Denoising Diffusion Models for Plug-and-Play Image Restoration

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

NTIRE 2023

Yuanzhi Zhu[1], Kai Zhang[1,], Jingyun Liang[1], Jiezhang Cao[1], Bihan Wen[2], Radu Timofte[1,3], Luc Van Gool[1,4]
[1]ETH Zürich [2]Nanyang Technological University [3]University of Würzburg [4]KU Leuven

Background

Plug-and-play Image Restoration (IR) has been widely recognized as a flexible and interpretable method for solving various inverse problems by utilizing any off-the-shelf denoiser as the implicit image prior. However, most existing methods focus on discriminative Gaussian denoisers. Although diffusion models have shown impressive performance for high-quality image synthesis, their potential to serve as a generative denoiser prior to the plug-and-play IR methods remains to be further explored.

Methods

In the previously published research by Zhang *et al.* [3], the Half-Quadratic-Splitting (HQS) algorithm is employed to iteratively tackle the Maximum A Posteriori (MAP) problem $\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \frac{1}{2\sigma_n^2}\|\mathbf{y}-\mathcal{H}(\mathbf{x})\|^2 + \lambda\mathcal{P}(\mathbf{x})$ within the context of IR tasks. The algorithm's process can be stated mathematically as follows:

$$\begin{cases} \mathbf{z}_k = \arg\min\limits_{\mathbf{z}} \dfrac{1}{2(\sqrt{\lambda/\mu})^2}\|\mathbf{z}-\mathbf{x}_k\|^2 + \mathcal{P}(\mathbf{z}) \\ \mathbf{x}_{k-1} = \arg\min\limits_{\mathbf{x}} \|\mathbf{y}-\mathcal{H}(\mathbf{x})\|^2 + \mu\sigma_n^2\|\mathbf{x}-\mathbf{z}_k\|^2. \end{cases}$$

To harness the generative power of diffusion models, we adopt the diffusion sampling schedule, converting the original optimization methodology into a sampling technique. This introduces a novel plug-and-play sampling framework, defined as follows:

$$\begin{cases} \mathbf{x}_0^{(t)} = \arg\min\limits_{\mathbf{z}} \dfrac{1}{2\bar{\sigma}_t^2}\|\mathbf{z}-\mathbf{x}_t\|^2 + \mathcal{P}(\mathbf{z}) \\ \hat{\mathbf{x}}_0^{(t)} = \arg\min\limits_{\mathbf{x}} \|\mathbf{y}-\mathcal{H}(\mathbf{x})\|^2 + \rho_t\|\mathbf{x}-\mathbf{x}_0^{(t)}\|^2 \\ \mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{\mathbf{x}}_0 + \sqrt{1-\bar{\alpha}_{t-1}}(\sqrt{1-\zeta}\hat{\epsilon} + \sqrt{\zeta}\epsilon_t) \end{cases}$$

Compared to the DDIM sampling framework, our method introduces an additional data subproblem. This extra component serves to guide the sampling process towards the posterior distribution $p(\mathbf{x}|\mathbf{y})$.

Qualitative Results

Sampling Framework

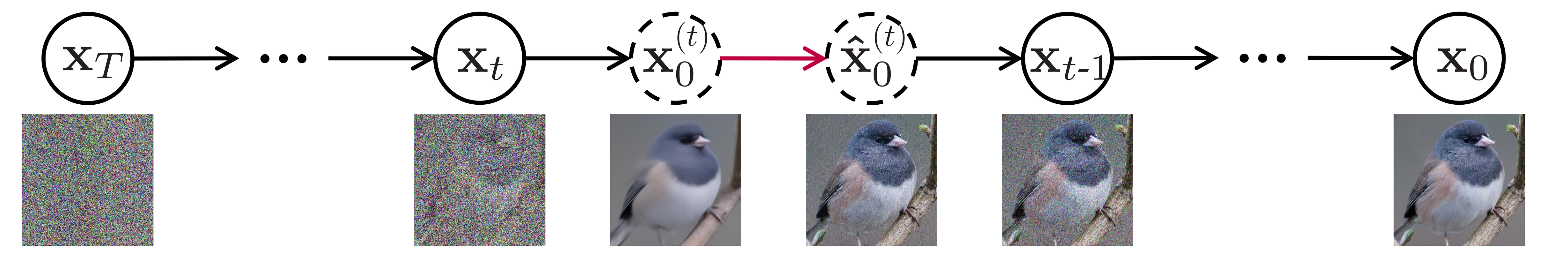

Illustration of our sampling method. For every state $\mathbf{x}_t$, following the prediction of the estimated $\mathbf{x}_0^{(t)}$ by the diffusion model, the measurement $\mathbf{y}$ is incorporated by solving the data proximal subproblem (indicated by the red arrow). Subsequently, the next state $\mathbf{x}_{t-1}$ is derived by adding *i.i.d* noise back and thus completing one step of reverse diffusion sampling.

Quantitative Results

FFHQ	$\sigma_n = 0.05$	Deblur (Gaussian)			Deblur (motion)			SR (×4)		
Method	NFEs ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓
DiffPIR	100	27.36	**59.65**	**0.236**	**26.57**	**65.78**	**0.255**	26.64	**65.77**	0.260
DPS [1]	1000	25.46	65.57	0.247	23.31	73.31	0.289	25.77	67.01	**0.256**
DDRM [2]	20	25.93	101.89	0.298	-	-	-	27.92	89.43	0.265
DPIR [3]	>20	**27.79**	123.99	0.450	26.41	146.44	0.467	**28.03**	133.39	0.456

ImageNet	$\sigma_n = 0.05$	Deblur (Gaussian)			Deblur (motion)			SR (×4)		
Method	NFEs ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓
DiffPIR	100	22.80	**93.36**	**0.355**	**24.01**	**124.63**	**0.366**	23.18	**106.32**	0.371
DPS [1]	1000	19.58	138.80	0.434	17.75	184.45	0.491	22.16	114.93	0.383
DDRM [2]	20	22.33	160.73	0.427	-	-	-	23.89	118.55	**0.358**
DPIR [3]	>20	**23.86**	189.92	0.476	23.60	210.31	0.489	**23.99**	204.83	0.475

FFHQ	$\sigma_n = 0.0$	Inpaint (box)		Inpaint (random)			Deblur (Gaussian)			Deblur (motion)			SR (×4)		
Method	NFEs ↓	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓	PSNR ↑	FID ↓	LPIPS ↓
DiffPIR	20	35.72	0.117	34.03	30.81	0.116	30.74	46.64	0.170	37.03	20.11	0.084	29.17	58.02	0.187
DiffPIR	100	**25.64**	**0.107**	**36.17**	**13.68**	**0.066**	**31.00**	**39.27**	**0.152**	37.53	**11.54**	**0.064**	29.52	**47.80**	**0.174**
DPS [1]	1000	43.49	0.145	34.65	33.14	0.105	27.31	51.23	0.192	26.73	58.63	0.222	27.64	59.06	0.209
DDRM [2]	20	37.05	0.119	31.83	56.60	0.164	28.40	67.99	0.238	-	-	-	30.09	68.59	0.188
DPIR [3]	>20	-	-	-	-	-	30.52	96.16	0.350	**38.39**	27.55	0.233	**30.41**	96.16	0.362

References

[1] Hyungjin Chung and et al. Diffusion posterior sampling for general noisy inverse problems. *in ICLR*, 2023.

[2] Bahjat Kawar and et al. Denoising diffusion restoration models. *in NeurIPS*, 2022.

[3] Kai Zhang and et al. Plug-and-play image restoration with deep denoiser prior. *in IEEE PAMI*, 2021.

More Information

Web https://yuanzhi-zhu.github.io/about/
Email yuazhu@student.ethz.ch
kai.zhang@vision.ee.ethz.ch
Arxiv https://arxiv.org/abs/2305.08995
GitHub https://github.com/yuanzhi-zhu/DiffPIR
WeChat yuanzhi-zhu

Diffusion Art QR Code