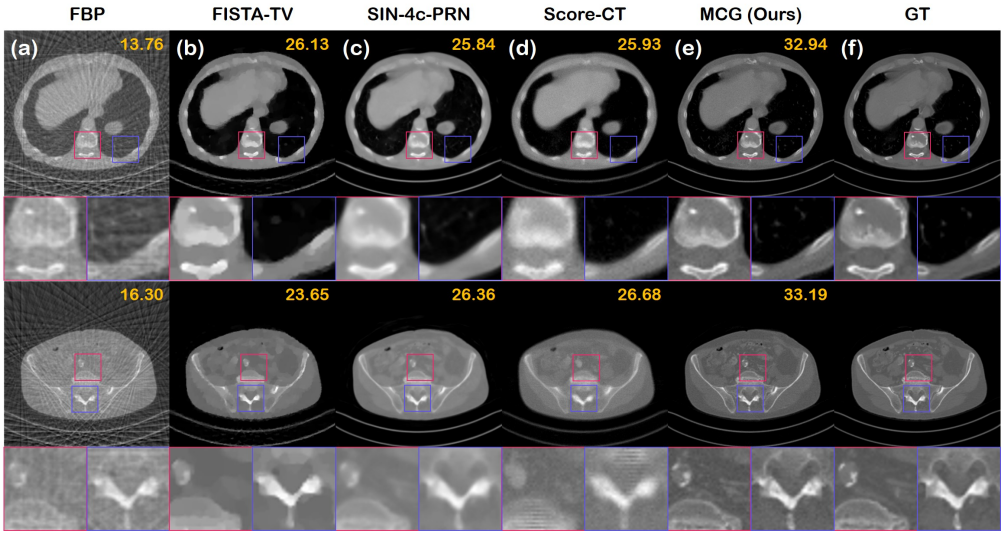

Figure 5: CT reconstruction results on AAPM dataset (30 view measurement). (a) FBP, (b) FISTA-TV [3], (c) SIN-4c-PRN [41], (d) Score-CT [34], (e) MCG (ours), and (f) ground truth (GT). Yellow numbers indicate the PSNR metric.

erasing the pixel values. Nevertheless, MCG generalizes perfectly well to such different measurement conditions, while other methods have a large performance gap between the different mask shapes. We further note two appealing properties of MCG as an inverse problem solver: 1) the ability to generate multiple solutions given a condition, and 2) the ability to maintain perfect measurement consistency. The former ability often lacks in supervised learning-based methods [37, 41], and the latter is often not satisfied for unsupervised GAN-based solutions [8, 4].

6 Conclusion

In this work, we proposed a general framework that can greatly enhance the performance of the diffusion model-based solvers for solving inverse problems. We showed several promising applications - inpainting, colorization, sparse view CT reconstruction, and showed that our method can outperform the current state-of-the-art methods. We analyzed our method theoretically and show that MCG prevents the data generation process from falling off the manifold, thereby reducing the errors that might accumulate at every step. Further, we showed that MCG controls the direction tangent to the data manifold, whereas the score function controls the direction that is normal, such that the two components complement each other.

	AAPM (256×256)			
Views	**18**		**30**	
Method	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
MCG (ours)	**33.57**	**0.956**	**36.09**	**0.971**
Score-POCS (ours)	30.77	0.907	32.68	0.923
Score-CT [34]	29.85	0.897	31.97	0.913
SIN-4c-PRN [41]	26.96	0.850	30.23	0.917
cGAN [12]	24.38	0.823	27.45	0.927
FISTA-TV [3]	21.57	0.791	23.92	0.861

Table 3: Quantitative evaluation (PSNR, SSIM) of CT reconstruction task. **Bold**: best.

Limitations and Broader Impact The proposed method is inherently stochastic since the diffusion model is the main workhorse of the algorithm. When the dimension m is pushed to low values, at times, our method fails to produce high quality reconstructions, albeit being better than the other methods overall. We note that our method is slow to sample from, inheriting the existing limitations of diffusion models. This would likely benefit from leveraging recent solvers aimed at accelerating the inference speed of diffusion models. In line with the arguments of other generative model-based inverse problem solvers, our method is a solver that relies heavily on the underlying diffusion model, and can thus potentially create malicious content such as deepfakes. Further, the reconstructions could intensify the social bias that is already existent in the training dataset.

References

[1] Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

[2] Lynton Ardizzone, Carsten Lüth, Jakob Kruse, Carsten Rother, and Ullrich Köthe. Guided image generation with conditional invertible neural networks. *arXiv preprint arXiv:1907.02392*, 2019.

[3] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[4] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. In *International Conference on Machine Learning*, pages 537–546. PMLR, 2017.

[5] Thorsten M Buzug. Computed tomography. In *Springer handbook of medical technology*, pages 311–342. Springer, 2011.

[6] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. ILVR: Conditioning method for denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[7] Hyungjin Chung, Byeongsu Sim, and Jong Chul 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.

[8] Giannis Daras, Joseph Dean, Ajil Jalal, and Alexandros G Dimakis. Intermediate layer optimization for inverse problems using deep generative models. In *International Conference on Machine Learning*, 2021.

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

[10] Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[11] Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

[12] Muhammad Usman Ghani and W Clem Karl. Deep learning-based sinogram completion for low-dose ct. In *2018 IEEE 13th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*, pages 1–5. IEEE, 2018.

[13] Richard Gordon, Robert Bender, and Gabor T Herman. Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography. *Journal of theoretical Biology*, 29(3):471–481, 1970.

[14] 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 I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.

[16] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022.

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[18] Zahra Kadkhodaie and Eero Simoncelli. Stochastic solutions for linear inverse problems using the prior implicit in a denoiser. In *Advances in Neural Information Processing Systems*, volume 34, pages 13242–13254. Curran Associates, Inc., 2021.

[19] Eunhee Kang, Junhong Min, and Jong Chul Ye. A deep convolutional neural network using directional wavelets for low-dose x-ray ct reconstruction. *Medical physics*, 44(10):e360–e375, 2017.

[20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[21] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *ICLR Workshop on Deep Generative Models for Highly Structured Data*, 2022.

[22] Daniil Kazantsev, Edoardo Pasca, Martin J Turner, and Philip J Withers. Ccpi-regularisation toolkit for computed tomographic image reconstruction with proximal splitting algorithms. *SoftwareX*, 9:317–323, 2019.

[23] Kwanyoung Kim and Jong Chul Ye. Noise2score: Tweedie's approach to self-supervised image denoising without clean images. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[24] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014*.

[25] Beatrice Laurent and Pascal Massart. Adaptive estimation of a quadratic functional by model selection. *Annals of Statistics*, pages 1302–1338, 2000.

[26] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. RePaint: Inpainting using Denoising Diffusion Probabilistic Models. *arXiv preprint arXiv:2201.09865*, 2022.

[27] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2017.

[28] Jialun Peng, Dong Liu, Songcen Xu, and Houqiang Li. Generating diverse structure for image inpainting with hierarchical VQ-VAE. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10775–10784, 2021.

[29] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[30] Herbert E Robbins. An empirical bayes approach to statistics. In *Breakthroughs in statistics*, pages 388–394. Springer, 1992.

[31] Simo Särkkä and Arno Solin. *Applied stochastic differential equations*, volume 10. Cambridge University Press, 2019.

[32] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015*.

[33] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *9th International Conference on Learning Representations, ICLR*, 2021.

[34] Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging with score-based generative models. In *International Conference on Learning Representations*, 2022.

[35] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *9th International Conference on Learning Representations, ICLR*, 2021.

[36] Charles M Stein. Estimation of the mean of a multivariate normal distribution. *The annals of Statistics*, pages 1135–1151, 1981.

[37] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2149–2159, 2022.

[38] Github PK Tool, Nov Sun Mon Tue Wed Thu, and Fri Sat. dkazanc/tomobar.

[39] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[40] Ziyu Wan, Jingbo Zhang, Dongdong Chen, and Jing Liao. High-fidelity pluralistic image completion with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4692–4701, 2021.

[41] Haoyu Wei, Florian Schiffers, Tobias Würfl, Daming Shen, Daniel Kim, Aggelos K Katsaggelos, and Oliver Cossairt. 2-step sparse-view ct reconstruction with a domain-specific perceptual network. *arXiv preprint arXiv:2012.04743*, 2020.

[42] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, 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.

[43] Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Aggregated contextual transformations for high-resolution image inpainting. *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[44] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[45] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] We discuss the limitations in (6).
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] We discuss potential negative impacts in (6).
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] We provide all proofs of results in supplementary material.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We include all code for our experiments in the supplementary material. We will release the code once the paper is published.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] Due to our limited resources we do not have time to run multiple sets of experiments.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes] We have cited the original works that released the datasets.

(b) Did you mention the license of the assets? [No] Licenses are standard and can be found online.

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We include our implementation as the supplementary material. We will release the code upon publication.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] All datasets used in our work are publicly available.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]