Appendix

A. Other HQS-based Sampling Methods

A.1. HQS as one diffusion step

For each of the conditional reverse diffusion step t, we are actually solving the MAP estimation problem on noise level β_t:

$$
\begin{aligned}
&\hat{\mathbf{x}}_t = \arg\min_{\mathbf{x}_t} \frac{1}{2\sigma_n^2}\|\mathbf{y}-\mathcal{H}(\mathbf{x}_t)\|^2 + \lambda\mathcal{P}(\mathbf{z}_t) \\
&s.t. \quad \mathbf{x}_t = \mathbf{z}_t = \sqrt{1-\beta_t}\mathbf{z}_{t-1} + \sqrt{\beta_t}\epsilon
\end{aligned} \tag{16}
$$

With the HQS trick, now we have to solve

$$
\begin{cases}
\hat{\mathbf{x}}_t = \arg\min\limits_{\mathbf{x}_t} \|\mathbf{y}-\mathcal{H}(\mathbf{x}_t)\|^2 + \mu\sigma_n^2\|\mathbf{x}_t-\hat{\mathbf{z}}_t\|^2 & \text{(17a)} \\
\hat{\mathbf{z}}_t = \arg\min\limits_{\mathbf{z}_t} \dfrac{1}{2(\sqrt{\lambda/\mu})^2}\|\mathbf{z}_t-\hat{\mathbf{x}}_t\|^2 + \mathcal{P}(\mathbf{z}_t) & \text{(17b)}
\end{cases}
$$

for each reverse diffusion step. We define $\sigma_t = \sqrt{\lambda/\mu}$ where σ_t is the relative noise level between $\mathbf{x}_t$ and $\mathbf{z}_t$ with $\sigma_t = \sqrt{\frac{\beta_t}{1-\beta_t}}$.

To build the connection between (17b) and a reverse diffusion step (7), we first rewrite (17b) as

$$
\begin{aligned}
\hat{\mathbf{z}}_{t-1} = &\arg\min_{\mathbf{z}_{t-1}} \frac{1}{2(\frac{\beta_t}{1-\beta_t})}\|\sqrt{1-\beta_t}\mathbf{z}_{t-1}+\sqrt{\beta_t}\epsilon-\hat{\mathbf{x}}_t\|^2 \\
&+\mathcal{P}(\sqrt{1-\beta_t}\mathbf{z}_{t-1}+\sqrt{\beta_t}\epsilon).
\end{aligned} \tag{18}
$$

Note that we have $\nabla_{\mathbf{x}}\mathcal{P}(\mathbf{x}) = -\nabla_{\mathbf{x}}\log p(\mathbf{x}) = -\mathbf{s}_\theta(\mathbf{x})$. For any ϵ_0 sampled from $\mathcal{N}(\mathbf{0},\mathbf{I})$, we have

$$
\sqrt{1-\beta_t}\hat{\mathbf{z}}_{t-1}+\sqrt{\beta_t}\epsilon_0 \approx \hat{\mathbf{x}}_t + \frac{\beta_t}{1-\beta_t}\mathbf{s}_\theta(\hat{\mathbf{x}}_t,t) \tag{19}
$$

minimize the RHS of (18) as first-order approximation of the proximal operator, which is also a standard gradient step with step length $\frac{\beta_t}{1-\beta_t}$. Then $\hat{\mathbf{z}}_{t-1}$ can be solved as:

$$
\begin{aligned}
\hat{\mathbf{z}}_{t-1} = &\frac{1}{\sqrt{1-\beta_t}}(\hat{\mathbf{x}}_t+(\beta_t+o(\beta_t))\mathbf{s}_\theta(\hat{\mathbf{x}}_t,t)) \\
&+\sqrt{\beta_t}(1+o(\beta_t))\epsilon_0' \\
\approx &\frac{1}{\sqrt{\alpha_t}}(\hat{\mathbf{x}}_t+\beta_t\mathbf{s}_\theta(\hat{\mathbf{x}}_t,t))+\sqrt{\beta_t}\epsilon_0'
\end{aligned} \tag{20}
$$

where $\epsilon_0' = -\epsilon_0$ is also a sample from $\mathcal{N}(\mathbf{0},\mathbf{I})$ and (20) is the same as reverse process of DDPM (7).

A.2. DPS as a Special Case

For (17a), we can write similarly to Section 3.2:

$$
\hat{\mathbf{x}}_t \approx \hat{\mathbf{z}}_t - \frac{\sigma_t^2}{2\lambda\sigma_n^2}\nabla_{\mathbf{z}_t}\|\mathbf{y}-\mathcal{H}(\mathbf{z}_t)\|^2 \tag{21}
$$

With the Theorem 1 from DPS [8]

$$
\nabla_{\mathbf{x}_t}\log p_t(\mathbf{y}|\mathbf{x}_t) \simeq \nabla_{\mathbf{x}_t}\log p(\mathbf{y}|\hat{\mathbf{x}}_0) \tag{22}
$$

(21) turned into:

$$
\hat{\mathbf{x}}_t \approx \hat{\mathbf{z}}_t - \frac{\sigma_t^2}{2\lambda\sigma_n^2}\nabla_{\mathbf{z}_t}\|\mathbf{y}-\mathcal{H}(\mathbf{z}_0)\|^2 \tag{23}
$$

By setting $\zeta_t = \frac{\sigma_t^2}{2\lambda\sigma_n^2} = \frac{1}{2\rho_t}$, we are now able to reproduce the sampling strategy in DPS.

Moreover, we can use the conclusion from [53] that

$$
\nabla_{\mathbf{x}_t}\log p_t(\mathbf{x}_t \mid \mathbf{y}) \approx \nabla_{\mathbf{x}_t}\log p_t(\mathbf{x}_t) + \nabla_{\mathbf{x}_t}\log p_t(\mathbf{y}_t \mid \mathbf{x}_t),
$$

where $\mathbf{y}_t = \sqrt{\bar{\alpha}_t}\mathbf{y} + \sqrt{1-\bar{\alpha}_t}\epsilon$ is the measurement $\mathbf{y}$ at the given noise level and $\mathbf{y}_t$ is assumed to be the measurement from $\mathbf{x}_t$.

As a result, we can write a variant of (23) as

$$
\hat{\mathbf{x}}_t \approx \hat{\mathbf{z}}_t - \frac{\sigma_t^2}{2\lambda\sigma_n^2}\nabla_{\mathbf{z}_t}\|\mathbf{y}_t-\mathcal{H}(\mathbf{z}_t)\|^2 \tag{24}
$$

To distinguish them, we call the original DPS as DPSy_0 and the algorithm with (24) as DPSy_t. The algorithm of DPSy_t is:

Algorithm 2 Extended Sampling I: DPSy_t

```
Require: $\mathbf{s}_\theta, T, \mathbf{y}, \sigma_n, \{\sigma_t\}_{t=1}^T, \lambda$
Initialize $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0},\mathbf{I})$
for $t = T$ to $1$ do
    $\epsilon_t \sim \mathcal{N}(\mathbf{0},\mathbf{I})$
    $\mathbf{z}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t,t)\right) + \sqrt{\beta_t}\epsilon_t$ // one step reverse diffusion sampling
    $\mathbf{x}_{t-1} = \mathbf{z}_{t-1} - \frac{\sigma_t^2}{2\lambda\sigma_n^2}\nabla_{\mathbf{z}_{t-1}}\|\mathbf{y}_{t-1}-\mathcal{H}(\mathbf{z}_{t-1})\|^2$ // Solving data proximal subproblem
end for
return $\mathbf{x}_0$
```

B. Experimental Details

B.1. Hyperparameters Values

We list the hyperparametrs values for different tasks and datasets in 3.

NFE=20	$\sigma_y = 0.05$				$\sigma_y = 0.0$	
Dataset	**FFHQ 256x256**		**ImgaeNet 256x256**		**FFHQ 256x256**	
Hyperparameters	λ	ζ	λ	ζ	λ	ζ
Inpaint (box)	-	-	-	-	6.0	1.0
Inpaint (random)	-	-	-	-	3.0	1.0
Deblur (gauss)	8.0	0.5	12.0	0.9	15.0	0.5
Deblur (motion)	7.0	0.8	7.0	1.0	25.0	1.0
SR ($\times 4$)	8.0	0.4	10.0	0.5	9.0	0.2

NFE=100	$\sigma_y = 0.05$				$\sigma_y = 0.0$	
Dataset	FFHQ 256x256		ImgaeNet 256x256		FFHQ 256x256	
Hyperparameters	λ	ζ	λ	ζ	λ	ζ
Inpaint (box)	-	-	-	-	6.0	0.5
Inpaint (random)	-	-	-	-	7.0	1.0
Deblur (gauss)	7.0	0.3	8.0	0.3	12.0	0.4
Deblur (motion)	7.0	0.4	8.0	0.7	7.0	0.9
SR ($\times 4$)	8.0	0.2	9.0	0.5	6.0	0.3

Table 3. Hyperparameters for different tasks.

B.2. Closed-form Solutions

In this section, we will introduce the specific degradation models and fast solutions of (12b) for image restoration tasks including SR, deblurring and inpainting.

Image Inpainting. In this work, we only consider the noiseless inpainting. The degradation model of masked image for inpainting can be expressed as

$$\mathbf{y} = \mathbf{M} \odot \mathbf{x}, \tag{25}$$

where $\mathbf{M}$ is any user-defined mask and is a matrix with boolean elements, and $\odot$ denotes element-wise multiplication. The image inpainting task is to recover the missing pixels from the known pixels as $\mathbf{y}$. The closed-from solution of (12b) is given by [57]

$$\mathbf{x}_0 = \frac{\mathbf{M} \odot \mathbf{y} + \rho_t \mathbf{z}_0}{\mathbf{M} + \rho_t}, \tag{26}$$

and the division here is also element-wise.

Image Deblurring. The linear degradation model for image deblurring with Gaussian noise is generally expressed as

$$\mathbf{y} = \mathbf{x} \otimes \mathbf{k} + \mathbf{n}, \tag{27}$$

where $\otimes$ is two-dimensional convolution operator applied on all image channels. By assuming $\otimes$ is also a circular convolution operator, the analytical solution of (12b) is given by [57]

$$\mathbf{x}_0 = \mathcal{F}^{-1}\left(\frac{\overline{\mathcal{F}(\mathbf{k})}\mathcal{F}(\mathbf{y}) + \rho_t \mathcal{F}(\mathbf{z}_0)}{\overline{\mathcal{F}(\mathbf{k})}\mathcal{F}(\mathbf{k}) + \rho_t}\right), \tag{28}$$

where the $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ denote Fast Fourier Transform (FFT) and its inverse.

Single Image Super-Resolution (SISR). In this work, we consider bicubic SR, which has the following degradation model

$$\mathbf{y} = \mathbf{x} \downarrow_{sf}^{bicubic} + \mathbf{n}, \tag{29}$$

where $\downarrow_{sf}^{bicubic}$ denotes bicubic downsamling with downscaling factor sf.

We can then solve (12b) with the following iterative back-projection (IBP) solution

$$\mathbf{x}_0 = \mathbf{z}_0 - \gamma(\mathbf{y} - \mathbf{z}_0 \downarrow_{sf}^{bicubic}) \uparrow_{sf}^{bicubic}, \tag{30}$$

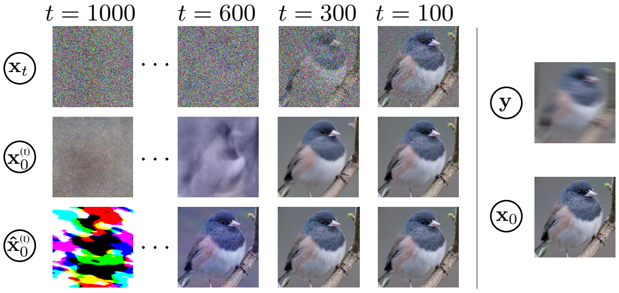

Figure 9. Reverse diffusion process in DiffPIR

where $\uparrow_{sf}^{bicubic}$ denotes bicubic interpolation with upscaling factor sf, γ is the step size. Through experiment, we found that it's better to use $\gamma_t = \frac{\gamma}{1+\rho_t}$ which will decrease with time. To get the solution accurately, we IBP for more than one iteration for each timestep t.

For bicubic SR, we can also solve (12b) in closed-form with an approximated bicubic kernels $\mathbf{k}$ [57]

$$\mathbf{x}_0 = \mathcal{F}^{-1}\left(\frac{1}{\rho_t}\left(\mathbf{d} - \overline{\mathcal{F}(\mathbf{k})} \odot_s \frac{(\mathcal{F}(\mathbf{k})\mathbf{d}) \Downarrow_s}{(\overline{\mathcal{F}(\mathbf{k})}\mathcal{F}(\mathbf{k})) \Downarrow_s + \rho_t}\right)\right), \tag{31}$$

where $\mathbf{d} = \overline{\mathcal{F}(\mathbf{k})}\mathcal{F}(\mathbf{y} \uparrow_{sf}) + \rho_t \mathcal{F}(\mathbf{z}_0)$ and $\uparrow_{sf}$ denotes the standard s-fold upsampler, and where $\odot_s$ denotes distinct block processing operator with element-wise multiplication, $\Downarrow_s$ denotes distinct block downsampler, i.e., averaging the $s \times s$ distinct blocks. In general, the closed-form solution (31) should outperform iterative solutions (30) in quantitative metrics, since the former contains fewer hyperparameters.

C. Additional Ablation Study

In this section, we illustrate the reverse diffusion process by showing the intermediate results in Figure 9. We observed that in the beginning, the analytical solution offers no help and motivate us to skip this phase. As mentioned in Section 4.4, we found by experiment t_{start} the end timestep for this phase.

D. Additional Visual Results

In this section, we provide additional visual examples for FFHQ and ImageNet datasets to show the ability of our method. In Figure 10 we demonstrate that DPSy_t and DPSy_0 both work well on IR tasks like deblurring and SR. In Figure 11, we show the diversity of SR reconstructions with diffusion model as generative prior. In Figure 12 and 13, we show that our proposed DiffPIR is capable to handle various blur kernels (both motion and Gaussian) and masks, respectively.

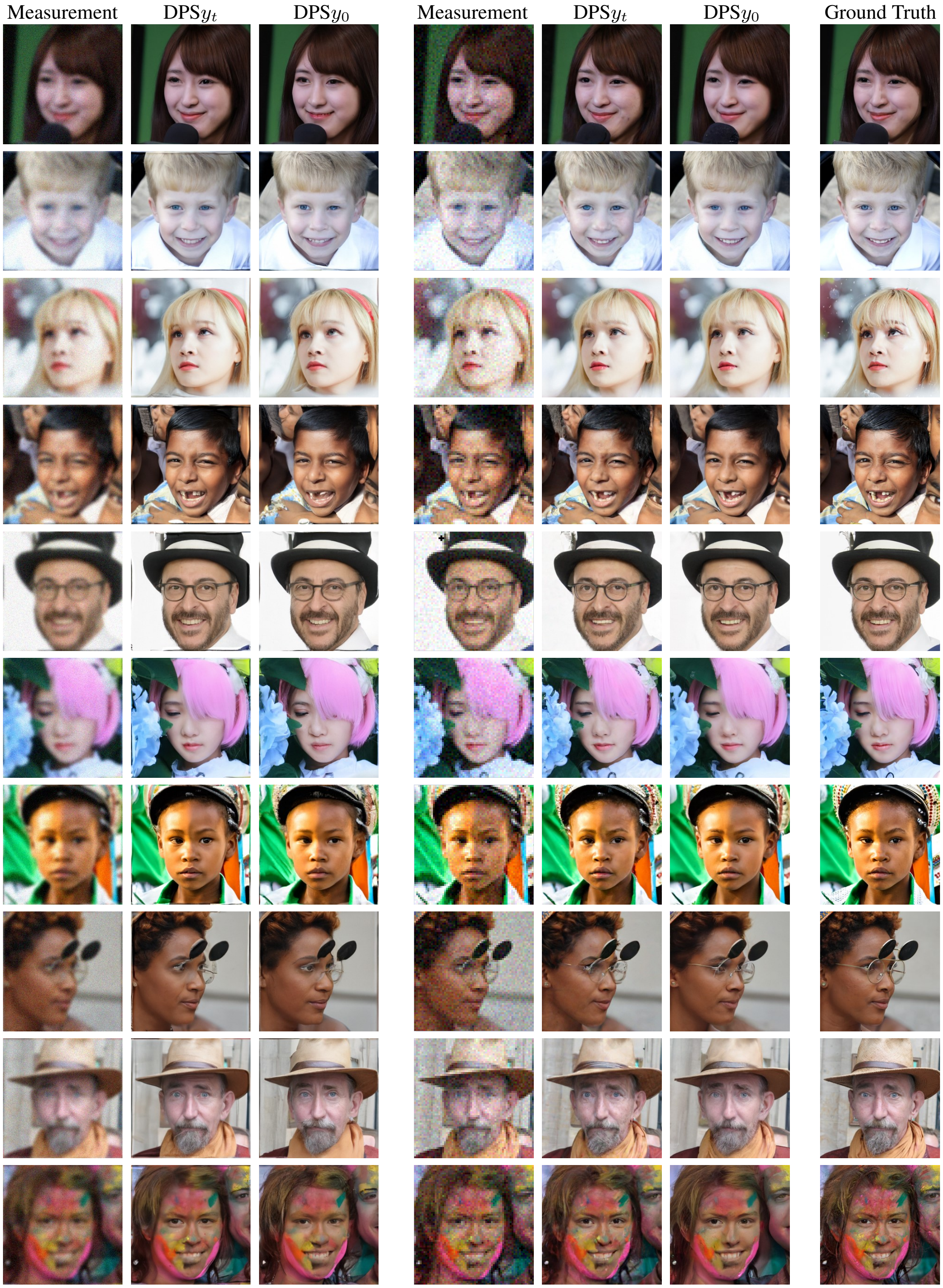

Figure 10. Qualitative results of DPSy_t and DPSy_0 (both 1000 NFEs) for Gaussian deblurring (left) and 4× SR (right) with $\sigma_n = 0.05$

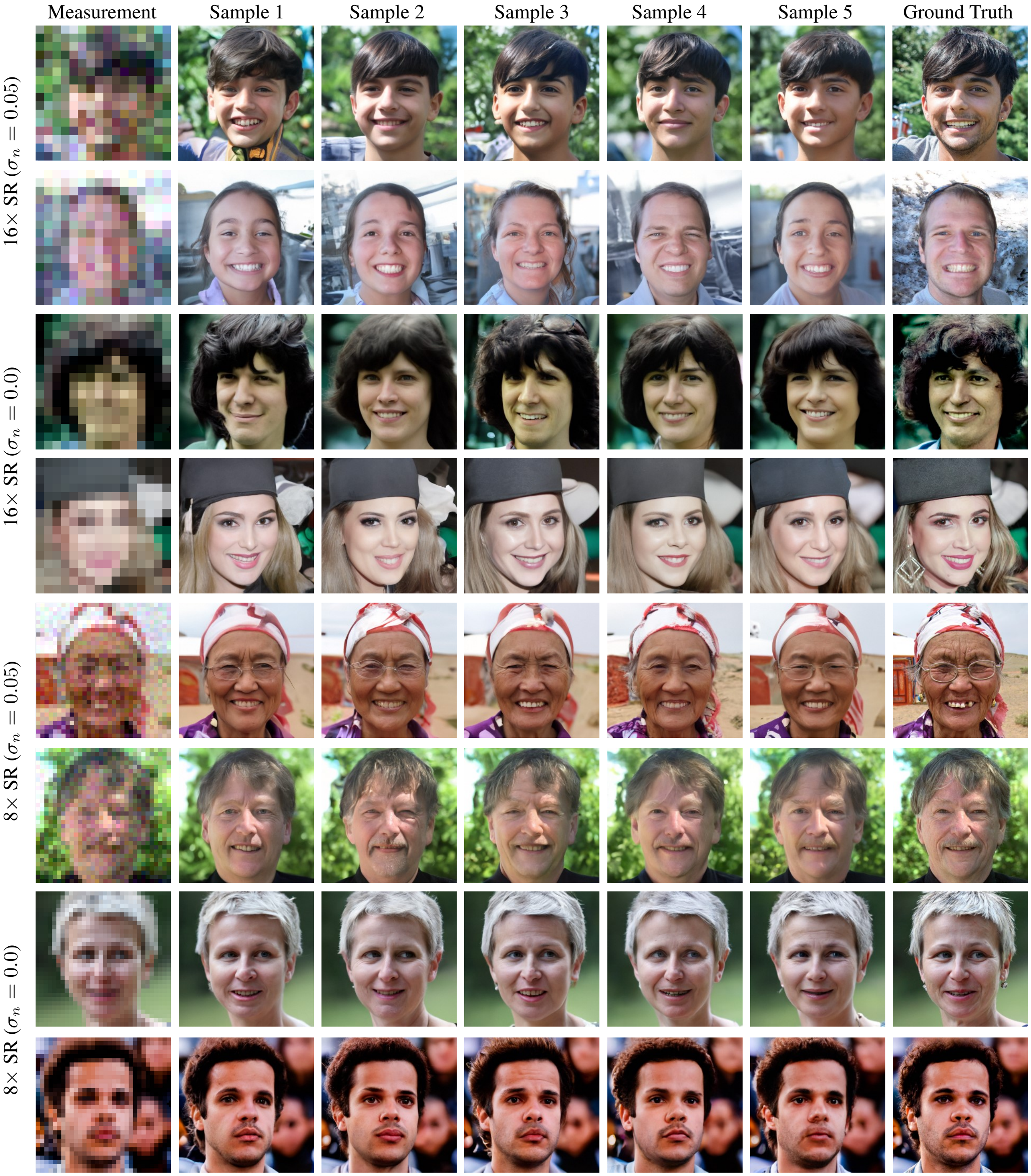

Figure 11. Qualitative results of DiffPIR (100 NFEs) for 8× and 16× SR with $\sigma_n = 0.0$ and $\sigma_n = 0.05$.

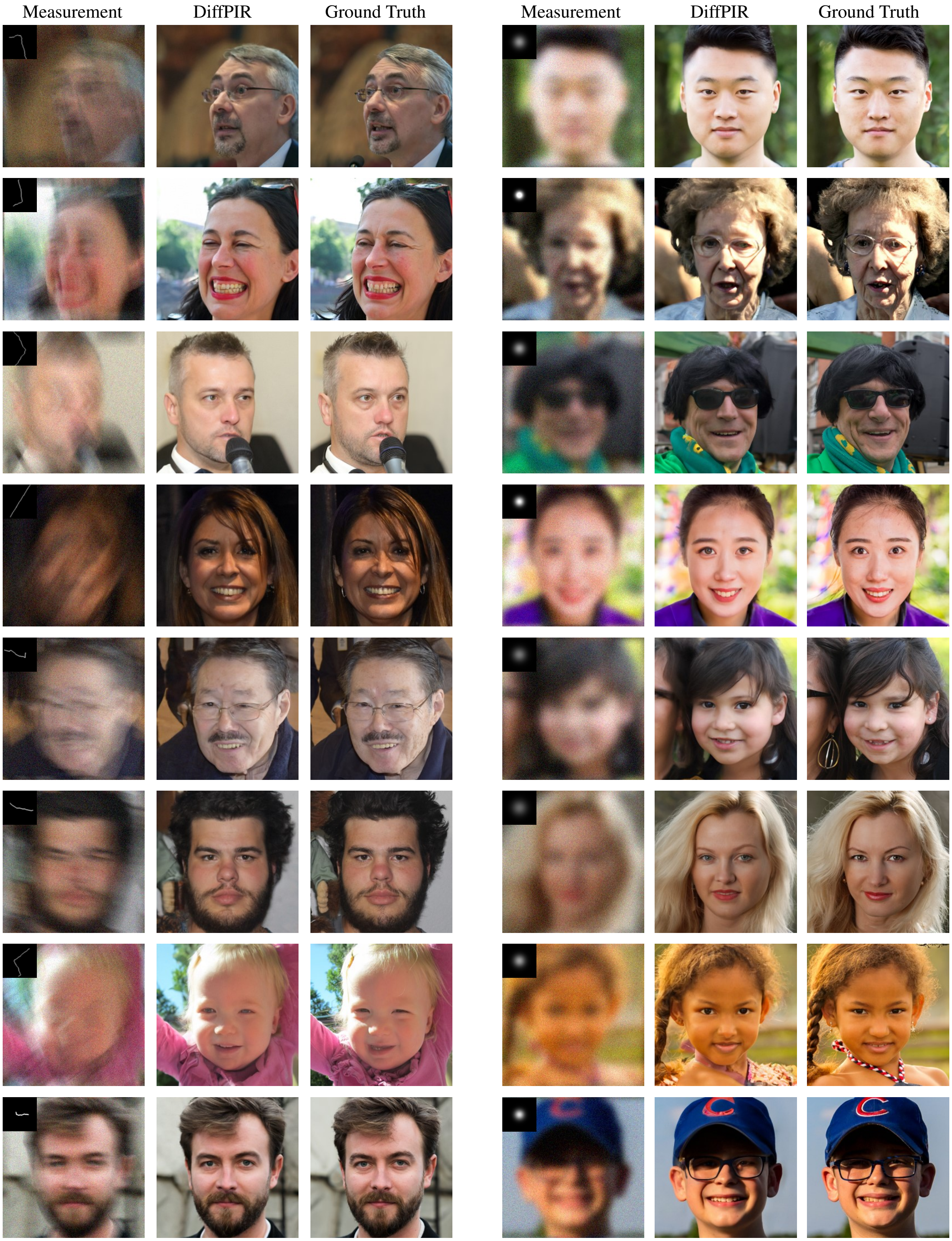

Figure 12. Qualitative results of DiffPIR (100 NFEs) for motion deblurring (left) and Gaussian deblurring (right) with $\sigma_n = 0.05$

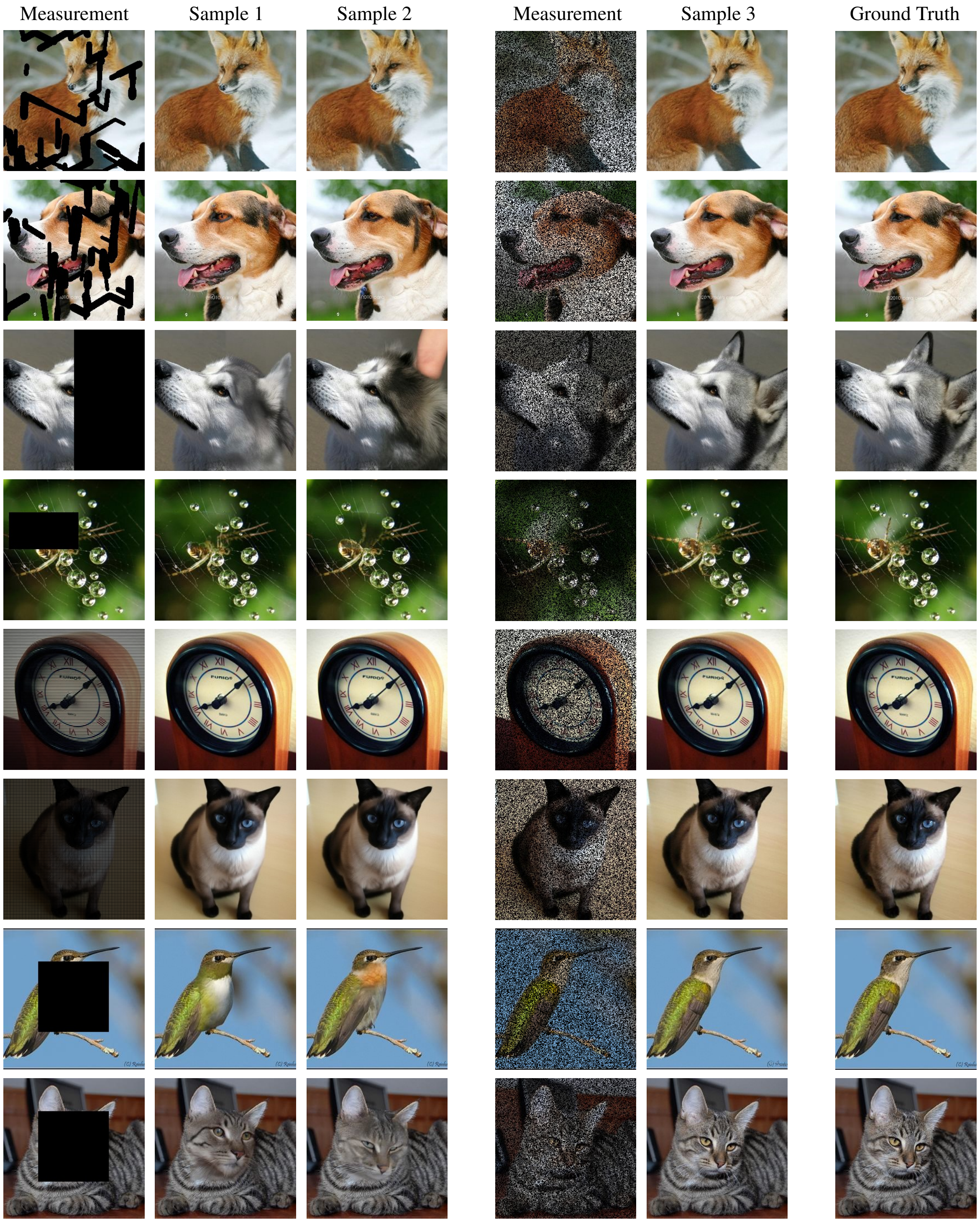

Figure 13. Qualitative results of DiffPIR (100 NFEs) for inpainting with different masks ($\sigma_n = 0.0$)