# Appendix

This section includes more results and details that did not fit into the main paper due to space limitation. Particularly, we offer expanded theoretical analysis in §A and implementation details in §B, along with other supportive analysis. These sections provide a deeper understanding and comprehensive context to the research presented in the main body of the paper.

## A. Theoretical Analysis

### A.1. Posterior mean and covariance using Tweedie's formula

**Proposition A.1** ([14, 40]). *Given $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ and $\epsilon \sim \mathcal{N}(0, I)$, denote by $\bar{X}_0 = \mathbb{E}_{X_0 \sim p_t(X_0|X_t=\mathbf{x}_t)}[X_0]$ the posterior mean of $p_t(X_0|X_t = \mathbf{x}_t)$. Then, for the variance preserving SDE or DDPM sampling, $p_t(X_0|X_t = \mathbf{x}_t)$ has mean*

$$\mathbb{E}_{X_0 \sim p_t(X_0|X_t=\mathbf{x}_t)}[X_0] = \frac{\mathbf{x}_t}{\sqrt{\bar{\alpha}_t}} + \frac{(1-\bar{\alpha}_t)}{\sqrt{\bar{\alpha}_t}}\nabla_{\mathbf{x}_t} \log p_t(X_t = \mathbf{x}_t)$$

*and covariance*

$$\mathbb{E}_{X_0 \sim p_t(X_0|X_t=\mathbf{x}_t)}\left[\left(X_0 - \bar{X}_0\right)\left(X_0 - \bar{X}_0\right)^T\right] = \frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}\left(\mathbf{I} + (1-\bar{\alpha}_t)\nabla^2_{\mathbf{x}_t} \log p_t(X_t = \mathbf{x}_t)\right).$$

*Proof.* Given $\mathbf{x}_t = \mu + \sigma\epsilon$, where $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, we know that $\mathbf{x}_t \sim \mathcal{N}\left(\mu, \sigma^2\mathbf{I}\right)$. From [14, Section 2], we have

$$\begin{aligned}
\mathbb{E}[\mu|\mathbf{x}_t] &= \mathbf{x}_t + \sigma^2\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t), \\
\mathbb{V}[\mu|\mathbf{x}_t] &= \sigma^2\left(1+\sigma^2\right)\nabla^2_{\mathbf{x}_t} \log p_t(\mathbf{x}_t),
\end{aligned}$$

where $\mathbb{E}[\mu|\mathbf{x}_t]$ and $\mathbb{V}[\mu|\mathbf{x}_t]$ denote the conditional mean and the conditional variance, respectively. Since $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ in our case, we get

$$\begin{aligned}
\mathbb{E}\left[\sqrt{\bar{\alpha}_t}\mathbf{x}_0|\mathbf{x}_t\right] &= \sqrt{\bar{\alpha}_t}\mathbb{E}[\mathbf{x}_0|\mathbf{x}_t] = \mathbf{x}_t + (1-\bar{\alpha}_t)\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t), \\
\mathbb{V}\left[\sqrt{\bar{\alpha}_t}\mathbf{x}_0|\mathbf{x}_t\right] &= \bar{\alpha}_t\mathbb{V}[\mathbf{x}_0|\mathbf{x}_t] = (1-\bar{\alpha}_t)\left(1+(1-\bar{\alpha}_t)\right)\nabla^2_{\mathbf{x}_t} \log p_t(\mathbf{x}_t),
\end{aligned}$$

which upon rearrangement yields the following:

$$\begin{aligned}
\mathbb{E}_{X_0 \sim p_t(X_0|X_t=\mathbf{x}_t)}[X_0] &= \frac{\mathbf{x}_t}{\sqrt{\bar{\alpha}_t}} + \frac{(1-\bar{\alpha}_t)}{\sqrt{\bar{\alpha}_t}}\nabla_{\mathbf{x}_t} \log p_t(X_t = \mathbf{x}_t) \\
\mathbb{E}_{X_0 \sim p_t(X_0|X_t=\mathbf{x}_t)}\left[\left(X_0 - \bar{X}_0\right)\left(X_0 - \bar{X}_0\right)^T\right] &= \frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}\left(\mathbf{I} + (1-\bar{\alpha}_t)\nabla^2_{\mathbf{x}_t} \log p_t(X_t = \mathbf{x}_t)\right).
\end{aligned}$$

This completes the proof of the statement. □

### A.2. First-order Tweedie sampler

**Theorem A.2.** *(First-order Tweedie Estimator [8]). Given measurements $\mathbf{y} = \mathcal{A}(\mathbf{z}_T) + \mathbf{n}$, $\mathbf{n} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2_{\mathbf{y}}\mathbf{I}\right)$ and the first-order approximation $p_{T-t}(\mathbf{y}|Z_t) \approx p_{T-t}(\mathbf{y}|\bar{Z}_T)$, define the Jensen's gap as:*

$$\mathcal{J} \coloneqq \left|\mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}[p_{T-t}(\mathbf{y}|Z_t)] - p_{T-t}(\mathbf{y}|\bar{Z}_T)\right|,$$

*where $\bar{Z}_T \coloneqq \mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}[Z_T]$. Then, the error due to first-order approximation is upper bounded by*

$$\mathcal{J} \leq \frac{d}{\sqrt{2\pi\sigma^2_{\mathbf{y}}}} \exp\left(-\frac{1}{2\sigma^2_{\mathbf{y}}}\right) \|\nabla_z \mathcal{A}(\mathbf{z})\| m_1,$$

*where $\|\nabla_{\mathbf{z}}\mathcal{A}(\mathbf{z})\| \coloneqq \max_{\mathbf{z}} \|\mathcal{A}(\mathbf{z})\|$ and $m_1 \coloneqq \int \|Z_T - \bar{Z}_T\| \, p_{T-t}(Z_T|Z_t) dZ_T$.*

Since $\|\nabla_{\mathbf{z}}\mathcal{A}(\mathbf{z})\|$ and $m_1$ are finite for most inverse problems, the Jensen's gap goes to zero as $\sigma_{\mathbf{y}} \to \infty$, leading to less approximation error in (2). This setting is of less practical significance because as $\sigma_{\mathbf{y}} \to \infty$, the measurements $\mathbf{y} = \mathcal{A}(\mathbf{z}_T) + \sigma_{\mathbf{y}}\epsilon, \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ provide no meaningful information about $\mathbf{z}_T$. Thus, sampling from the posterior $p_0(Z_T|\mathbf{y}) = p_0(X_0|\mathbf{y})$ is as good as sampling from the prior $p_0(X_0)$. On the other hand, when $\sigma_{\mathbf{y}} \to 0$, the problem is reduced to a noiseless setting which is relatively easier to deal with. In practically relevant settings where $\sigma_y$ is non-zero and finite, the Jensen's gap could be arbitrarily large. This leads to a bias in reconstruction and sub-optimal performance in various tasks as we show in §5.

### A.3. Second-order Tweedie sampler from surrogate loss

**Theorem A.3** (Tweedie Sampler from Surrogate Loss)**.** *Suppose **Assumption 4.2** and **Assumption 4.3** hold. Let $\hat{\mathcal{L}}(\mathbf{y}, Z_t)$ denote the function:*

$$\hat{\mathcal{L}}(\mathbf{y}, Z_t) \coloneqq \log\left(p_{T-t}(\mathbf{y}|\bar{Z}_T)\right) + \log\left(1 - \frac{1-\bar{\alpha}_t}{\bar{\alpha}_t} md - (1-\bar{\alpha}_t) m\, Trace\left(\nabla^2 \log p_{T-t}(Z_t)\right)\right).$$

*For $\lambda = \mathcal{O}(\frac{1}{\sigma_{\mathbf{y}}^2})$ and $\gamma = \mathcal{O}(\frac{\eta}{d})$, the following holds: $\hat{\mathcal{L}}(\mathbf{y}, Z_t) \leq \log p_{T-t}\left(\mathbf{y}|Z_t\right)$. Further, the gradient of $\hat{\mathcal{L}}(\mathbf{y}, Z_t)$ is given by:*

$$\nabla\hat{\mathcal{L}}(\mathbf{y}, Z_t) = -\frac{1}{2\sigma_{\mathbf{y}}^2}\nabla\|\mathbf{y} - \mathbf{A}\bar{Z}_T\|^2 - \frac{(1-\bar{\alpha}_t)m}{\left(1 - \frac{1-\bar{\alpha}_t}{\bar{\alpha}_t} md - (1-\bar{\alpha}_t) m\, Trace\left(\nabla^2 \log p_{T-t}(Z_t)\right)\right)} \nabla Trace\left(\nabla^2 \log p_{T-t}(Z_t)\right).$$

*Proof.* We want to compute the following:

$$\begin{aligned}
\log p_{T-t}(\mathbf{y}|Z_t) &= \log \int p_{T-t}(\mathbf{y}|Z_t, Z_T) p_{T-t}(Z_T|Z_t) dZ_T \\
&\overset{(i)}{=} \log \int p_{T-t}(\mathbf{y}|Z_T) p_{T-t}(Z_T|Z_t) dZ_T \\
&= \log \mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[p_{T-t}(\mathbf{y}|Z_T)\right] \qquad (7)
\end{aligned}$$

where (i) is because $\mathbf{y}$ is independent of $Z_t$ given $Z_T$. Denote by $\bar{Z}_T = \mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[Z_T\right]$. Now, using Taylor series expansion at $\bar{Z}_T$, for some $\tilde{Z}_T \in \mathcal{B}_r\left(\bar{Z}_T\right) \coloneqq \{Z \in \mathbb{R}^d | \|Z - \bar{Z}_T\| \leq r\}, r = \|Z_T - \bar{Z}_T\|$, we get

$$\begin{aligned}
&\log \mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[p_{T-t}(\mathbf{y}|Z_T)\right] \\
&= \log \mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[p_{T-t}(\mathbf{y}|\bar{Z}_T) + \langle \nabla p_{T-t}(\mathbf{y}|Z_t)\,|_{\bar{Z}_T}, Z_T - \bar{Z}_T\rangle + \frac{1}{2}(Z_T - \bar{Z}_T)^T \nabla^2 p_{T-t}(\mathbf{y}|\tilde{Z}_T)(Z_T - \bar{Z}_T)\right] \\
&= \log\left(p_{T-t}(\mathbf{y}|\bar{Z}_T) + \frac{1}{2}\mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[(Z_T - \bar{Z}_T)^T \nabla^2 p_{T-t}(\mathbf{y}|\tilde{Z}_T)(Z_T - \bar{Z}_T)\right]\right),
\end{aligned}$$

where the last step follows from linearity of expectation and the fact that $\langle \nabla p_{T-t}(\mathbf{y}|Z_t)\,|_{\bar{Z}_T}, \mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[Z_T\right] - \bar{Z}_T\rangle = 0$. Since $\log(a+b) = \log(a) + \log(1 + b/a)$ for $a > 0$ and $p_{T-t}(\mathbf{y}|\bar{Z}_T) > 0$ due to **Assumption 4.2**, the above expression simplifies to

$$\begin{aligned}
&\log \mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[p_{T-t}(\mathbf{y}|Z_T)\right] \\
&= \log\left(p_{T-t}(\mathbf{y}|\bar{Z}_T)\right) + \log\left(1 + \frac{\mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[(Z_T - \bar{Z}_T)^T \nabla^2 p_{T-t}(\mathbf{y}|\tilde{Z}_T)(Z_T - \bar{Z}_T)\right]}{2p_{T-t}(\mathbf{y}|\bar{Z}_T)}\right) \\
&= \log\left(p_{T-t}(\mathbf{y}|\bar{Z}_T)\right) + \log\left(1 + \mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[(Z_T - \bar{Z}_T)^T \left(\frac{\nabla^2 p_{T-t}(\mathbf{y}|\tilde{Z}_T)}{2p_{T-t}(\mathbf{y}|\bar{Z}_T)}\right)(Z_T - \bar{Z}_T)\right]\right) \\
&\geq \log\left(p_{T-t}(\mathbf{y}|\bar{Z}_T)\right) + \log\left(1 - m\,\mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[(Z_T - \bar{Z}_T)^T (Z_T - \bar{Z}_T)\right]\right) \\
&= \log\left(p_{T-t}(\mathbf{y}|\bar{Z}_T)\right) + \log\left(1 - m\, Trace\left(\mathbb{E}_{Z_T \sim p_{T-t}(Z_T|Z_t)}\left[(Z_T - \bar{Z}_T)\left(Z_T - \bar{Z}_T\right)^T\right]\right)\right) \\
&= \log\left(p_{T-t}(\mathbf{y}|\bar{Z}_T)\right) + \log\left(1 - m\, Trace\left(\frac{1-\bar{\alpha}_{T-t}}{\bar{\alpha}_{T-t}}\left(I + (1-\bar{\alpha}_{T-t})\nabla^2 \log p_{T-t}(Z_t)\right)\right)\right) \\
&= \log\left(p_{T-t}(\mathbf{y}|\bar{Z}_T)\right) + \log\left(1 - \frac{1-\bar{\alpha}_{T-t}}{\bar{\alpha}_{T-t}} md - (1-\bar{\alpha}_{T-t}) m\, Trace\left(\nabla^2 \log p_{T-t}(Z_t)\right)\right) \coloneqq \hat{\mathcal{L}}(\mathbf{y}, Z_t)
\end{aligned}$$

This completes the proof of the first part, $\hat{\mathcal{L}}(\mathbf{y}, Z_t) \leq \log p_{T-t}\left(\mathbf{y}|Z_t\right)$.

Next, the gradient of the lower bound with respect to $Z_t$ becomes:

$$\begin{aligned}
&\nabla\hat{\mathcal{L}}(\mathbf{y}, Z_t) \\
&= -\frac{1}{2\sigma_{\mathbf{y}}^2}\nabla\|\mathbf{y} - \mathbf{A}\bar{Z}_T\|^2 + \nabla\log\left(1 - \frac{1-\bar{\alpha}_{T-t}}{\bar{\alpha}_{T-t}}md - (1-\bar{\alpha}_{T-t})m\, Trace\left(\nabla^2\log p_{T-t}(Z_t)\right)\right) \\
&= -\frac{1}{2\sigma_{\mathbf{y}}^2}\nabla\|\mathbf{y} - \mathbf{A}\bar{Z}_T\|^2 - \frac{(1-\bar{\alpha}_{T-t})m}{\left(1 - \frac{1-\bar{\alpha}_{T-t}}{\bar{\alpha}_{T-t}}md - (1-\bar{\alpha}_{T-t})m\, Trace\left(\nabla^2\log p_{T-t}(Z_t)\right)\right)}\nabla Trace\left(\nabla^2\log p_{T-t}(Z_t)\right),
\end{aligned}$$

where the last step follows from $\nabla\left(1 - \frac{1-\bar{\alpha}_{T-t}}{\bar{\alpha}_{T-t}}md\right) = 0$. □

**Implication:** From the above result, we have

$$\nabla\hat{\mathcal{L}}(\mathbf{y}, Z_t) \simeq -\lambda\nabla\|\mathbf{y} - \mathbf{A}\bar{Z}_T\|^2 - \gamma\nabla\left(Trace\left(\nabla^2\log p_{T-t}(Z_t)\right)\right),$$

where $\lambda = \mathcal{O}\left(\frac{1}{\sigma_{\mathbf{y}}^2}\right)$ and $\gamma = \mathcal{O}\left(\frac{\eta}{d}\right)$ are hyper-parameters to be tuned in practice.

**Connection with the surrogate loss:** The gradient of the lower bound $\hat{\mathcal{L}}(\mathbf{y}, Z_t)$ is equal to the negative gradient of the surrogate loss function $\mathcal{L}(\mathbf{y}, Z_t)$ introduced in §3.2 and §4, i.e., $\nabla\hat{\mathcal{L}}(\mathbf{y}, Z_t) \simeq -\nabla\mathcal{L}(\mathbf{y}, Z_t)$, when the constants $\lambda$ and $\gamma$ are chosen appropriately. More precisely, as given in the statement of the **Theorem A.3**, these gradients are equal when $\lambda = \frac{-1}{2\sigma_{\mathbf{y}}^2}$ and $\gamma = \frac{-(1-\bar{\alpha}_{T-t})m}{\left(1 - \frac{1-\bar{\alpha}_{T-t}}{\bar{\alpha}_{T-t}}md - (1-\bar{\alpha}_{T-t})m\, Trace(\nabla^2\log p_{T-t}(Z_t))\right)}$. In our implementation, we use $\nabla\mathcal{L}(\mathbf{y}, Z_t)$ that results in proximal gradient *descent* in **Algorithm 1**.

**Remark A.4** (Second-order Tweedie for Gaussian Prior)**.** *Recall from Appendix A.3 that we want to compute*

$$\log p_{T-t}(\mathbf{y}|Z_t) = \log\mathbb{E}_{Z_T\sim p_{T-t}(Z_T|Z_t)}\left[p_{T-t}(\mathbf{y}|Z_T)\right].$$

*Let us suppose that the prior is Gaussian, i.e., $p_T(Z_T) = \mathcal{N}(Z_T; \boldsymbol{\mu}, \mathbf{I})$. Then, the forward and reverse process become Gaussian processes. Therefore, we can compute the posterior mean and covariance analytically using Proposition A.1 as:*

$$\begin{aligned}
\mathbb{E}_{Z_T\sim p_{T-t}(Z_T|Z_t=\mathbf{z}_t)}\left[Z_T\right] &= \frac{\mathbf{z}_t}{\sqrt{\bar{\alpha}_{T-t}}} + \frac{(1-\bar{\alpha}_{T-t})}{\sqrt{\bar{\alpha}_{T-t}}}\nabla_{\mathbf{z}_t}\log p_{T-t}(Z_t = \mathbf{z}_t) \\
&= \frac{\mathbf{z}_t}{\sqrt{\bar{\alpha}_{T-t}}} + \frac{(1-\bar{\alpha}_{T-t})}{\sqrt{\bar{\alpha}_{T-t}}}\left(\sqrt{\bar{\alpha}_{T-t}}\boldsymbol{\mu} - \mathbf{z}_t\right) \\
&= \sqrt{\bar{\alpha}_{T-t}}\mathbf{z}_t + (1-\bar{\alpha}_{T-t})\boldsymbol{\mu}, \qquad (8)
\end{aligned}$$

$$\mathbb{E}_{Z_T\sim p_{T-t}(Z_T|Z_t=\mathbf{z}_t)}\left[\left(Z_T - \bar{Z}_T\right)\left(Z_T - \bar{Z}_T\right)^T\right] = \frac{1-\bar{\alpha}_{T-t}}{\bar{\alpha}_{T-t}}\left(\mathbf{I} + (1-\bar{\alpha}_{T-t})\nabla^2_{\mathbf{z}_t}\log p_{T-t}(Z_t = \mathbf{z}_t)\right) = (1-\bar{\alpha}_{T-t})\,\mathbf{I}. \qquad (9)$$

*Thus, we obtain $p_{T-t}\left(Z_T|Z_t = \mathbf{z}_t\right) = \mathcal{N}\left(Z_T; \sqrt{\bar{\alpha}_{T-t}}\mathbf{z}_t + (1-\bar{\alpha}_{T-t})\,\boldsymbol{\mu}, (1-\bar{\alpha}_{T-t})\,\mathbf{I}\right)$ [4]. Following similar arguments from the proof in Appendix A.3, if we truncate $p_{T-t}\left(\mathbf{y}|Z_T\right)$ up to second-order terms in Taylor's expansion, then the lower bound only has an additive error by appropriately chosen stepsize. Hence, the gradients match up to some scaling factor.*

*Note that our theoretical analysis is provided for pixel-space diffusion models. However, it easily extends to latent diffusion models using proof techniques from PSLD [43]. Importantly, the latent space of latent diffusion models, such as Stable Diffusion [41] is usually Gaussian, which makes STSL a reasonable algorithm in practice.*

[4] Instead of expanding the term inside expectation as in Appendix A.3, we can exactly compute $p_{T-t}\left(\mathbf{y}|Z_t = \mathbf{z}_t\right)$ by its second-order Taylor's expansion around the posterior mean. Therefore, for a Gaussian prior, this second-order approximation is exact. However, a similar treatment requires Hessian for non-Gaussian prior, which is computationally expensive in practice.

### A.4. Computation using Hutchinson's Trace Estimator

Given $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, the trace of the Hessian can be efficiently computed as:

$$\mathbb{E}\left[\epsilon^T \left(\nabla \log p_{T-t}\left(Z_t + \epsilon\right) - \nabla \log p_{T-t}\left(Z_t\right)\right)\right] - \mathcal{O}(\|\epsilon\|^3) \simeq Trace\left(\nabla^2 \log p_{T-t}(Z_t)\right).$$

To see this, for $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, using Taylor series expansion of the score, we get

$$\nabla \log p_{T-t}\left(Z_t + \epsilon\right) \simeq \nabla \log p_{T-t}\left(Z_t\right) + \nabla^2 \log p_{T-t}(Z_t)\epsilon + \mathcal{O}\left(\|\epsilon\|^2\right).$$

Subtracting $\nabla \log p_{T-t}\left(Z_t\right)$ from both sides, and taking projection onto $\epsilon$, we have

$$\epsilon^T \left(\nabla \log p_{T-t}\left(Z_t + \epsilon\right) - \nabla \log p_{T-t}\left(Z_t\right)\right) \simeq \epsilon^T \nabla^2 \log p_{T-t}(Z_t)\epsilon + \mathcal{O}(\|\epsilon\|^3).$$

The claim follows by taking the expectation of both sides and applying Hutchinson's trace estimator [20] as given below:

$$\begin{aligned}\mathbb{E}\left[\epsilon^T \left(\nabla \log p_{T-t}\left(Z_t + \epsilon\right) - \nabla \log p_{T-t}\left(Z_t\right)\right)\right] - \mathcal{O}(\|\epsilon\|^3) &\simeq \mathbb{E}_{\epsilon\sim\mathcal{N}(0,I)}\left[\left(\epsilon^T \nabla^2 \log p_{T-t}(Z_t)\epsilon\right)\right] \\ &= Trace\left(\nabla^2 \log p_{T-t}(Z_t)\right)\end{aligned}$$

The last step above involves an approximation of a higher derivative through an expectation of random projections of perturbed function evaluations. This approach has been well studied in online learning settings and with formal guarantees (e.g., Lemma 2.1 in [15]). In our case, the approximation additionally involves a "centering" with $\epsilon^T \nabla \log p_{T-t}\left(Z_t\right)$. While this terms is zero in expectation, it is useful to keep because as we discuss in Section 3.2, we are evaluating the expectation through stochastic averaging with finitely many steps. This centering decreases the magnitude of each step, thus resulting in variance improvement (and thus a less noisy approximation with a fewer number of steps).

## B. Additional Experimental Evaluation

### B.1. Implementation Details

**Image Inversion:** We follow the same experimental setup as prior works [8, 43], and use the measurement operators provided in their original source code: DPS[5] and PSLD[6]. We employ a Gaussian blur kernel (size $61 \times 61$, $\sigma = 3.0$) for *Gaussian deblurring* and a motion blur kernel (size $61 \times 61$, intensity 0.5) for *motion deblurring* tasks. For *super-resolution*, we use $4\times$ and $8\times$ downsampling as measurement operator. Additionally, we introduce 2% salt and pepper noise for *denoising* and 40% drop rate for *random inpainting* tasks. For *free-form inpainting*, we adopt the 10%-20% damage range as utilized in prior works [10, 45].

Our refinement module in **Algorithm 1** uses the Adam optimizer, with an initial learning rate of $1e-2$ and decrementing by a factor of 0.998 per diffusion time step. This process optimizes the latents with stochastic averaging. Notably, STSL exhibits robustness across various tasks, showing minimal sensitivity to hyper-parameter changes. Therefore, we maintain consistent configurations for all tasks, where $N = 2$, $\eta = 0.02$, $\nu = 2$ and $\lambda = 1$. We use $K = 5$ and $T = 50$ as defualt and conduct extensive ablation studies for free-form image inpainting task in §B.4. Following the experimental setting of P2L[10], we add independent and identically distributed Gaussian noise $\mathcal{N}\left(\mathbf{0}, 0.01^2\right)$ to each pixel.

**Image Editing:** In image editing, we use a single stochastic averaging step $K = 1$ since the latents have been refined during proximal gradient updates. We use $\nu = 0.02$ for the contrastive loss without normalization by the data dimension $d$, $\lambda = 1$ for the measurement loss and the same coefficient for Hutchinson's trace estimator $\eta = 0.02$ as in inversion. More details are elaborated in **Algorithm 2**. For the qualitative demonstration, we compare with NTI[7] and a commercial platform that is publicly available. We conduct the experiments using the latest version of the commercial software by November 2023.

**Reproducibility:** The pseudo-code of STSL for inverse is given in **Algorithm 1** and editing in **Algorithm 2**. All the hyper-parameter details are provided in §5 and §B.1.

### B.2. Computational Complexity

Table 2 provides a comparative analysis of the runtime performance across various state-of-the-art methods. NFEs are computed based on the required reverse and optimization steps. For instance, P2L [10] demands 1000 reverse steps, accompanied

[5] https://github.com/DPS2022/diffusion-posterior-sampling
[6] https://github.com/LituRout/PSLD
[7] https://github.com/google/prompt-to-prompt/

**Algorithm 2:** Second-order Tweedie sampler from Surrogate Loss (STSL) for image inversion and editing task

**Input:** Diffusion time steps $T$, observed $\mathbf{y}$, measurement operator $\mathbf{A}$, encoder $\mathcal{E}$, decoder $\mathcal{D}$, learned score $\boldsymbol{s}_\theta$, target text "*prompt*", text encoder $\Phi$

**Tunable Parameters:** likelihood strength $\lambda$, stochastic averaging steps $K$, second-order correction stepsize $\eta$,

**Output:** Edited Image $\mathcal{D}(Z_T)$

Initialization: $\overrightarrow{Z}_0 = \mathcal{E}(\mathbf{A}^T\mathbf{y})$ ▷ DDIM forward process [35, 48]

**for** $t = 0$ **to** $T-1$ **do**

$\quad \overrightarrow{Z}_{t+1} \leftarrow \sqrt{\frac{\bar{\alpha}_{t+1}}{\bar{\alpha}_t}}\overrightarrow{Z}_t - \left(\sqrt{\frac{1}{\bar{\alpha}_{t+1}}-1} - \sqrt{\frac{1}{\bar{\alpha}_t}-1}\right)\left(\sqrt{1-\bar{\alpha}_t}\right) s_\theta(\overrightarrow{Z}_t, t)$

**end**

Initialization: $Z_0 = \overrightarrow{Z}_T$ ▷ proposed reverse process for image inversion

**for** $t = 0$ **to** $T-1$ **do**

$\quad$ **for** $k = 0$ **to** $K$ **do**

$\quad\quad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ▷ stochastic averaging

$\quad\quad \bar{Z}_T \leftarrow (Z_t + (1-\bar{\alpha}_{T-t})\boldsymbol{s}_\theta(Z_t, T-t))/\sqrt{\bar{\alpha}_{T-t}}$

$\quad\quad Z_t \leftarrow Z_t - \lambda\nabla\|\mathbf{y} - \mathbf{A}\mathcal{D}(\bar{Z}_T)\| - (\eta/d)\nabla\left(\epsilon^T \boldsymbol{s}_\theta\left(Z_t+\epsilon, T-t\right) - \epsilon^T \boldsymbol{s}_\theta\left(Z_t, T-t\right)\right)$

$\quad$ **end**

$\quad \bar{Z}_T \leftarrow (Z_t + (1-\bar{\alpha}_{T-t})\boldsymbol{s}_\theta(Z_t, T-t))/\sqrt{\bar{\alpha}_{T-t}}$

$\quad Z_{t+1} \leftarrow \frac{\sqrt{\alpha_{T-t}}(1-\bar{\alpha}_{T-t-1})}{1-\bar{\alpha}_{T-t}} Z_t + \frac{\sqrt{\bar{\alpha}_{T-t-1}}(1-\alpha_{T-t})}{1-\bar{\alpha}_{T-t}} \bar{Z}_T$

**end**

Initialization: $Z_0 = \overrightarrow{Z}_T$ ▷ proposed reverse process for image editing

**for** $t = 0$ **to** $T-1$ **do**

$\quad \bar{Z}_T \leftarrow (Z_t + (1-\bar{\alpha}_{T-t})\boldsymbol{s}_\theta(Z_t, T-t, \varphi_t))/\sqrt{\bar{\alpha}_{T-t}}$

$\quad f\left(Z_t, T-t, \varphi_t\right) = \sqrt{\bar{\alpha}_{T-t-1}}\bar{Z}_T + \sqrt{1-\bar{\alpha}_{T-t-1}}\sqrt{1-\bar{\alpha}_{T-t}}\boldsymbol{s}_\theta(Z_t, T-t, \varphi_t)$

$\quad \hat{\varphi}_t = \arg\min_{\varphi_t}\|Z_{t+1} - f\left(Z_t, T-t, \varphi_t\right)\|_2^2$ ▷ Null-optimization

$\quad \hat{Z}_{t+1} \leftarrow CAC(Z_t, T-t, \hat{\varphi}_t, \Phi\{\text{"}prompt\text{"}\})$ ▷ Cross-Attention-Control (CAC) [17]

$\quad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

$\quad \bar{Z}_T \leftarrow (Z_t + (1-\bar{\alpha}_{T-t})\boldsymbol{s}_\theta(Z_t, T-t, \Phi\{\text{"}prompt\text{"}\}))/\sqrt{\bar{\alpha}_{T-t}}$

$\quad Z_{t+1} \leftarrow \hat{Z}_{t+1} - \lambda\nabla\|\mathbf{y} - \mathbf{A}\mathcal{D}(\bar{Z}_T)\| - \frac{\eta}{d}\nabla\epsilon^T\left(\boldsymbol{s}_\theta\left(Z_t+\epsilon, T-t, \Phi\{\text{"}prompt\text{"}\}\right) - \boldsymbol{s}_\theta\left(Z_t, T-t, \Phi\{\text{"}prompt\text{"}\}\right)\right)$

**end**

**return** $\mathcal{D}(Z_T)$

by at least one prompt tuning step per iteration, accumulating in a total of 2000 NFEs. The best results of P2L [10] are obtained with around 5000 NFEs, which amounts to **30 mins** of runtime per image. Other baseline methods require 1000 reverse steps. The best results of PSLD/GML-DPS [43] are obtained with 1000 NFEs, which amounts to **12 mins** of runtime per image. Our STSL framework demonstrates *efficiency* by employing only 50 DDIM steps coupled with 5 stochastic averaging steps, resulting in a considerably lower count of 250 NFEs. This translates into significantly lower runtime of **under 3 min** with a considerable gain in performance. Note that the runtime of PDM-solvers is lower because the underlying generative model is smaller compared to large-scale foundation models, such as Stable Diffusion. Despite smaller runtime, PDM-solvers are subpar SoTA solvers [10, 43] leveraging these foundation models.

### B.3. More Qualitative Results

We present extended results of the proposed method and compare with SoTA solvers in motion deblurring (Figure 4), SRx8 (Figure 5), and Gaussian deblurring (Figure 6). Notably, STSL demonstrates superior capability in preserving intricate image details and reducing artifact generation, particularly in text-rich images. This is exemplified in the last images of Figures 4 and 5, where text clarity and legibility are visibly enhanced. Furthermore, unlike other methods that tend to introduce spurious textures, our approach consistently maintains high image fidelity, reinforcing the effectiveness of STSL in complex scenarios.

Our results also showcase the adaptability of STSL in image editing tasks. In Figures 9 and 10, we illustrate that conventional editing methods struggle with corrupted input images, whereas STSL-CAT achieves high-fidelity editing under these conditions. Furthermore, STSL-CAT excels in maintaining the integrity of the image even when the input is not corrupted,


Figure 4. **Qualitative results on motion deblurring:** Odd rows represent the full image, while even rows show a zoomed-in view of the green box. The red boxes indicate artifacts from various methods. STSL demonstrates superior performance in preserving image details while simultaneously minimizing artifacts and fake textures. The competitive baselines: PSLD [43] and P2L[10] introduce artifacts and fake texture that might be mistaken as sharpness of the reconstructed image. Observe the high fidelity text restoration by the proposed approach STSL in the last row.

| **Method** | LPIPS↓ | PSNR↑ | SSIM↑ | $K$ | $T$ | NFEs | Initialization |
|---|---|---|---|---|---|---|---|
| STSL-I (Ours) | 0.279 | 30.61 | 81.53 | 5 | 50 | 250 | Alg. 1 Line 5 |
| STSL-III (Ours) | 0.282 | 30.79 | 82.55 | 5 | 200 | 1000 | Alg. 1 Line 5 |
| STSL-II (Ours) | 0.386 | 29.65 | 77.16 | 5 | 50 | 250 | Gaussian |
| STSL-IV (Ours) | <u>0.311</u> | 30.29 | 81.74 | 5 | 200 | 1000 | Gaussian |
| STSL-V (Ours) | **0.291** | 30.65 | 82.48 | 2 | 1000 | 2000 | Gaussian |
| P2L [10] | 0.321 | 31.29 | **85.16** | 2 | 1000 | 2000 | Gaussian |
| PSLD [43] | 0.344 | **31.54** | 84.20 | 1 | 1000 | 1000 | Gaussian |
| GML-DPS [43] | 0.364 | <u>31.49</u> | 84.00 | 1 | 1000 | 1000 | Gaussian |
| LDPS [43] | 0.379 | 31.34 | 84.45 | 1 | 1000 | 1000 | Gaussian |
| LDIR [16] | 0.386 | 31.24 | <u>84.87</u> | 1 | 1000 | 1000 | Gaussian |
| DPS [8] | 0.368 | 28.96 | 69.89 | 1 | 1000 | 1000 | Gaussian |

Table 5. **Quantitative results of the free-form inpainting task on ImageNet-1K**. STSL-I/III are initialized from the forward latent $Z_0 \sim p_T(Z_0|\mathcal{E}(\mathbf{A}^T\mathbf{y}))$ while all the other methods are initialized with Gaussian noise $Z_0 \sim \pi_d$. As discussed in §B.4, STSL-I/III sometimes leaves small missing areas as shown in Figure 8 even though it better reconstructs unmasked regions of the image. To make a fair comparison, we only consider the methods using the same initialization from the Gaussian noise that successfully inpaint all the missing regions. In this setting, STSL-IV and STSL-V still outperform SoTA solver PSLD [43] and P2L [10] using the same number of NFEs: 1000 and 2000, respectively.

as demonstrated in Figure 11. These qualitative results, supporting the quantitative data presented in Table 4(f), reveal that the integration of CAT with NTI preserve image *content*, such as in areas of the nose and eyes while changing the *style*. The *refinement* of forward latents (§3.2) further contributes to this improvement in rendering details from the corrupt images.

## B.4. Free-form Inpainting

The main body of the paper contains quantitative results on standard datasets. In this section, we provide additional quantitative results on free-form inpainting [10], which targets to *generate* missing pixels in the blank areas as opposed to *restore* corrupted pixels. Following prior works [10, 43], we initialize the reverse process at $Z_0 \sim \pi_d$ in STSL-II/IV/V. STSL-I/III are initialized at the forward latent $Z_0 \sim p_T(Z_0|\mathcal{E}(\mathbf{A}^T\mathbf{y}))$. Table 6 and Table 5 show the quantitative evaluation on FFHQ ($512 \times 512$) and ImageNet ($512 \times 512$), respectively.

We conduct ablation studies to analyze the latency-optimization trade-offs. As shown in Table 5, STSL uses different combinations of stochastic averaging steps $K$ and DDIM steps $T$. When compared with methods with the same number of NFEs, STSL outperforms the SoTA solvers PSLD [43] and P2L [10] in terms of LPIPS and achieves comparable results in terms of PSNR/SSIM. Figure 7 illustrates the qualitative results on ImageNet.

| Method | LPIPS | PSNR | SSIM |
|---|---|---|---|
| STSL (ours) | **0.260** | 31.30 | 87.56 |
| P2L [10] | 0.273 | 32.44 | **91.00** |
| PSLD [43] | 0.312 | 32.42 | 88.58 |
| GML-DPS [43] | 0.335 | **32.45** | 88.66 |
| LDPS [43] | 0.372 | 32.12 | 88.15 |
| LDIR [16] | 0.338 | 32.25 | 90.28 |

Table 6. Additional quantitative results on FFHQ-1K.

**Limitation:** Figure 8 shows the failure cases of our proposed inverse problem solver STSL in free-form inpainting. We observe that the large blocks of missing pixels are embedded into the forward latents in STSL-I/III , which is hard to refine using proximal gradient updates. Therefore, the masked regions of the final reconstruction sometimes contain incomplete pixels. This issue arises due to imperfect encoder-decoder of the Stable Diffusion foundation model [43], and could be partly circumvented by slowing down the diffusion process to $T = 1000$ steps and initializing the reverse process at $Z_0 \sim \pi_d$ as in PSLD [43] and P2L [10]. We recommend following this recipe for free-form inpainting.

The proposed inverse problem solver uses $\mathbf{A}^T$ from DPS [8], which is set to identity for some tasks. It might be better to

use Jax implementation of $A^T$ for improved performance as in P2L [10].

**Future work:** Our approach does not tune the prompt used in the generative foundation model. Integrating prompt-tuning [10] into our pipeline might prove beneficial.

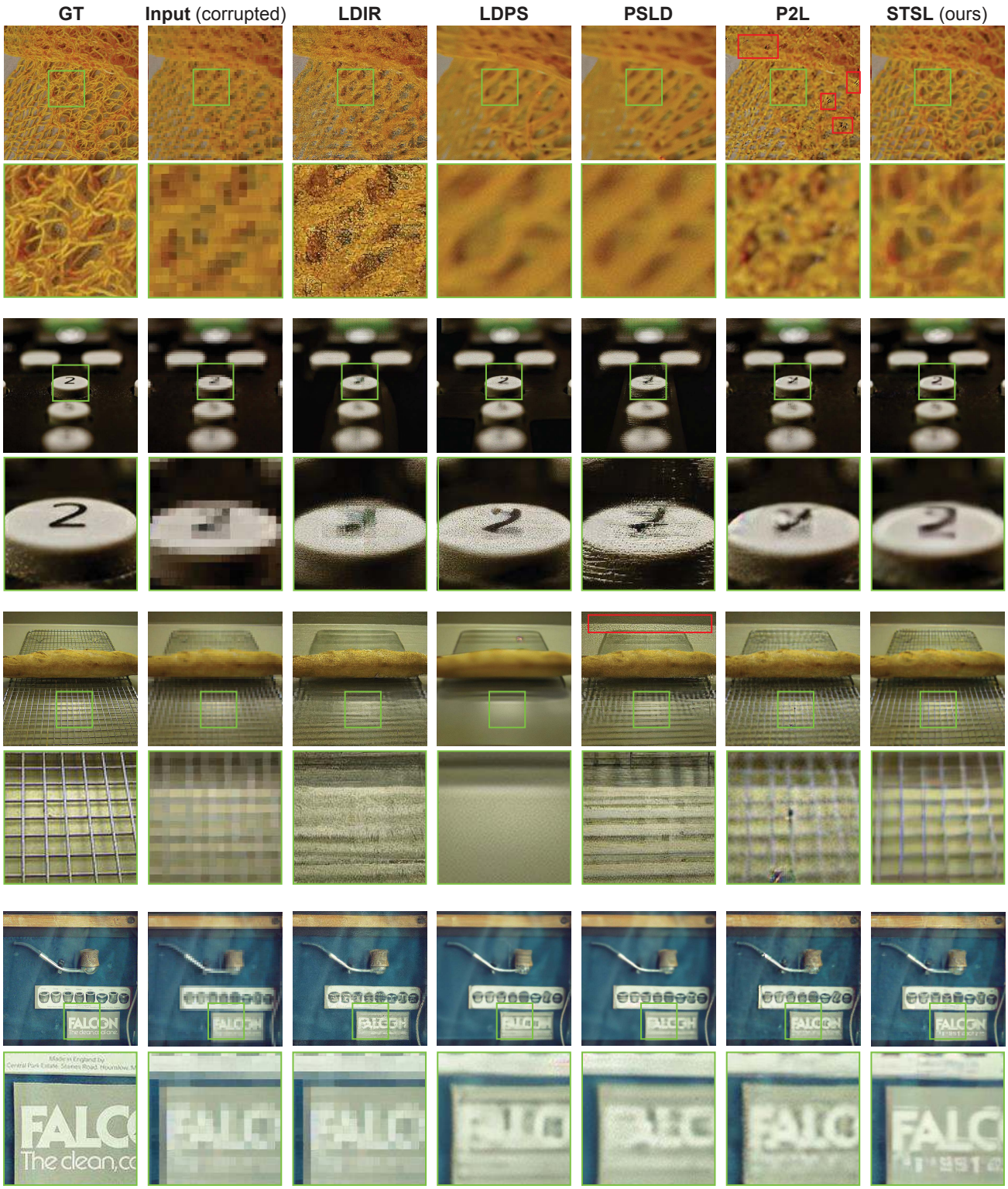


Figure 5. **Qualitative results on SRx8:** Odd rows represent the full image, while even rows show a zoomed-in view of the green box. The red boxes indicate artifacts from various methods. STSL restores image details without introducing artifacts (row 1) and shows its potentiality in restoring images with complicated patterns (row 2 and row 6). The competitive baselines: PSLD [43] and P2L [10] suffer from artifacts that are clearly visible in the highlighted regions.


Figure 6. **Qualitative results on Gaussian deblurring:** Odd rows represent the full image, while even rows show a zoomed-in view of the green box. The red boxes indicate artifacts from various methods. Row 4 and row 8 demonstrate the superior performance of STSL in restoring text and preserving details.

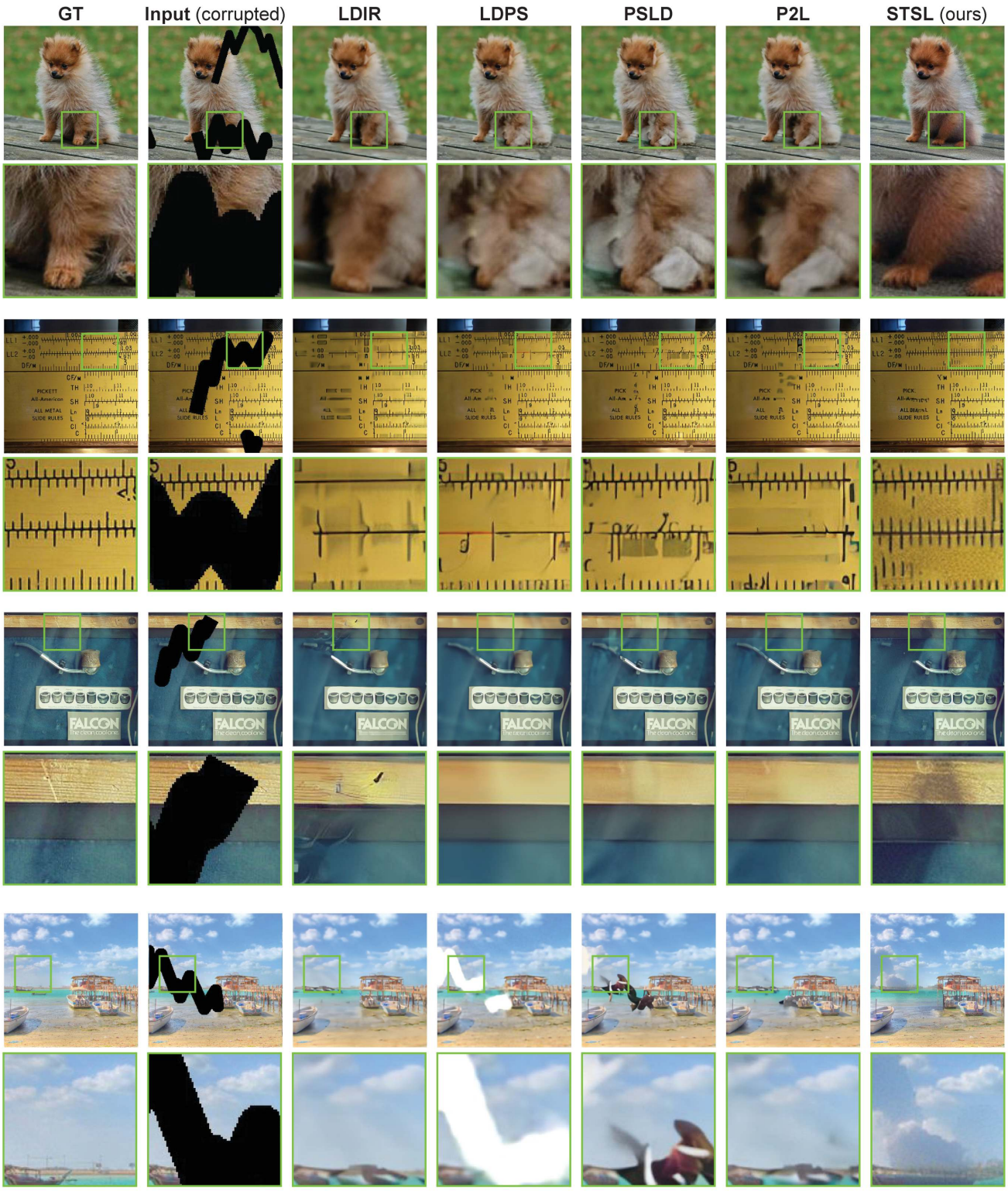


Figure 7. **Qualitative results on free-form inpainting:** Odd rows represent the full image, while even rows show a zoomed-in view of the green box. Note that the model is expected to generate new content that harmonizes with the rest of the pixels, but not necessarily reproduce the same image. This is because the goal is to sample the posterior $p(X|\mathbf{y})$. The outputs from STSL contain more detailed patterns (row 6) and clear edges (row 2&4).

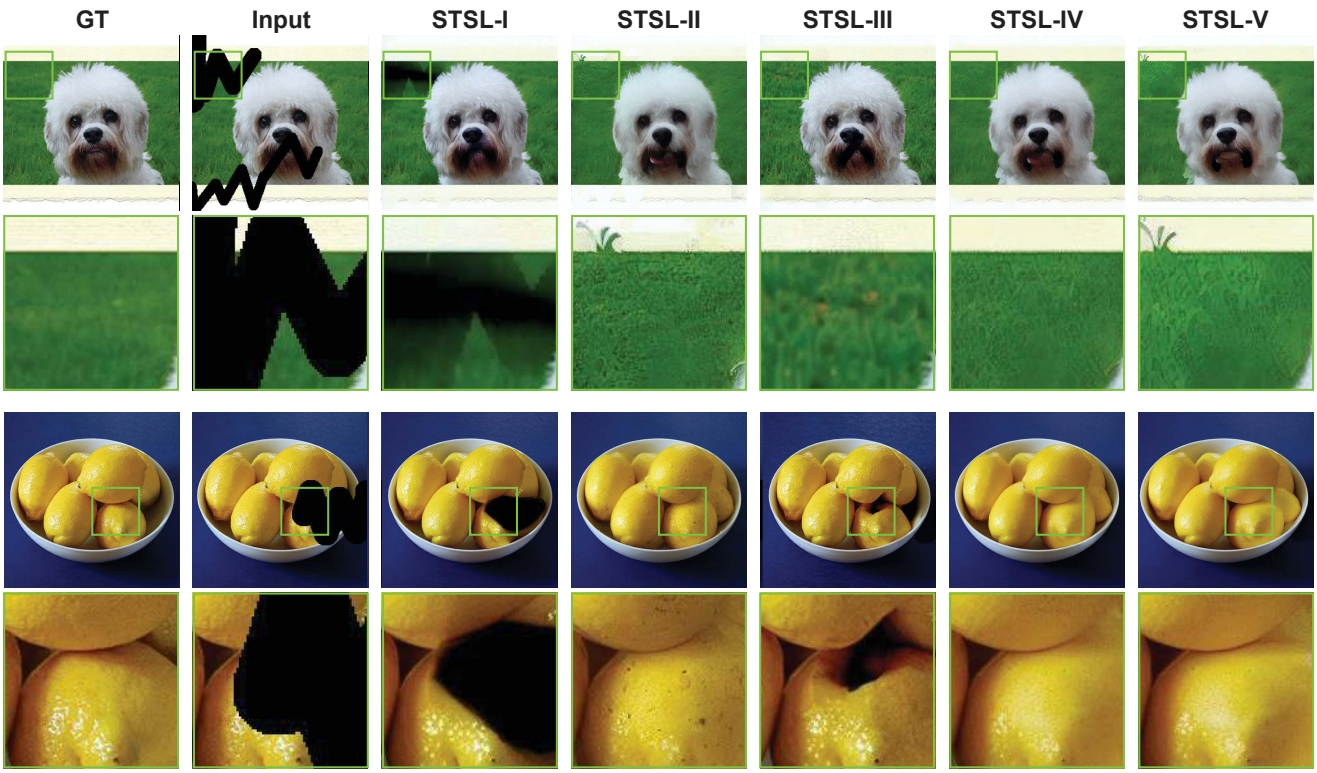


Figure 8. **Failure cases of free-form inpainting:** The restored images appear sharp when initialized with the forward latents $Z_0 \sim p_T(Z_0|\mathcal{E}(\mathbf{A}^T\mathbf{y}))$ in STSL-I/III, while the images with the reverse process initialized at $Z_0 \sim \pi_d$ yield more complete inpainting results (STSL-II/IV/V). One may choose the initialization and the corresponding hyper-parameters as per the requirement in practice.

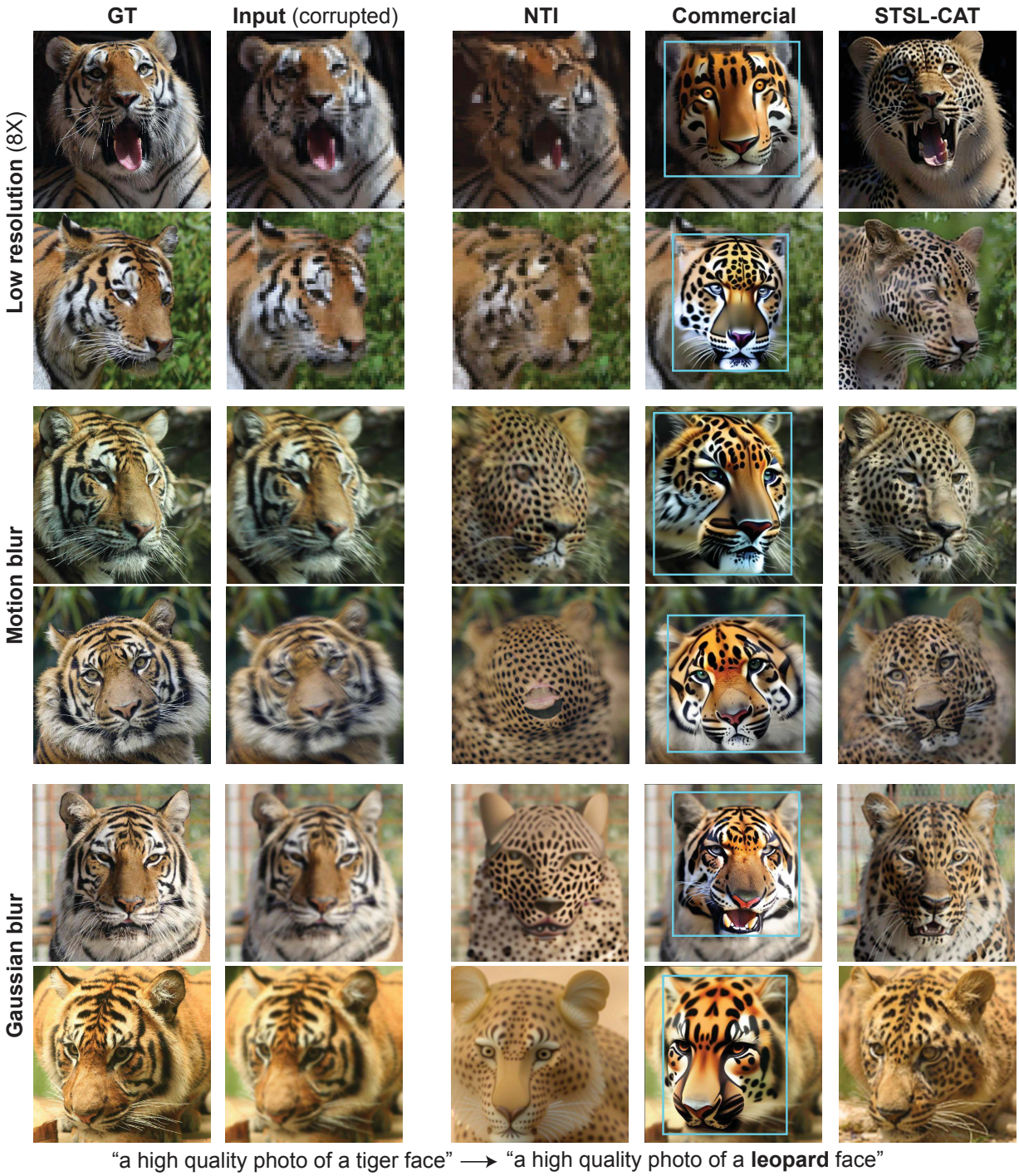


Figure 9. **Qualitative results on image editing on the corrupted images "tiger" to "leopard".** While NTI[35] fails to conduct high-fidelity image editing when various corruptions are presented, the commercial software synthesizes artistic visual objects without preserving the content of the source image. Furthermore, the proposed method STSL-CAT localizes the intended edits without manual intervention, which is necessary for the commercial software.

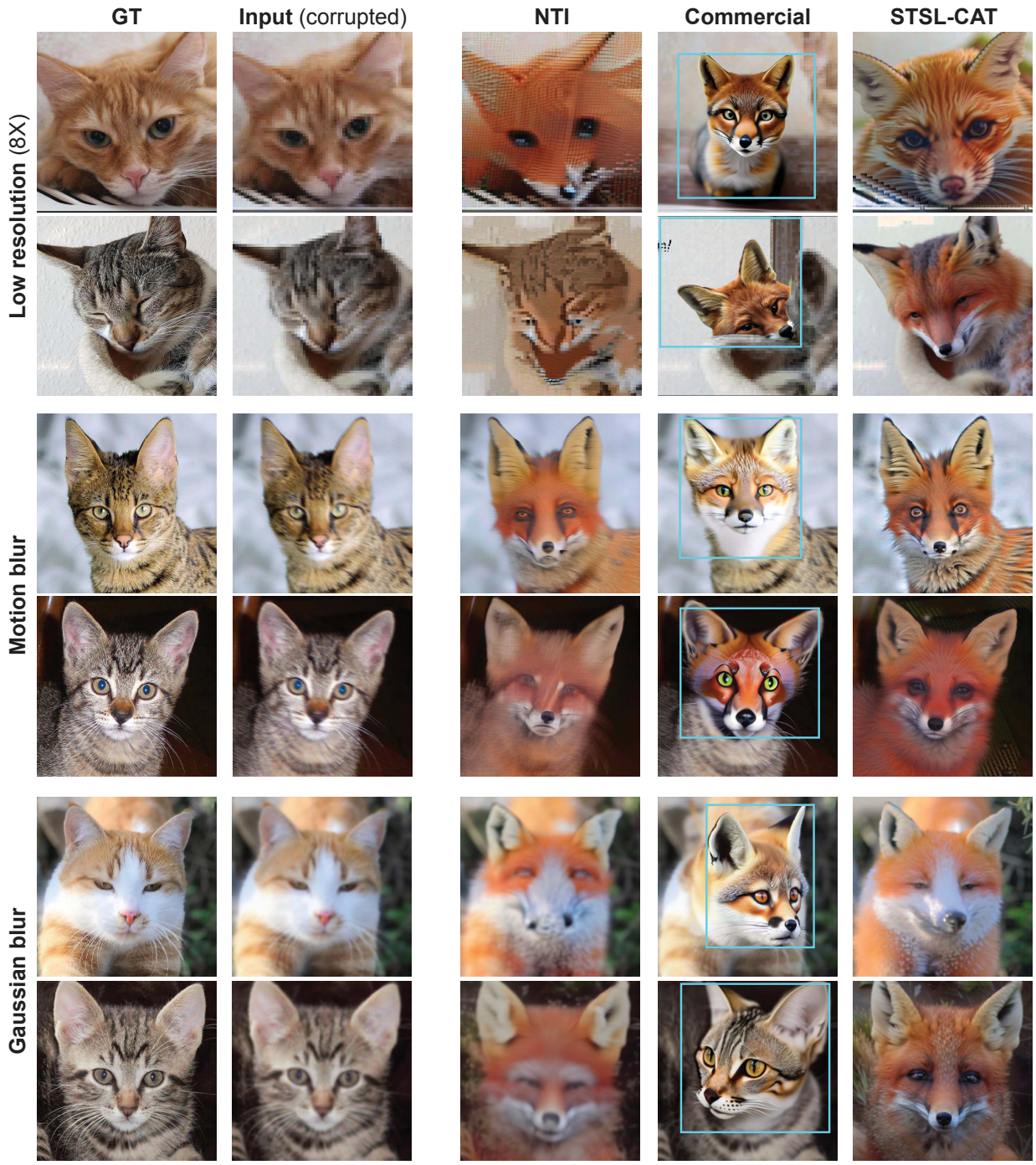


Figure 10. **Qualitative results on image editing on the corrupted images "cat" to "fox".** The proposed method STSL preserves the *content* of the source image while performing *text-guided image editing* on corrupt images.


Figure 11. **Qualitative results on Image editing on the clean images.** Cross attention tuning (CAT) helps preserve image details with NTI [35] (NTI-CAT), and STSL-CAT further enhances the quality of the image by refining the forward latents.