

#### 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  $\hat{\mathbf{x}}$  =  $\arg\min_{\mathbf{x}} \frac{1}{2\sigma_{\mathbf{x}}^2} \|\mathbf{y} - \mathcal{H}(\mathbf{x})\|^2 + \lambda \mathcal{P}(\mathbf{x})$  within the context of IR tasks. The algorithm's process can be stated mathematically as follows:

$$\begin{cases} \mathbf{z}_{k} = \arg\min_{\mathbf{z}} \frac{1}{2(\sqrt{\lambda/\mu})^{2}} \|\mathbf{z} - \mathbf{x}_{k}\|^{2} + \mathcal{P}(\mathbf{z}) \\ \mathbf{x}_{k-1} = \arg\min_{\mathbf{x}} \|\mathbf{y} - \mathcal{H}(\mathbf{x})\|^{2} + \mu \sigma_{n}^{2} \|\mathbf{x} - \mathbf{z}_{k}\|^{2} \end{cases}$$

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:

$$\begin{cases} \mathbf{x}_{0}^{(t)} = \arg\min_{\mathbf{z}} \frac{1}{2\bar{\sigma}_{t}^{2}} \|\mathbf{z} - \mathbf{x}_{t}\|^{2} + \mathcal{P}(\mathbf{z}) \\ \mathbf{\hat{x}}_{0}^{(t)} = \arg\min_{\mathbf{x}} \|\mathbf{y} - \mathcal{H}(\mathbf{x})\|^{2} + \rho_{t} \|\mathbf{x} - \mathbf{x}_{0}^{(t)}\|^{2} \\ \mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \mathbf{\hat{x}}_{0} + \sqrt{1 - \bar{\alpha}_{t-1}} (\sqrt{1 - \zeta} \hat{\epsilon} + \sqrt{\zeta} \epsilon_{t}) \end{cases}$$

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  $p(\mathbf{x}|\mathbf{y})$ .

# DENOISING DIFFUSION MODELS FOR PLUG-AND-PLAY IMAGE RESTORATION

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

### QUALITATIVE RESULTS

R 0.05)







REFERENCES

*ICLR*, 2023.







































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

[2] Bahjat Kawar and et al. Denoising diffusion

restoration models. in NeuIPS, 2022.

| Hyungjin Chung and et al. Diffusion posterior

sampling for general noisy inverse problems. in



**Illustration of our sampling method.** For every state  $x_t$ , following the prediction of the estimated  $x_0^{(t)}$ by the diffusion model, the measurement y is incorporated by solving the data proximal subproblem (indicated by the red arrow). Subsequently, the next state  $x_{t-1}$  is derived by adding *i.i.d* noise back and thus completing one step of reverse diffusion sampling.

## QU.

FFH Met Diff DPS DD DPI

Ima Met Diff

DPS DD DP 

Meth DiffP DiffP

DPS [ DDR DPIR





#### SAMPLING FRAMEWORK

AN	TITA	TIV	E RE	ESUL	TS											
IQ	$\sigma_n$	$\sigma_n = 0.05$		Deblu	ır (Gaus	sian)	Deblur (mo			tion)		<b>SR (</b> ×4)				
thod	N	NFEs $\downarrow$		$PSNR \uparrow F$		$D \downarrow LPIPS \downarrow$		$PSNR \uparrow FI$		LPIPS $\downarrow$		PSNR ↑	FID	$FID \downarrow L$		
PIR		100	27.	.36	59.65	0.236	2	26.57	65.78	0.255		26.64	65.77		0.260	
5[1]	[1] 1000		25	.46	65.57	0.247	2	23.31	73.31	0.289		25.77	67.01		0.256	
RM [	<b>м</b> [2] 20		25.	25.93 101.89 0.298					_		27.92	7.92 89.43		0.265		
R [3]		>20		<b>27.79</b> 1		123.99 0.450		26.41		146.44 0.467		28.03	133.39		0.456	
igeN	et $\sigma_n$	$\sigma_n = 0.05$		Deblur (Gaussian)				Deblur (motion)					<b>SR (</b> ×4)			
thod	Ν	NFEs $\downarrow$		$PSNR \uparrow FID \downarrow$		LPIPS	$\downarrow$ PS	SNR↑	$FID\downarrow$	$\downarrow  LPIPS \downarrow$		$PSNR \uparrow FID \downarrow$		↓ ]	$\downarrow$ LPIPS $\downarrow$	
PIR		100 22		.80	93.36 0.355		24.01		124.63	<b>0.366</b>		23.18	106.32		0.371	
5[1]	1] 1000		19	19.58 138.80 0.434		]	17.75 184.45		0.491		22.16	114.9	93	0.383		
RM [	RM [2] 20		22	22.33 160		0.427				_		23.89	89 118.55		0.358	
[R [3]		>20 23.86		.86	189.92	0.476		23.60	3.60 210.31		0.489		<b>23.99</b> 204.8		3 0.475	
<u>)</u>	$\sigma_n = 0.0$	= 0.0 <b>Inpaint (box)</b>		) Inpaint (random)			Del	blur (Gaus	Del	olur (mo	tion) SR		SR (×4	<b>K (</b> ×4)		
od	NFEs $\downarrow$	$\overline{\text{FID}}\downarrow$	LPIPS $\downarrow$	PSNR ↑	$FID\downarrow$	LPIPS ↓	PSNR↑	$FID\downarrow$	LPIPS $\downarrow$	PSNR↑	FID ↓	LPIPS $\downarrow$	$PSNR\uparrow$	$FID\downarrow$	LPIPS .	
[R	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	
R	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	
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	
VI [2] [3]	20 >20	37.05	0.119	31.83	56.6U -	0.164 -	28.40 30 52	67.99 96.16	0.238	-	- 27 55	- 0.233	30.09 <b>30 41</b>	68.59 96.16	0.188	

AN	TITA	TIV	E RI	ESUL	TS												
IQ	$\sigma_n$	$\sigma_n = 0.05$		Deblur (Gaussian)				Deblur (motion)					<b>SR (</b> ×4)				
thod	N	$FEs \downarrow$	PSI	NR ↑	FID ↓	LPIPS	$\downarrow$ PS	SNR ↑	$FID\downarrow$	LPIPS ↓		PSNR ↑	FID	↓ ]	$\Box PIPS \downarrow$		
PIR		100	27	7.36	59.65	0.236	r 2	26.57	65.78	0.255		26.64	65.77		0.260		
5 [1] RM [2 R [3]	2]	1000 20 >20	25 25 27	5.46 5.93 7 <b>.79</b>	65.570.247101.890.298123.990.450			23.31 - 26.41	73.31       0.289         146.44       0.467		89 67	25.77 27.92 <b>28.03</b>	67.01 89.43 133.39		<b>0.256</b> 0.265 0.456		
ageNo	et $\sigma_n$	= 0.05	5	Deblu	eblur (Gaussian)			Del	olur (mo		<b>SR (</b> ×4)						
thod	Ν	IFEs ↓	$\downarrow$ PSNR $\uparrow$		FID↓ LPIPS		$\downarrow$ PSNR $\uparrow$		FID ↓	$\downarrow$ LPIPS $\downarrow$		PSNR↑	FID	↓ I	$\square PIPS \downarrow$		
fPIR		100	22	2.80	93.36	0.355	24.01		124.63	3 0.366		23.18	106.32		0.371		
5 [1] RM [2 IR [3]	2]	1000 20 >20	19.58 22.33 <b>23.86</b>		138.80 160.73 189.92	$0.434 \\ 0.427 \\ 0.476$	17.75 - 23.60		184.45 - 210.31	0.491 - 0.489		22.16 23.89 <b>23.99</b>	114.93 118.55 204.83		0.383 <b>0.358</b> 0.475		
2	$\sigma_n = 0.0$	= 0.0 <b>Inpaint (box)</b>		x) Inpaint (random)			Del	Deblur (Gaussian) D			Deblur (motion)			<b>SR (</b> ×4)			
od	NFEs $\downarrow$	FID↓	LPIPS $\downarrow$	PSNR ↑	FID↓	LPIPS $\downarrow$	PSNR↑	$\mathrm{FID}\downarrow$	LPIPS $\downarrow$	PSNR $\uparrow$	$FID\downarrow$	LPIPS $\downarrow$	PSNR ↑	FID ↓	LPIPS 🗸		
IR IR	20 100	35.72 <b>25.64</b>	0.117 <b>0.107</b>	34.03 <b>36.17</b>	30.81 <b>13.68</b>	0.116 <b>0.066</b>	30.74 <b>31.00</b>	46.64 <b>39.27</b>	0.170 <b>0.152</b>	37.03 37.53	20.11 <b>11.54</b>	0.084 <b>0.064</b>	29.17 29.52	58.02 <b>47.80</b>	0.187 <b>0.174</b>		
1] M [2] [3]	1000 20 >20	43.49 37.05 -	0.145 0.119 -	34.65 31.83 -	33.14 56.60 -	0.105 0.164 -	27.31 28.40 30.52	51.23 67.99 96.16	0.192 0.238 0.350	26.73 - <b>38.39</b>	58.63 - 27.55	0.222	27.64 30.09 <b>30.41</b>	59.06 68.59 96.16	0.209 0.188 0.362		

## **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/yuanzhizhu/DiffPIR
- WeChat yuanzhi-zhu



![](_page_0_Picture_80.jpeg)

![](_page_0_Picture_81.jpeg)

## DIFFUSION ART QR CODE

![](_page_0_Picture_83.jpeg)