A Score-based Diffusion Models

Here, we review the continues form of diffusion model introduced for completeness. The forward process of diffusion can be formulated by an Itô SDE:

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}, \tag{18}$$

where $\mathbf{f}(\cdot, t) : \mathbb{R}^d \mapsto \mathbb{R}^d$ is a drift coefficient function, $g(t) \in \mathbb{R}$ is a scalar function known as the diffusion coefficient, and $\mathbf{w} \in \mathbb{R}^d$ is the standard Wiener process. The forward process of DDPM can be viewed as variance-preserving (VP) SDE [51] as the total variance is preserved.

Correspondingly, the reversed generative (i.e., denoising) process is given by the reverse-time SDE:

$$d\mathbf{x} = [\mathbf{f}(\mathbf{x}, t) - g^2(t)\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)]dt + g(t)d\bar{\mathbf{w}}, \tag{19}$$

where $d\bar{\mathbf{w}}$ denotes the standard Wiener process running backward in time and $p_t(\mathbf{x}_t)$ denotes the marginal probability density w.r.t. $\mathbf{x}$ at time t. In practice, a time-dependent network $\mathbf{s}_\theta(\mathbf{x}_t, t)$ parameterized by θ is trained to approximate the (Stein) score function $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)$ with score-matching method [54]:

$$\min_\theta \mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_t}\left[\|\mathbf{s}_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)\|_2^2\right], \tag{20}$$

where t is uniformly sampled from $[0, T]$, $\mathbf{x}_0 \sim q_{data}(\mathbf{x})$ and $\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)$. Once we have aceess to the well-trained $\mathbf{s}_\theta(\mathbf{x}_t, t)$, an clean sample can be derived by simulating the generative reverse-time SDE (19) using numerical solvers (e.g. Euler-Maruyama).

B Additional Details

B.1 Derivation for $\mathbf{x}^*_{t-1}$

We show the detailed derivation and explanation for the $\mathbf{x}^*_{t-1}$ in Equation (9). Our motivation was to give the stochastic sampling process a supervision at inference, trying to make the sampled $\mathbf{x}_{t-1}$ in the vicinity of the theoretically derived solution. Recall the inference distributions defined in [47]:

$$\begin{aligned} & q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \\ &= \mathcal{N}(\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2} \cdot \frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{\sqrt{1 - \bar{\alpha}_t}}, \sigma_t^2\mathbf{I}). \end{aligned} \tag{21}$$

By setting $\sigma_t = 0$, we get our deterministic denoised estimate $\mathbf{x}^*_{t-1}$

$$\begin{aligned} \mathbf{x}^*_{t-1} &= \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_{t-1}} \cdot \frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{\sqrt{1 - \bar{\alpha}_t}} \\ &= \frac{\sqrt{1 - \bar{\alpha}_{t-1}}}{\sqrt{1 - \bar{\alpha}_t}}\mathbf{x}_t + (\sqrt{\bar{\alpha}_{t-1}} - \frac{\sqrt{\bar{\alpha}_t} \cdot \sqrt{1 - \bar{\alpha}_{t-1}}}{\sqrt{1 - \bar{\alpha}_t}})\mathbf{x}_0. \end{aligned} \tag{22}$$

Then we have the approximation for $\mathcal{A}\mathbf{x}^*_{t-1}$:

$$\mathcal{A}\mathbf{x}^*_{t-1} = \frac{\sqrt{1 - \bar{\alpha}_{t-1}}}{\sqrt{1 - \bar{\alpha}_t}}\mathcal{A}\mathbf{x}_t + (\sqrt{\bar{\alpha}_{t-1}} - \frac{\sqrt{\bar{\alpha}_t} \cdot \sqrt{1 - \bar{\alpha}_{t-1}}}{\sqrt{1 - \bar{\alpha}_t}})\mathcal{A}\mathbf{x}_0. \tag{23}$$

By applying $\mathbf{y} = \mathcal{A}\mathbf{x}_0 + \mathbf{n}$,

$$\begin{aligned} & \mathcal{A}\mathbf{x}^*_{t-1} \\ &= \frac{\sqrt{1 - \bar{\alpha}_{t-1}}}{\sqrt{1 - \bar{\alpha}_t}}\mathcal{A}\mathbf{x}_t + (\sqrt{\bar{\alpha}_{t-1}} - \frac{\sqrt{\bar{\alpha}_t} \cdot \sqrt{1 - \bar{\alpha}_{t-1}}}{\sqrt{1 - \bar{\alpha}_t}})(\mathbf{y} - \mathbf{n}) \\ &\approx \frac{\sqrt{1 - \bar{\alpha}_{t-1}}}{\sqrt{1 - \bar{\alpha}_t}}\mathcal{A}\mathbf{x}_t + (\sqrt{\bar{\alpha}_{t-1}} - \frac{\sqrt{\bar{\alpha}_t} \cdot \sqrt{1 - \bar{\alpha}_{t-1}}}{\sqrt{1 - \bar{\alpha}_t}})\mathbf{y} \end{aligned} \tag{24}$$

The approximation of last step holds when σ_y is assumed in a small range as $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \sigma_\mathbf{y}\mathbf{I})$.

Proposition B.1. *Assume that $\|\mathbf{y}\|_2 \leq Y, \|\mathbf{x}_0\|_2 \leq X, \|\mathbf{x}'_{t-1}\|_2 \leq X, \|\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y})\|_2 \leq E$ are bounded and continuous, there exists a upper bound for $\|\mathcal{A}\left(\mathbf{x}^i_{t-1} - \mathbf{x}^*_{t-1}\right)\|_2^2$ which is:*

$$\|\mathcal{A}\left(\mathbf{x}^i_{t-1} - \mathbf{x}^*_{t-1}\right)\|_2^2 \leq (\mathcal{A}E - \mathcal{A}X + \mathcal{A}\sigma_t)^2. \tag{25}$$

It is proven that some linear (i.e. super-resolution and inpainting) operators $\mathcal{A}$ can be modeled as matrices [8] which are bounded and linear. And for σ_t, it is usually set to a small value which is also bounded. Thus, we can conclude that the above upper bound holds.

Proof.

$$\begin{aligned} & \|\mathcal{A}\left(\mathbf{x}^i_{t-1} - \mathbf{x}^*_{t-1}\right)\|_2^2 \\ &= \|\mathcal{A}\mathbf{x}^i_{t-1} - \mathcal{A}\mathbf{x}^*_{t-1}\|_2^2 \end{aligned} \tag{26}$$

$$\begin{aligned} & \overset{(a)}{\approx} \|\mathcal{A}\left(\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y}) + \sigma_t\mathbf{z}^i_t\right) - \\ & \left(\sqrt{\bar{\alpha}_{t-1}}\mathbf{y} + \sqrt{1 - \bar{\alpha}_{t-1}} \cdot \frac{\mathcal{A}\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{y}}{\sqrt{1 - \bar{\alpha}_t}}\right)\|_2^2, \end{aligned} \tag{27}$$

where $\mathbf{z}^i_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and the $\overset{(a)}{\approx}$ is from Eq. (11). Here, we apply trigonometric inequalities to the above equation, we have:

$$\begin{aligned} & \|\mathcal{A}\left(\mathbf{x}^i_{t-1} - \mathbf{x}^*_{t-1}\right)\|_2^2 \\ & \overset{(a)}{\approx} \|\mathcal{A}\left(\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y}) + \sigma_t\mathbf{z}^i_t\right) - \\ & \left(\sqrt{\bar{\alpha}_{t-1}}\mathbf{y} + \sqrt{1 - \bar{\alpha}_{t-1}} \cdot \frac{\mathcal{A}\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{y}}{\sqrt{1 - \bar{\alpha}_t}}\right)\|_2^2 \end{aligned} \tag{28}$$

$$\begin{aligned} & \approx \|\left(\mathcal{A}\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y}) - \sqrt{\bar{\alpha}_{t-1}}\mathbf{y}\right) + \\ & \left(\sigma_t\mathcal{A}\mathbf{z}^i_t - \sqrt{1 - \bar{\alpha}_{t-1}} \cdot \frac{\mathcal{A}\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{y}}{\sqrt{1 - \bar{\alpha}_t}}\right)\|_2^2 \end{aligned} \tag{29}$$

$$\begin{aligned} & \approx \left(\|\mathcal{A}\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y}) - \sqrt{\bar{\alpha}_{t-1}}\mathbf{y}\|_2 + \right. \\ & \left. \|\sigma_t\mathcal{A}\mathbf{z}^i_t - \sqrt{1 - \bar{\alpha}_{t-1}} \cdot \frac{\mathcal{A}\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{y}}{\sqrt{1 - \bar{\alpha}_t}}\|_2\right)^2. \end{aligned} \tag{30}$$

Since $\|\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y})\|_2 \leq E$, $\|\bar{\alpha}_t\|_2 \leq A$, $\|\bar{\alpha}_t\|_2 \leq A$, $\|\bar{\alpha}_{t-1}\|_2 \leq A$,, and $\|\mathbf{y}\|_2 \leq Y$, we have the upper bound for $\|\mathcal{A}\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y}) - \sqrt{\bar{\alpha}_{t-1}}\mathbf{y}\|_2$, and thus $\|\mathcal{A}\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y}) - \sqrt{\bar{\alpha}_{t-1}}\mathbf{y}\|_2 \leq \mathcal{A}E + \sqrt{A}Y$. Similarly, we have the upper bound for $\|\sqrt{1 - \bar{\alpha}_{t-1}} \cdot \frac{\mathcal{A}\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{y}}{\sqrt{1-\bar{\alpha}_t}}\|_2$, which is $\|\sqrt{1 - \bar{\alpha}_{t-1}} \cdot \frac{\mathcal{A}\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{y}}{\sqrt{1-\bar{\alpha}_t}}\|_2 = \|\frac{\sqrt{1-\bar{\alpha}_{t-1}}\mathcal{A}\mathbf{x}_t}{\sqrt{1-\bar{\alpha}_t}} - \frac{\sqrt{\bar{\alpha}_t(1-\bar{\alpha}_{t-1})}y}{\sqrt{1-\bar{\alpha}_t}}\| \leq \mathcal{A}X + \sqrt{A}Y$. Therefore, we have:

$$\begin{aligned} & \|\mathcal{A}\left(\mathbf{x}^i_{t-1} - \mathbf{x}^*_{t-1}\right)\|_2^2 \\ & \overset{(a)}{\approx} \|\mathcal{A}\left(\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y}) + \sigma_t\mathbf{z}^i_t\right) - \\ & \left(\sqrt{\bar{\alpha}_{t-1}}\mathbf{y} + \sqrt{1 - \bar{\alpha}_{t-1}} \cdot \frac{\mathcal{A}\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{y}}{\sqrt{1 - \bar{\alpha}_t}}\right)\|_2^2 \end{aligned} \tag{31}$$

$$\leq \left(\mathcal{A}E + \mathcal{A}X + 2\sqrt{A}Y + \mathcal{A}\sigma_t\right)^2. \tag{32}$$

Chung et al. [8] has shown that some linear operations $\mathcal{A}$ (such as super-resolution and inpainting) can be represented as matrices that are linear and bounded. Moreover, σ_t is typically chosen to be a small and finite number. Therefore, we can affirm that the upper bound in the previous equation is valid. □

Proposition B.2. *For the random variable* $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ *and its objective function:*

$$f(\mathbf{z}) = \|\mathcal{A}(\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y}) + \sigma_t \mathbf{z} - C_1 \mathbf{x}_t) - C_2 \mathbf{y}\|_2^2, \tag{33}$$

where $C_1 = \sqrt{1-\bar{\alpha}_{t-1}}/\sqrt{1-\bar{\alpha}_t}$ *and* $C_2 = \sqrt{\bar{\alpha}_{t-1}} - \sqrt{\bar{\alpha}_t}\sqrt{1-\bar{\alpha}_{t-1}}/\sqrt{1-\bar{\alpha}_t}$*. Thus, with M trials, each consisting of N samples, we have the variance for* $f(\mathbf{z})$*, which is* $Var(f(\mathbf{z}))$*. We have*

$$Var_{DPS}(f(\mathbf{z})) > Var_{MC}(f(\mathbf{z})) > Var_{Ours}(f(\mathbf{z})), \tag{34}$$

here, $Var_{DPS}(f(\mathbf{z}))$ *is the variance of DPS [8],* $Var_{MC}(f(\mathbf{z}))$ *is the variance of Monte Carlo sampling, and* $Var_{Ours}(f(\mathbf{z}))$ *is the variance of our proximal sampling method.*

Proof. For each trial, we only take one sample $\mathbf{z}_1$ from the distribution, and compute $f(\mathbf{z}_1)$. Thus, for DPS, the estimation of a single sample is $\hat{\mu}_{\text{single}} = f(\mathbf{z}_1)$, and the variance of the estimation is $\text{Var}(f(\mathbf{z}))$.

For Monte Carlo sampling, we draw N independently and identically distributed samples $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N$, and compute the function $f(\mathbf{z}_i)$ for each sample. Finally, we can get the estimation of a single trial as:

$$\hat{\mu}_{\text{MC}} = \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{z}_i). \tag{35}$$

By the central limit theorem, when N is large enough, the variance of $\hat{\mu}_{\text{MC}}$ is $\frac{\text{Var}(f(\mathbf{z}))}{N}$.

For our proximal sampling method, similar to Monte Carlo sampling, we also draw N independently and identically distributed samples $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N$ for each trial, and compute the function $f(\mathbf{z}_i)$ for each sample. However, we only get the estimation from the sample with the lowest objective function value:

$$\hat{\mu}_{\text{Ours}} = \arg\min_{\mathbf{z}} f(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N). \tag{36}$$

In contrast to Monte Carlo sampling, which averages all function values, our proximal sampling selects only the samples corresponding to the smallest function values for each trial. Thus, the set of function value selected by our method is a subset of those sampled by Monte Carlo sampling. Obviously, for multiple trials, the variance of our method is smaller than Monte Carlo sampling, and also smaller than random sampling as in DPS. Therefore, we have:

$$\text{Var}_{\text{DPS}}(f(\mathbf{z})) > \text{Var}_{\text{MC}}(f(\mathbf{z})) = \frac{\text{Var}_{\text{DPS}}(f(\mathbf{z}))}{N} > \text{Var}_{\text{Ours}}(f(\mathbf{z})). \tag{37}$$

□

C Additional Experimental Results

C.1 Less Error Accumulation

As discussed in Section 5.3, sampling in proximity to the measurement yields less error accumulation. This is achieved by treating the injected noise as an adaptive correction. To verify this, Figure 9 reports the Frobenius norm between true values $\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0$ and the predicted values $\boldsymbol{\mu}_\theta(\mathbf{x}_t, t, \mathbf{y})$.

The results validate that our proximal sampling exhibits superior predictive accuracy and less error accumulation compared to random sampling, consequently enhancing the precision in predicting subsequent samples.

D Experimental Details

D.1 Comparison Methods

Since Score-SDE, DDRM, MCG, DPS, and DiffPIR are all pixel-based diffusion models, we used the same pre-trained checkpoint for fair comparison.

PnP-ADMM: For PnP-ADMM we take the pre-trained model from DnCNN [59] repository, and set $\tau = 0.2$ and number of iterations to 10 for all inverse problem.

Score-SDE: For Score-SDE, data consistency projection is conducted after unconditional diffusion denoising at each step. We adopt the same projection settings as suggested in [9].

MCG, DPS: The experimental results are derived from the source code implementation provided by [8] with the default parameter setting as suggested in the paper, i.e. $\zeta_t = 1/\|\mathbf{y} - \mathcal{A}(\hat{\mathbf{x}}_{0|t})\|$ for all inverse problem on FFHQ dataset, $\zeta_t = 1/\|\mathbf{y} - \mathcal{A}(\hat{\mathbf{x}}_{0|t})\|$ for ImageNet SR and inpainting, $\zeta_t = 0.4/\|\mathbf{y} - \mathcal{A}(\hat{\mathbf{x}}_{0|t})\|$ for ImageNet Gaussian deblur, and $\zeta_t = 0.6/\|\mathbf{y} - \mathcal{A}(\hat{\mathbf{x}}_{0|t})\|$ for ImageNet motion deblur. The difference between these two methods is that MCG additionally applied data consistency steps as Euclidean projections onto the measurement set.

DDRM: We apply 20 NFEs DDIM [47] sampling with $\eta = 0.85, \eta_B = 1.0$ for all experiment as suggested in the paper.

LGD-MC: We follow the implementation of the algorithm in [49] to use a Monte Carlo estimate of the gradient correction to amend the denoising process. The number of Monte Carlo samples is set to 20.

DiffPIR: We use the original code and pre-trained models provided by [61]. We set the hyper-parameters consistent with the noiseless situation in the paper, i.e., SR: $\zeta = 0.3, \lambda = 6.0$ / motion deblur: $\zeta = 0.9, \lambda = 7.0$ / Gaussian deblur: $\zeta = 0.4, \lambda = 12.0$/ inpainting: $\zeta = 1.0/\lambda = 7.0$. Besides, we designate the sub_1_analytic as False in the motion deblurring task, since it directly leverages the pseudo-inverse of the fast Fourier transform, resulting in an unfair boost in performance [34].

D.2 Parameter Setting

Here, we list the hyper-parameter values for different tasks and datasets in Table 5.

Table 5: Hyper-parameter λ_t for each problem setting.

Dataset	FFHQ 256x256	ImgaeNet 256x256
Inpaint	1.0	1.0
Deblur (Gaussian)	1.0	0.5
Deblur (motion)	1.0	0.3
SR (×4)	1.0	1.0

D.3 Parameter Size and Speed

The results for computational time and parameter sizes on the FFHQ model are presented in Table 6. The parameter sizes of the diffusion-based methods remain consistent as the same model was used.

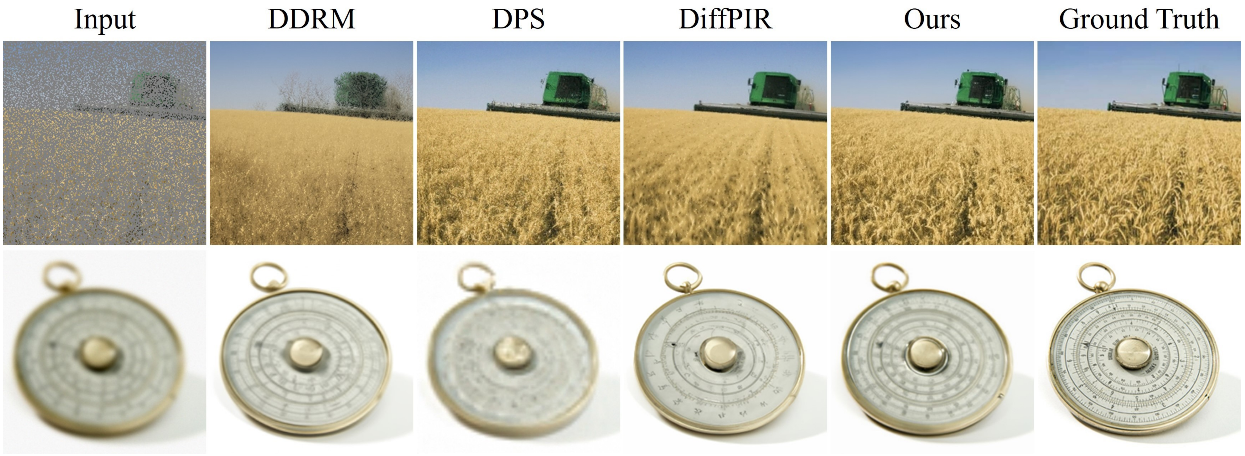

Figure 8: More visual comparison of inpainting and deblurring on ImageNet.

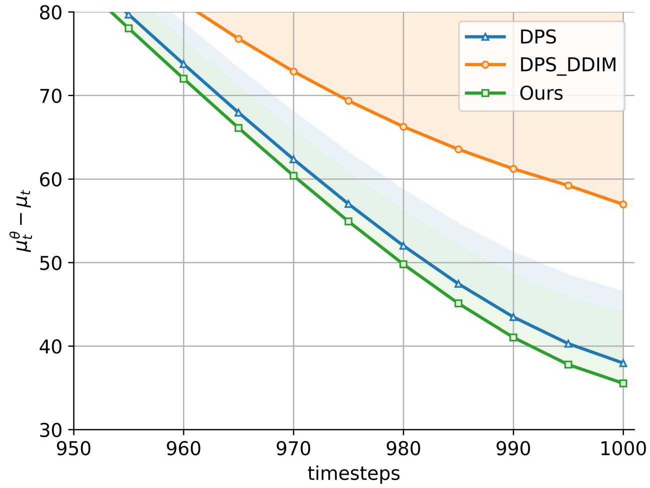

Figure 9: We report the average value of $\|\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0 - \mu_\theta(\mathbf{x}_t, t, \mathbf{y})\|_F$ on SR (×4) task. The results show that our method achieves better predictive accuracy and reduces error accumulation.

Table 6: Computational time and parameter size comparison

Method	PnP-ADMM	Score-SDE	MCG	DDRM	DPS	DiffPIR	Ours
Parameter (M)	0.56	93.56	93.56	93.56	93.56	93.56	93.56
Speed (s)	1.71	28.78	56.41	4.79	56.36	3.40	57.25

D.4 Source Code

Our implementation is now available at https://github.com/74587887/DPPS_code.

E Limitations and Future Work

Our approach notably enhances perceptual metrics, yet it demonstrates a less substantial improvement in distortion metrics and, in certain tasks, experiences a slight decline. While this observation aligns with the perception-distortion trade-off phenomena as described in the literature [3], we acknowledge it as a noteworthy issue that warrants further investigation in our subsequent studies. In addition, subsequent work of this paper aims to extend the application to diverse domains, including but not limited to medical image reconstruction.

F More Visual Results

In this section, we provide supplementary visual results to show the effectiveness of our proposed method. Figures 10 and 11 indicate that our method produces images with better details and quality as n increases. Figure 12 to Figure 15 show the robustness of our method across different random seeds, in line with the claims made in the paper. Figure 8 provides one more visual comparison for inpainting and Gaussian deblurring tasks.

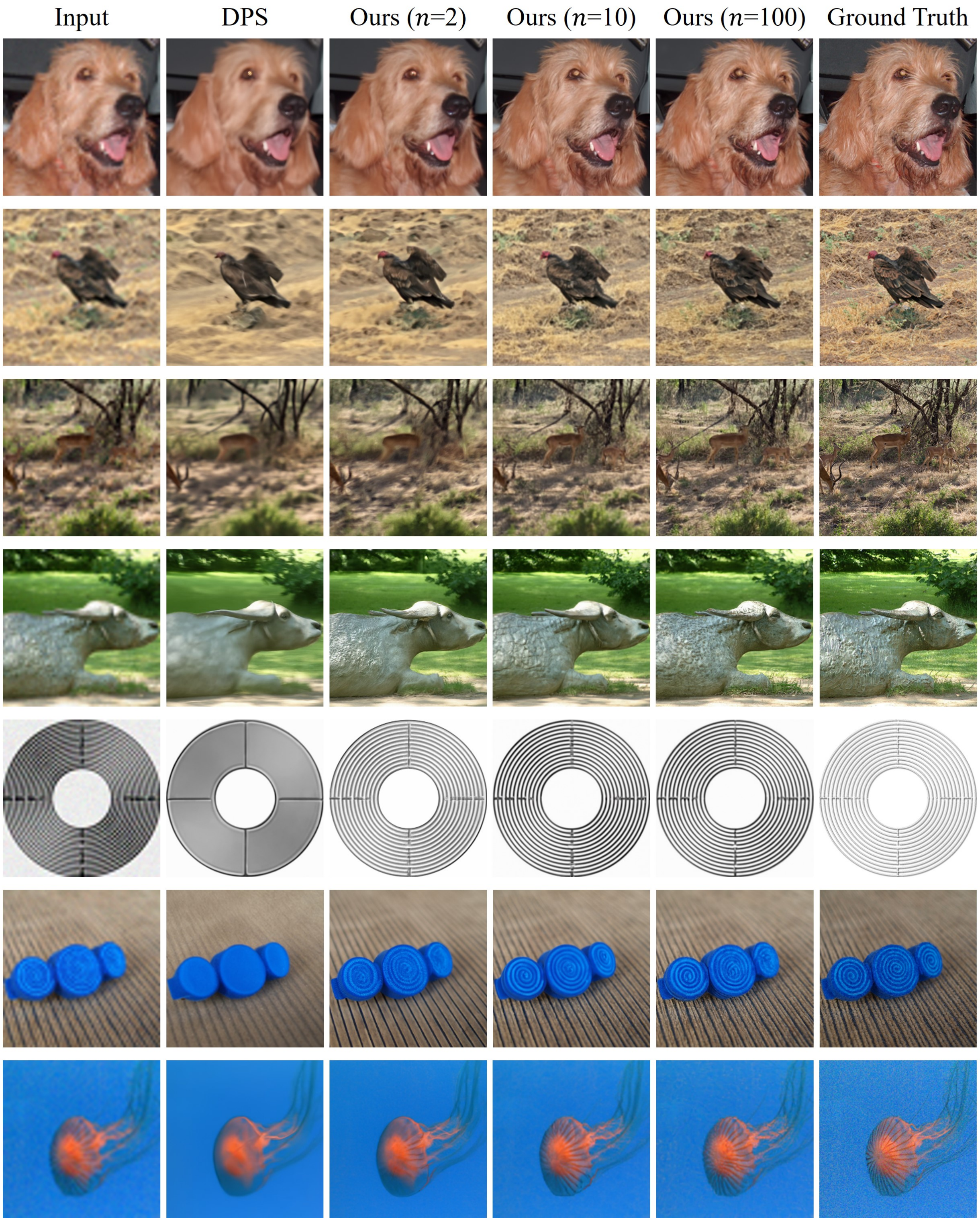

Figure 10: Qualitative results to illustrate the effectiveness of our proposed method and the impact of n on SR (×4) task with $\sigma_y = 0.01$.

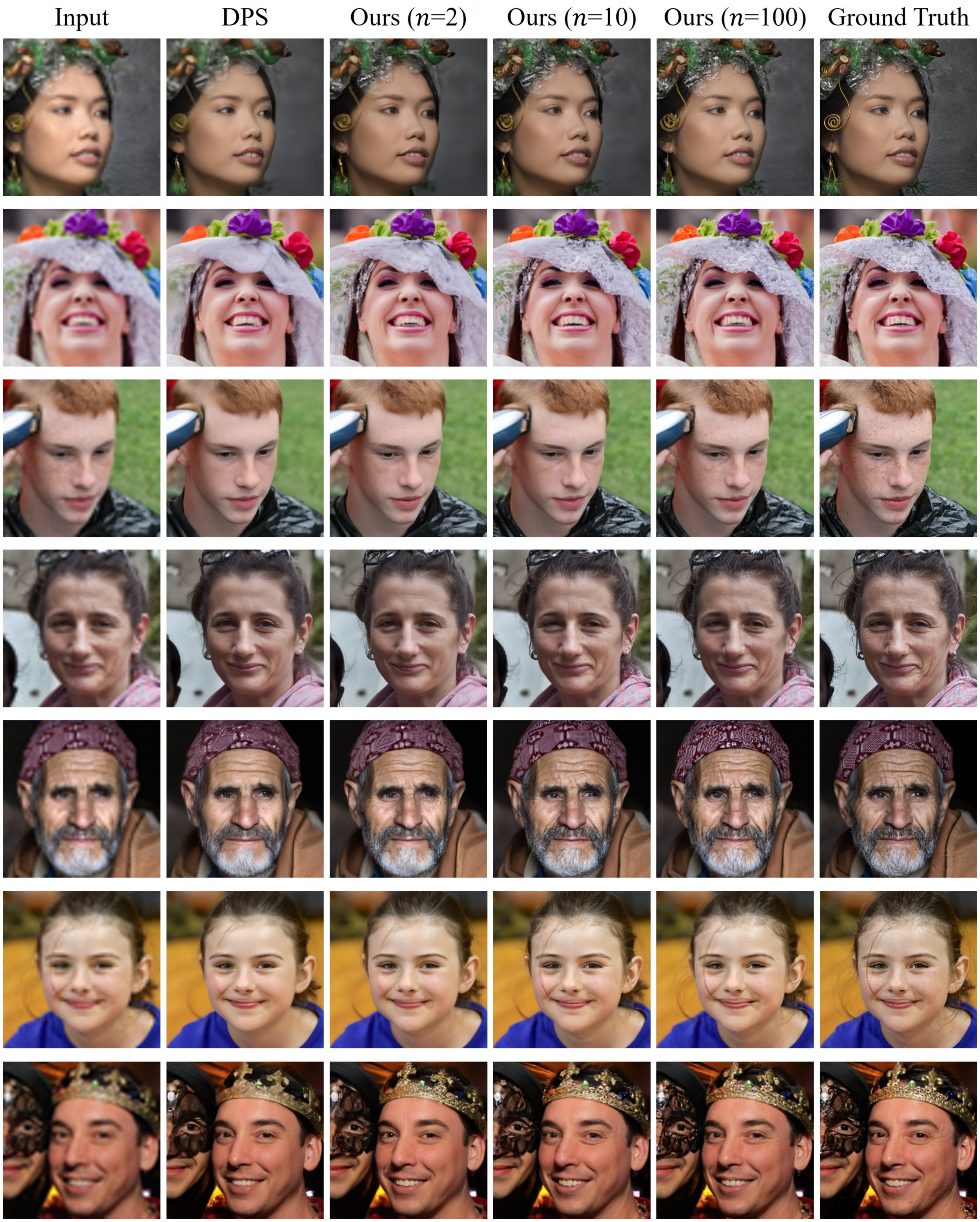

Figure 11: Qualitative results to illustrate the effectiveness of our proposed method and the impact of n on SR (×4) task with $\sigma_y = 0.01$.

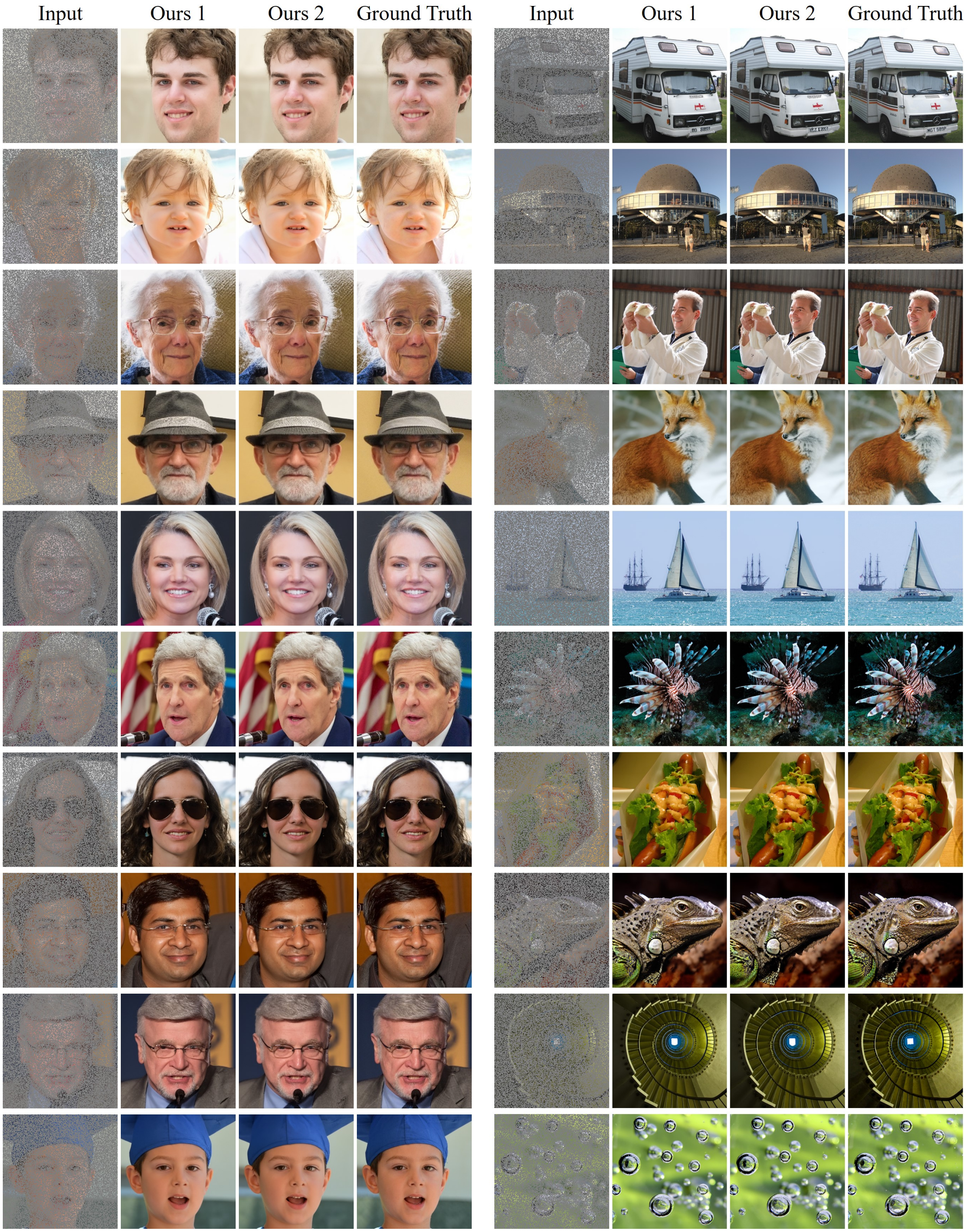

Figure 12: Qualitative inpainting results (Left FFHQ, Right ImageNet) with $\sigma_y = 0.01$.

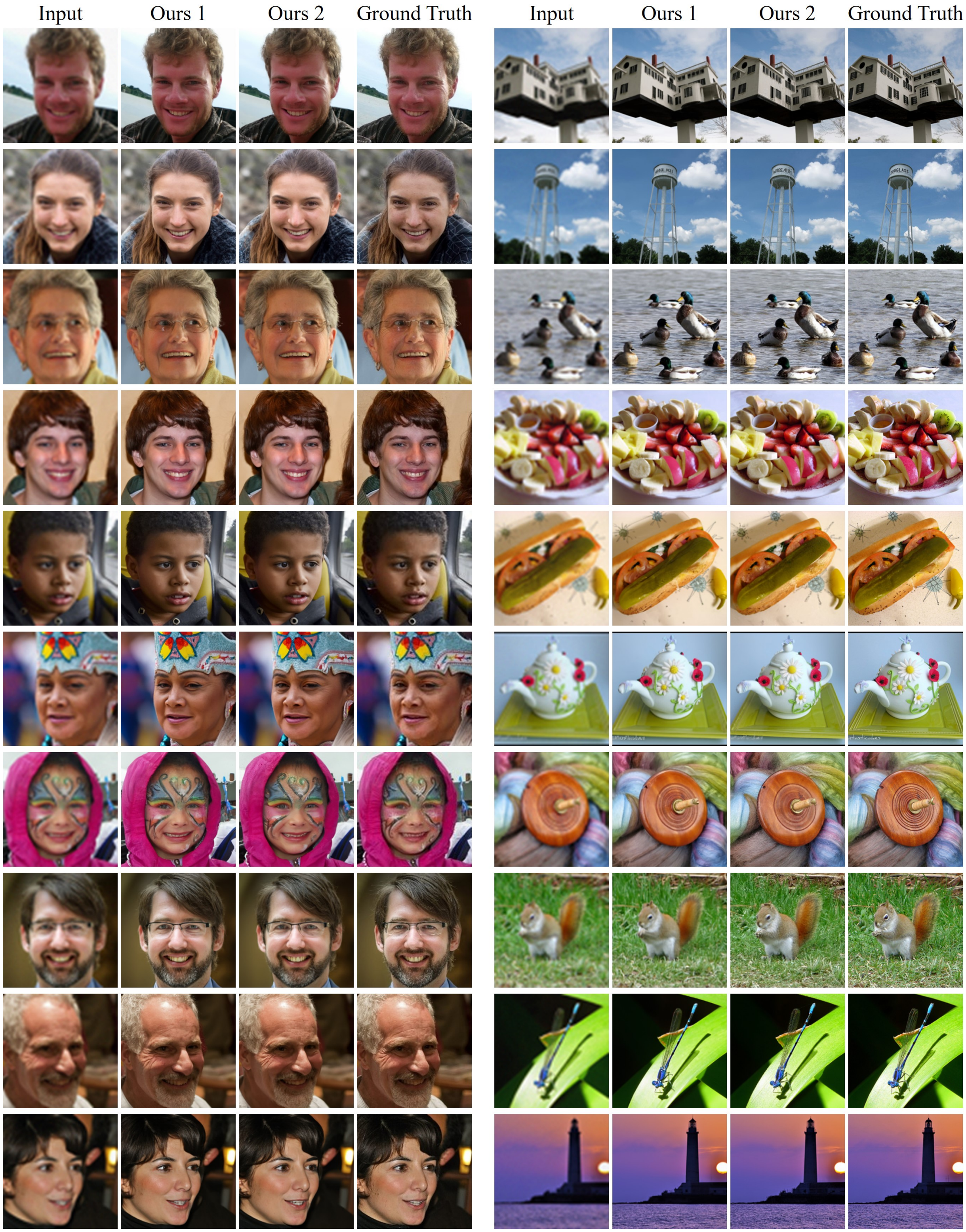

Figure 13: Qualitative SR (×4) results (Left FFHQ, Right ImageNet) with $\sigma_y = 0.01$.

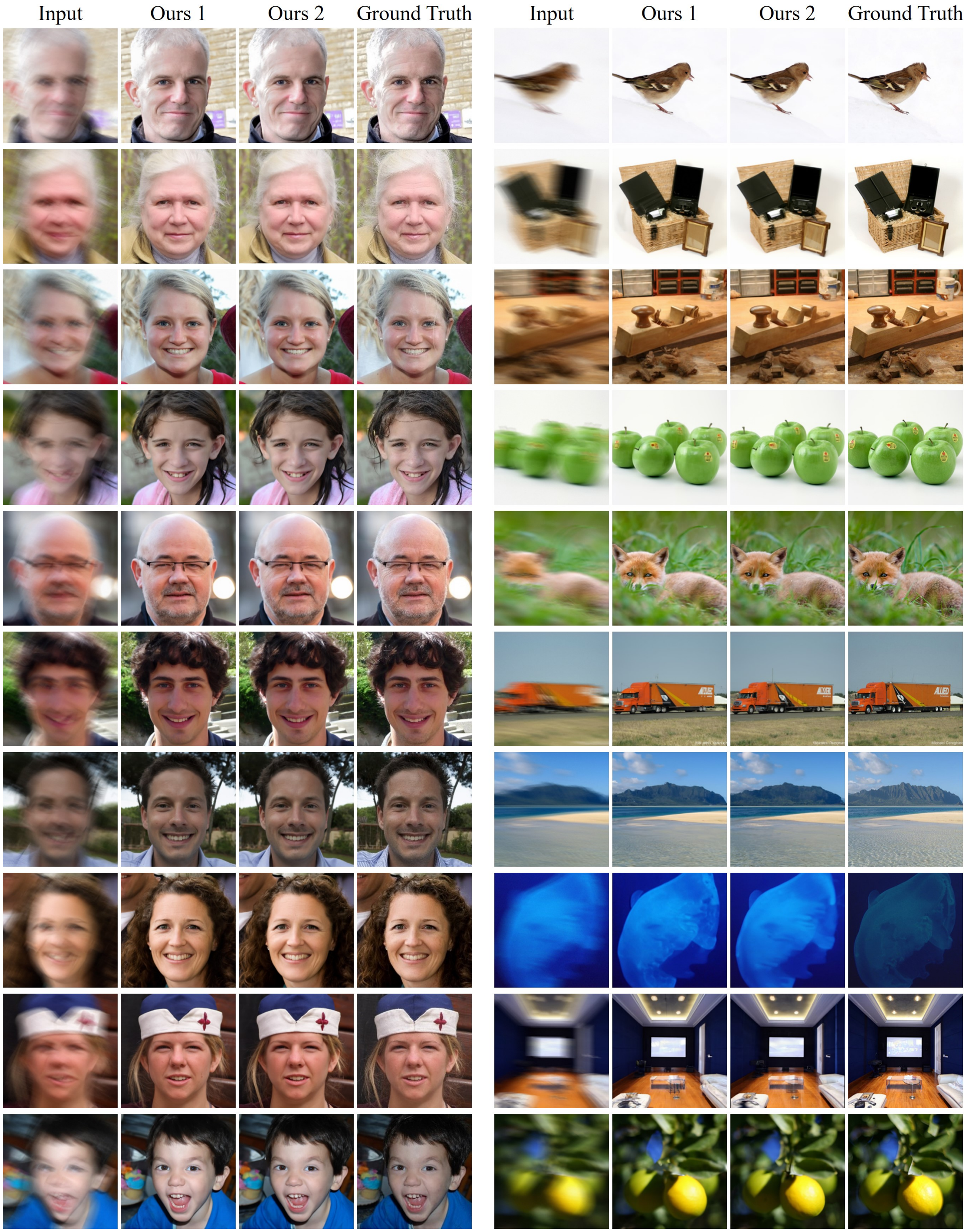

Figure 14: Qualitative motion deblurring results (Left FFHQ, Right ImageNet) with $\sigma_y = 0.01$.

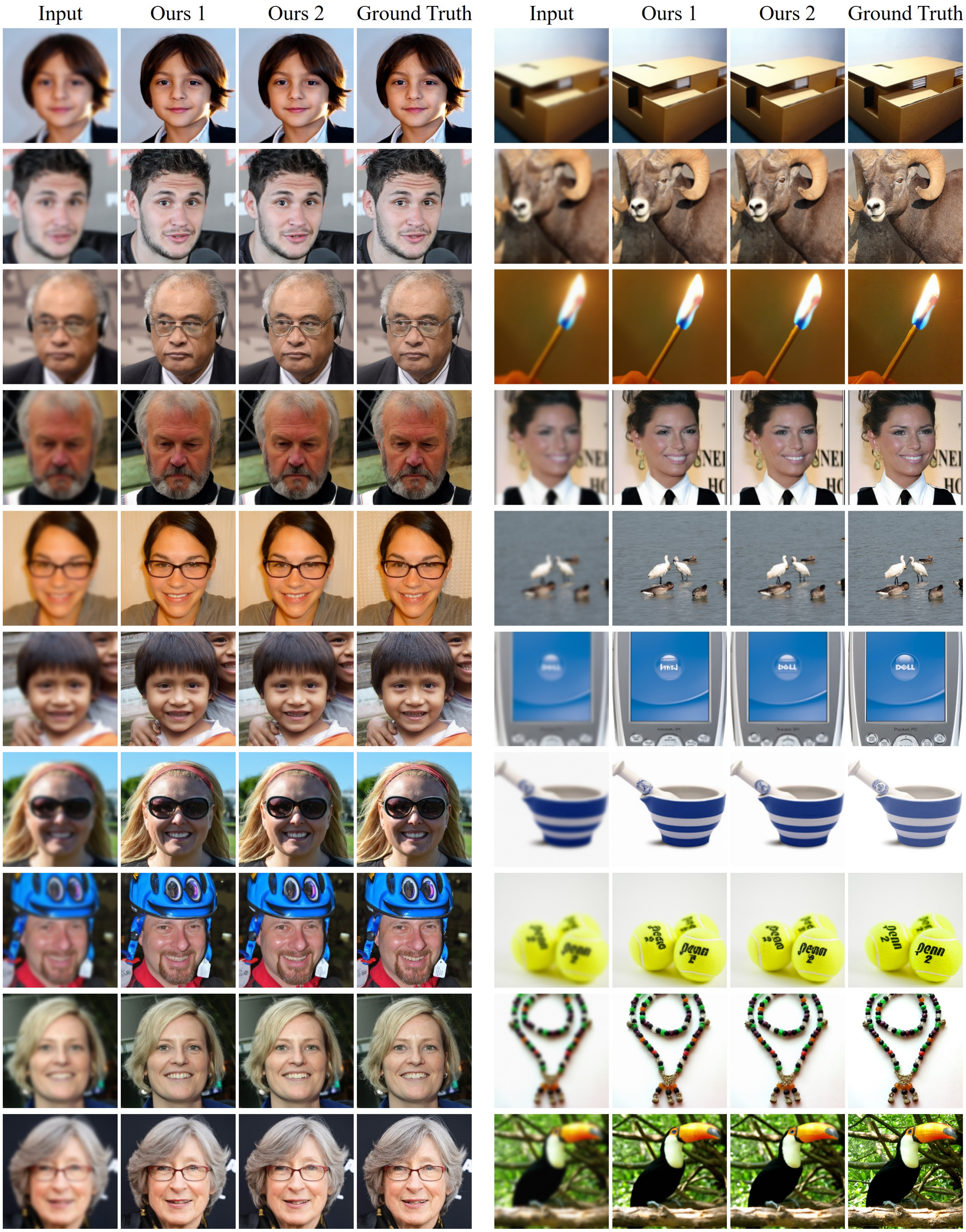

Figure 15: Qualitative Gaussian deblurring results (Left FFHQ, Right ImageNet) with $\sigma_y = 0.01$.