Solving Noisy Inverse Problems via Posterior Sampling: A Policy Gradient View-Point

Haoyue Tang[1], Tian Xie[1], Aosong Feng[2], Hanyu Wang[3]*, Chenyang Zhang[1] and Yang Bai[1]
[1]Meta AI, Sunnyvale, CA, USA
[2]Department of Electrical Engnineering, Yale University, New Haven, CT, USA
[3]Department of Computer Science, University of Maryland, College Park, MD, USA
{tanghaoyue13, tianxie, cyz, yba}@meta.com, hywang66@umd.edu, aosong.feng@yale.edu

Abstract

Solving image inverse problems (e.g., super-resolution and inpainting) requires generating a high fidelity image that matches the given input (the low-resolution image or the masked image). By using the input image as guidance, we can leverage a pretrained diffusion generation model to solve a wide range of image inversion tasks without task specific model fine-tuning. In this work, we propose diffusion policy gradient (DPG), a tractable computation method to estimate the score function given the guidance image. Our method is robust to both Gaussian and Poisson noise added to the input image, and improves the image restoration consistency and quality on FFHQ, ImageNet and LSUN datasets on both linear and non-linear image inversion tasks (inpainting, super-resolution, motion debluring, non-linear deblurring, etc.).

1 Introduction and Problem Formulation

Denoising Diffusion Probabilistic Models Ho et al. [2020], Sohl-Dickstein et al. [2015] provide tractable solutions to modeling a high quality image distribution. Their modeling and generation capabilities have been exploited in a wide range of image inverse problems Dhariwal and Nichol [2021], Blattmann et al. [2022], Rombach et al. [2021], Kawar et al. [2022], in which the goal is to generate a high quality image that matches the given input image. However, training a diffusion model from scratch is time-consuming. An alternative solution is to use the input image as guidance, and then generate the target image using a pretrained diffusion generation model through guided diffusion Ho and Salimans [2021], Dhariwal and Nichol [2021]. However, when the input guidance image is distorted by random noise and becomes inaccurate, solving image inversion problems becomes extremely challenging.

We now describe the noisy image inverse problem in more details. Suppose $\mathbf{x}_0$ represents a high quality image and let $p_0(\mathbf{x}_0)$ be its distribution. Let $\mathbf{y}$ be a noisy input image, which is obtained by feeding a high quality image $\mathbf{x}_0$ through an operator $\mathcal{A}$, i.e.,

$$\mathbf{y} = \mathcal{A}(\mathbf{x}_0) + \mathbf{n}, \tag{1}$$

where $\mathbf{n}$ is the distorted random noise. The operator $\mathcal{A}$ depends on the image inversion tasks. Notice that the operator $\mathcal{A}$ is often low-rank or invertible, making the computation of the inverse of $\mathbf{y}$ impossible.

An alternative solution to find $\mathbf{x}_0$ is to use the distribution $p_0(\mathbf{x}_0)$ as a priori, and compute and sample from the distribution $p_0(\mathbf{x}_0|\mathbf{y}) = \frac{p_0(\mathbf{x}_0)p_0(\mathbf{y}|\mathbf{x}_0)}{p(\mathbf{y})} \propto p_0(\mathbf{x}_0)p_0(\mathbf{y}|\mathbf{x}_0)$. Information about the prior

*Work done during an internship at Meta AI

DLDE-III Workshop in the 37th Conference on Neural Information Processing Systems (NeurIPS 2023).

$p_0(\mathbf{x}_0)$, i.e., the distribution of the high quality images can be obtained via an image generative model such as GAN and diffusion models. To solve image inverse problems for given input image $\mathbf{y}$ using diffusion models, we first revisit the forward diffusion process that turns $p_0(\mathbf{x}_0|\mathbf{y})$ into a Gaussian, i.e., $\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}\mathbf{w}, t \in [0, T]$, where $\beta(t) : [0, T] \mapsto \mathbb{R}^+$ is a monotonicallly increasing function and $\mathbf{w}$ is a Wiener process. Let $p_{t|0}(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t|\sqrt{\overline{\alpha}(t)}\mathbf{x}_0, (1 - \overline{\alpha}(t)\mathbf{I})$ be the conditional density function of $\mathbf{x}_t$ given $\mathbf{x}_0$, where $\overline{\alpha}(t) := \exp\left(-\int_0^t \beta(s)\mathrm{d}s\right)$. Sampling from $p_0(\mathbf{x}_0|\mathbf{y})$ requires running the reverse of the forward diffusion process, i.e., running the following SDE by starting from $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$:

$$\mathrm{d}\mathbf{x} = \left[-\frac{\beta(t)}{2}\mathbf{x} - \beta(t)\nabla_{\mathbf{x}} \log p_t(\mathbf{x}|\mathbf{y})\right]\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}\mathbf{w}. \tag{2}$$

Related Work To sample from equation 2, we need to know the score function of the $\mathbf{s}_t(\mathbf{x}_t, \mathbf{y}) := \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{y})$, which can be decomposed by:

$$\mathbf{s}_t(\mathbf{x}_t, \mathbf{y}) := \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{y}) = \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t, \mathbf{y}) = \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) + \nabla_{\mathbf{x}_t} \log p_t(\mathbf{y}|\mathbf{x}_t). \tag{3}$$

We notice that the first term in equation 3 is known by the pretrained diffusion model $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) = \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)$. The challenge is to compute the second term $\nabla_{\mathbf{x}} \log p_t(\mathbf{x}_t|\mathbf{y})$. There are currently two lines of work in utilizing diffusion models to solve image inverse problems. The first line of work utilizes the low rank structure of the operator $\mathcal{A}$, and directly plugs the known information $\mathbf{y}$ into the image generation process. SDEdit, Blended Diffusion and DiffEdit Meng et al. [2022], Avrahami et al. [2022, 2023], Couairon et al. [2023] solves image inpainting and editing tasks by plugging $\mathbf{y}$ directly into the pixel space. To solve a wider range of tasks such as super-resolution and deblurring, researchers further decompose $\mathcal{A}$ using the singular value decomposition (SVD) Song et al. [2021], Wang et al. [2023], Kawar et al. [2022], and plug the known information $\mathbf{y}$ into the spectral space of $\mathbf{x}_0$. However, those plug-in approaches can only work for linear inverse problems, and each tasks require a SVD decomposition of the operator $\mathcal{A}$. To solve a wider range of non-linear image inversion problems, another line of research use the input image $\mathbf{y}$ as a guidance, and generate the target image $\mathbf{x}_0$ using guided diffusion Chung et al. [2022, 2023], Meng and Kabashima [2022], Song et al. [2023c,a], Rout et al. [2023], Song et al. [2023b], Hu et al. [2023]. The challenge in input image guided approaches is to estimate the guidance score function $\nabla_{\mathbf{x}_t}\mathbb{E}_{p_{0|t}(\mathbf{x}_0|\mathbf{x}_t)}\left[p_0(\mathbf{y}|\mathbf{x}_0)\right]$ in each diffusion generation step t, where $\mathbf{x}_t$ is the intermediate steps of the generation process.

Our Contributions. To improve the image restoration quality and consistency, we propose a new method to estimate the score function $\nabla_{\mathbf{x}} \log p_t(\mathbf{y}|\mathbf{x}_t)$. Our contributions are summarized as follows:

(1) By viewing each noisy image as a policy and let the predicted clean image be a state that is selected by the policy, we propose diffusion policy gradient (DPG), a new method to estimate the score function given the input image $\mathbf{y}$.

(2) DPG does not need to compute a closed form psuedo-inverse or the spectral decomposition. With a pretrained diffusion generation model, we can solve a wide range of image inverse problems without model fine-tuning.

(3) Theoretically, the score function estimated by DPG is more accurate than DPS in the initial stages of the generation process. In experiments, DPG can restore more high-frequency details of the images. Quantitative evaluations on FFHQ, ImageNet and LSUN image restoration tasks show that the proposed method achieves performance improvement in both image restoration quality and consistency.

2 Methodology

2.1 Computing $\nabla_{\mathbf{x}} \log p_t(\mathbf{y}|\mathbf{x}_t)$ as Policy Gradient

We first decompose the second term $\nabla_{\mathbf{x}} \log p_t(\mathbf{x}_t|\mathbf{y})$ as follows:

$$\nabla_{\mathbf{x}_t} \log p_t(\mathbf{y}|\mathbf{x}_t) \propto \nabla_{\mathbf{x}_t} \int \underbrace{p_{0|t}(\mathbf{x}_0|\mathbf{x}_t)}_{\text{State Density Function}} \underbrace{p_0(\mathbf{y}|\mathbf{x}_0)}_{\text{Cost}} \mathrm{d}\mathbf{x}_0 =: \tilde{\mathbf{s}}_t(\mathbf{x}_t, \mathbf{y}). \tag{4}$$

We notice that the generated image $\mathbf{x}_0$ is determined by the intermediate noisy image $\mathbf{x}_t$, and the conditional probability $p_{0|t}(\mathbf{y}|\mathbf{x}_0)$ is highly related to the reconstruction cost between $\mathbf{y}$ and the predicted image $\mathbf{x}_0$. Moreover, the score function $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{y})$ is the gradient direction of the expected cost function $\int p_{0|t}(\mathbf{x}_0|\mathbf{x}_t)p_0(\mathbf{y}|\mathbf{x}_0)\mathrm{d}\mathbf{x}_0$. Therefore, the computation of the score function equation 4 is closely related to policy gradient in reinforcement learning, where $p_t(\mathbf{x}_t|\mathbf{x}_0)$ is the state occupation measure by choosing policy $\mathbf{x}_t$, and $p_0(\mathbf{y}|\mathbf{x}_0)$ is the cost. The following theorem enables us to compute the score function equation 4 from the policy gradient perspective:

Theorem 1 (Leibniz Rule) *For almost all $t \in [0, T]$, we can compute the score function $\tilde{\mathbf{s}}_t(\mathbf{x}_t, \mathbf{y})$ from equation 4 as follows:*

$$\tilde{\mathbf{s}}_t(\mathbf{x}_t, \mathbf{y}) = \mathbb{E}_{p_{0|t}(\mathbf{x}_0|\mathbf{x}_t)}\left[p_0(\mathbf{y}|\mathbf{x}_0)\nabla_{\mathbf{x}_t} \log p_{0|t}(\mathbf{x}_0|\mathbf{x}_t)\right] \tag{5}$$

Proof of Theorem 1 is provided in Appendix 4.1.

2.2 Implementation Details

Tractable Monte Carlo sampling Computing the score function in equation 5 requires sampling from $p_{0|t}(\mathbf{x}_0|\mathbf{x}_t)$, but the closed form distribution $p_{0|t}(\mathbf{x}_0|\mathbf{x}_t)$ is hard to compute. To approximate $\log p_{0|t}(\mathbf{x}_0|\mathbf{x}_t)$, similar to Chung et al. [2023], Song et al. [2023c], we select $q_{0|t}(\mathbf{x}_0|\mathbf{x}_t) = \mathcal{N}(\hat{\mathbf{x}}_0(\mathbf{x}_t), r_t^2\mathbf{I})$ to be a Gaussian distribution, where the mean $\hat{\mathbf{x}}_0(\mathbf{x}_t)$ is obtained by the Tweedie's estimation Efron [2011], Kim and Ye [2021], $\hat{\mathbf{x}}_0(\mathbf{x}_t) = \frac{1}{\sqrt{\overline{\alpha}_t}}\left(\mathbf{x}_t - \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\right)$. We select the variance $r_t = \frac{1}{C\times H\times W}\ell_{\mathbf{y}}(\mathbf{x})$, where C, H, W are the channels, height and weight of the $\mathbf{x}_0$ and $\ell_{\mathbf{y}}(\mathbf{x})$ is the reconstruction loss between $\mathbf{y}$ and the reconstructed image $\hat{\mathbf{x}}_0$. Then, by drawing N samples $\{\mathbf{x}_0^{(1)}, \cdots, \mathbf{x}_0^{(N)}\}$ from distribution $q_{0|t}(\mathbf{x}_0|\mathbf{x}_t)$, we can approximate the score function $\tilde{\mathbf{s}}_t(\mathbf{x}_t, \mathbf{y})$ in equation 5 via the Monte Carlo (MC) method $\mathbb{E}_{q_{0|t}(\mathbf{x}_0|\mathbf{x}_t)}\left[p_0(\mathbf{y}|\mathbf{x}_0)\nabla_{\mathbf{x}_t} \log q_{0|t}(\mathbf{x}_0|\mathbf{x}_t)\right] \approx -\frac{1}{2r_t^2 N}\sum_{i=1}^{N}\left(p_0(\mathbf{y}|\mathbf{x}_0^{(i)})\cdot\nabla_{\mathbf{x}_t}\|\mathbf{x}_0^{(i)} - \hat{\mathbf{x}}_0(\mathbf{x}_t)\|_2^2\right)$

Reward Shaping Similar to policy gradient in reinforcement learning, direct MC estimation of the policy gradient from suffers from high variance and low convergence rate. We leverage reward shaping Ng et al. [1999] by computing a bias term $b := \mathbb{E}_{p_{0|t}(\mathbf{x}_0|\mathbf{x}_t)}\left[p_0(\mathbf{y}|\mathbf{x}_0)\right]$ for each sample i using the leave-one-out cross-validation, $b^{(i)} := \frac{1}{N-1}\sum_{j=1, j\neq i}^{N} p_0\left(\mathbf{y}|\mathbf{x}_0^{(i)}\right)$. We can then improve the MC estimation by $\tilde{\mathbf{s}}_t(\mathbf{x}_t, \mathbf{y}) = -\frac{1}{2r_t^2 N}\sum_{i=1}^{N}\left((p_0(\mathbf{y}|\mathbf{x}_0^{(i)}) - b^{(i)})\times\nabla_{\mathbf{x}_t}\|\mathbf{x}_0^{(i)} - \hat{\mathbf{x}}_0(\mathbf{x}_t)\|_2^2\right)$

Score Function Normalization Notice that the score function computed after reward shaping contains only direction information. The exact norm of the gradient $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{y}|\mathbf{x}_t)$ is unknown. We observe from the classifier free conditional generation experiments that the norm of the conditional score function is almost the same as the score of the unconditional generation score, and the norm is stable during the whole diffusion inference process. Therefore, we simply rescale the computed gradient into a vector with norm C, i.e., assume that $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{y}|\mathbf{x}_t) \approx C\cdot\frac{1}{\|\tilde{\mathbf{s}}_t(\mathbf{x}_t,\mathbf{y})\|_2^2}\tilde{\mathbf{s}}_t(\mathbf{x}_t, \mathbf{y})$ and plug it into equation 3 to compute the score function $\mathbf{s}(\mathbf{x}_t, \mathbf{y})$, i.e.,

$$\mathbf{s}_t(\mathbf{x}_t, \mathbf{y}) \approx \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) + C\cdot\frac{\tilde{\mathbf{s}}_t(\mathbf{x}_t, \mathbf{y})}{\|\tilde{\mathbf{s}}_t(\mathbf{x}_t, \mathbf{y})\|_2^2}. \tag{6}$$

Algorithm 1 Diffusion Policy Gradient (DPG)

Require: T, $\mathbf{y}$, $\ell(\mathbf{y}, \mathcal{A}(\cdot))$
  $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$
  for $t = T-1$ to 0 **do**
    $\hat{\mathbf{x}}_0 \leftarrow \frac{1}{\sqrt{\overline{\alpha}_t}}\left(\mathbf{x}_t + (1-\overline{\alpha}_t\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\right)$
    $r_t \leftarrow \frac{1}{C\times H\times W}\ell_{\mathbf{y}}(\mathbf{x})$.
    $\boldsymbol{\xi}^{(i)} \sim \mathcal{N}(0, \mathbf{I}), \mathbf{x}_0^{(i)} \leftarrow \hat{\mathbf{x}}_0 + r_t\boldsymbol{\xi}^{(i)}, i = 1, \cdots, N_{\mathrm{mc}}$.
    $b^{(i)} \leftarrow \frac{1}{N_{\mathrm{mc}}-1}\sum_{j=1, j\neq i}^{N_{\mathrm{mc}}} p_0(\mathbf{y}|\mathbf{x}_0^{(i)})$
    $\tilde{\mathbf{s}}_t(\mathbf{x}_t, \mathbf{y}) \leftarrow \frac{1}{N_{\mathrm{mc}}}\sum_{i=1}^{N_{\mathrm{mc}}}(p_0(\mathbf{y}|\mathbf{x}_0^{(i)}) - b^{(i)})\nabla_{\mathbf{x}_t}\left(-\|\mathbf{x}_0^{(i)} - \hat{\mathbf{x}}_0(\mathbf{x}_t)\|_2^2\right)$
    $\mathbf{s}_t(\mathbf{x}_t, \mathbf{y}) \leftarrow \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) + C\frac{\tilde{\mathbf{s}}_t(\mathbf{x}_t,\mathbf{y})}{\|\tilde{\mathbf{s}}_t(\mathbf{x}_t,\mathbf{y})\|_2^2}$
    $\mathbf{x}_{t-1} \leftarrow$DDPM$(\mathbf{x}_t, \mathbf{s}_t(\mathbf{x}_t, \mathbf{y}))$.
  end for
  Return image $\mathbf{x}_0$

Using equation 6, we can solve the image inversion problems with the standard DDPM sampling method, which is displayed in Algorithm 1

3 Experiments

Experiment Setup Similar to Chung et al. [2023], Song et al. [2023b], we test the performance of our proposed algorithm on three datasets: the FFHQ 256×256 dataset Karras et al. [2019], the

ImageNet dataset Deng et al. [2009] and the LSUN-Bedroom dataset Yu et al. [2015]. We consider four types of image inversion tasks:

(1) Inpainting with a 128×128 masks added randomly on the figure; (2) 4×super-resolution with average pooling; (3) Gaussian deblurring with kernel size 61×61 and standard deviation of 3.0; (4) Motion deblurring with kernel size of 61 and intensity value 0.5 generated by[2]. We consider that the input image is noisy, i.e., Gaussian noise with variance $\sigma_{\mathbf{y}} = 0.05$ or Poisson noise with rate $\lambda = 1.0$ is added on the input image. For FFHQ experiments, we use the pretrained model from Chung et al. [2023] (trained on 4.9k images on FFHQ) and test the performance of 1k validation set; For Imagenet and LSUN experiments, we use the unconditional Imagenet and LSUN 256×256 generation model from Dhariwal and Nichol [2021] and the 1k images on ImageNet validation set [3] and the full LSUN-Bedroom validation set.

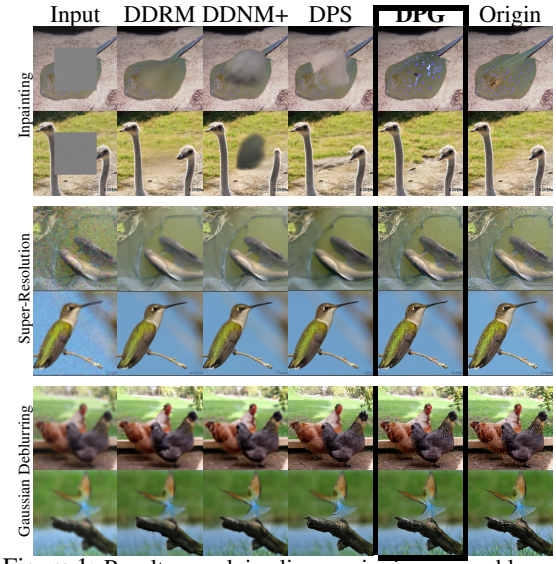

Figure 1: Results on solving linear noisy inverse problems with Gaussian noise $\sigma_{\mathbf{y}} = 0.05$ on ImageNet Dataset.

Evaluations We measure both the image restoration quality and consistency compared with the ground-truth image. For image restoration quality, we compute the Fréchet inception distance between the restored images and the training set; For image restoration consitency, we computes the LPIPS score Zhang et al. [2018] (VGG Net) between the restored image and the ground truth image. Quantative evaluation results are displayed in Table 3. Selected image restoration samples when the observation noise are Gaussian and Poisson are displayed in Fig. 1.

Table 1: Quantitative Results on Linear Inverse Problems with Gaussian Noise

Method	Inpainting		Super-Resolution		Deblur (Gauss)		Deblur (Motion)	
	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓
FFHQ 1k Validation Set								
DPG	**22.44**	0.181	**22.49**	**0.214**	**22.29**	**0.216**	**24.44**	**0.223**
DPS	33.12	0.216	39.35	**0.214**	44.05	0.257	39.02	0.242
DDRM	27.47	**0.172**	62.15	0.294	74.92	0.332	N/A	N/A
DDNM+	27.34	0.173	46.13	0.260	63.19	0.301	N/A	N/A
ImageNet 1k Validation Set								
DPG	41.86	0.258	**31.02**	**0.293**	**34.43**	**0.314**	**36.15**	**0.343**
DPS	45.95	0.267	43.60	0.340	62.65	0.434	56.08	0.386
DDRM	50.94	0.246	51.77	0.355	72.49	0.345	N/A	N/A
DDNM+	50.50	0.246	51.08	0.362	71.74	0.410	N/A	N/A
LSUN-Bedroom Validation Set								
DPG	**34.32**	0.218	**31.44**	**0.262**	**38.72**	**0.277**	**34.44**	**0.284**
DPS	35.91	0.218	37.42	0.284	48.10	0.320	50.09	0.358
DDRM	37.61	0.205	50.96	0.310	59.04	0.353	N/A	N/A
DDNM+	37.03	**0.204**	50.15	0.296	74.40	0.336	N/A	N/A

We compare the performance with the following methods: Denoising Diffusion Null Space models (DDNM+) Wang et al. [2023] for noisy problems, Diffusion Posterior Sampling (DPS) Chung et al. [2022], Denoising Diffusion Restoration Models (DDRM) Kawar et al. [2022] and the Psuedo-Inverse guided diffusion methods (ΠGDM) Song et al. [2023b] and the Reddiff Mardani et al. [2023]. The key paremeters for different methods are displayed in Appendix 6.

Analysis First, the FID and LPIPS score of our proposed DPG method is smaller than DPS method, indicating that DPG has a better image restoration quality than DPS method. This is because the estimation of the score function by DPG is more accurate than DPS, especially in the initial stages of the diffusion generation process. Therefore, the shape and structure of the image can be recovered in an earlier stage of the diffusion process, this gives room to recover high frequency details in later stage of the image generation. Notice that DDNM+ and DDRM uses a plug-in estimation, i.e., the known pixels in $\mathbf{y}$ are directly used in the generation process. Therefore, the PSNR of DDNM+ and DDRM are better than the proposed DPG method, but the recovered high frequency details are less.

4 Conclusions

In this paper, we proposed a new method to estimate the score function for solving image inversion problems. Our method is robust when the input image is perturbed by random noise, and can be used for solving non-linear inverse problems such as non-linear deblurring. Experiments demonstrate that the proposed method can improve image restoration quality in both human eye evaluation and

[2] `https://github.com/LeviBorodenko/motionblur`

[3] `https://github.com/XingangPan/deep-generative-prior/blob/master/scripts/imagenet_val_1k.txt`

quantitative metrics. In the future, we will test the performance of DPG method on non-differentiable image inverse tasks such as JPEG restoration.

References

Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18208–18218, June 2022.

Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *ACM Trans. Graph.*, 42(4), jul 2023. ISSN 0730-0301. doi: 10.1145/3592450. URL https://doi.org/10.1145/3592450.

Andreas Blattmann, Robin Rombach, Kaan Oktay, and Björn Ommer. Retrieval-augmented diffusion models, 2022. URL https://arxiv.org/abs/2204.11824.

Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints. In *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=nJJjv0JDJju.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=OnD9zGAGT0k.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=3lge0p5o-M-.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

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

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

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL https://openreview.net/forum?id=qw8AKxfYbI.

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

Yujie Hu, Yinhuai Wang, and Jian Zhang. Dear-gan: Degradation-aware face restoration with gan prior. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9):4603–4615, 2023. doi: 10.1109/TCSVT.2023.3244786.

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.

Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Advances in Neural Information Processing Systems*, 2022.

Kwanyoung Kim and Jong Chul Ye. Noise2score: Tweedie's approach to self-supervised image denoising without clean images. In *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=ZqEUs3sTRU0.

Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models. *arXiv preprint arXiv:2305.04391*, 2023.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=aBsCjcPu_tE.

Xiangming Meng and Yoshiyuki Kabashima. Diffusion model based posterior sampling for noisy linear inverse problems. *arXiv preprint arXiv:2211.12343*, 2022.

Andrew Y. Ng, Daishi Harada, and Stuart J. Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning*, ICML '99, page 278–287, San Francisco, CA, USA, 1999.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021.

Litu Rout, Negin Raoof, Giannis Daras, Constantine Caramanis, Alexandros G Dimakis, and Sanjay Shakkottai. Solving linear inverse problems provably via posterior sampling with latent diffusion models. *arXiv preprint arXiv:2307.00619*, 2023.

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

Bowen Song, Soo Min Kwon, Zecheng Zhang, Xinyu Hu, Qing Qu, and Liyue Shen. Solving inverse problems with latent diffusion models via hard data consistency, 2023a.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=9_gsMA8MRKQ.

Jiaming Song, Qinsheng Zhang, Hongxu Yin, Morteza Mardani, Ming-Yu Liu, Jan Kautz, Yongxin Chen, and Arash Vahdat. Loss-guided diffusion models for plug-and-play controllable generation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 32483–32498. PMLR, 23–29 Jul 2023c. URL https://proceedings.mlr.press/v202/song23k.html.

Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging with score-based generative models. In *NeurIPS 2021 Workshop on Deep Learning and Inverse Problems*, 2021. URL https://openreview.net/forum?id=4rFAhgrA0lA.

Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. *The Eleventh International Conference on Learning Representations*, 2023.

Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.