Diffusion Posterior Sampling for General Noisy Inverse Problems

Hyungjin Chung[*1,2], **Jeongsol Kim**[*1], **Michael T. Mccann**[2], **Marc L. Klasky**[2] **& Jong Chul Ye**[1]
[1]KAIST, [2] Los Alamos National Laboratory
{hj.chung, jeongsol, jong.ye}@kaist.ac.kr, {mccann, mklasky}@lanl.gov

任务背景 DIFFUSION POSTERIOR SAMPLING FOR GENERAL NOISY INVERSE PROBLEMS

Diffusion Posterior Sampling (从后验分布 $p(x \mid y)$进行采样)

$$\boldsymbol{Y} = \mathcal{A}(\boldsymbol{X}) + \sigma_{\boldsymbol{y}}\boldsymbol{Z}, \ \boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{0}, I_m).$$

General，Noisy

噪声问题：传统方法在频域中扩散（SVD）

缺点：计算量大、前向模型复杂度增加时更复杂（非线性逆问题）

已有扩散模型集中解决线性反问题：

修复、超分辨率、MRI，CT等问题

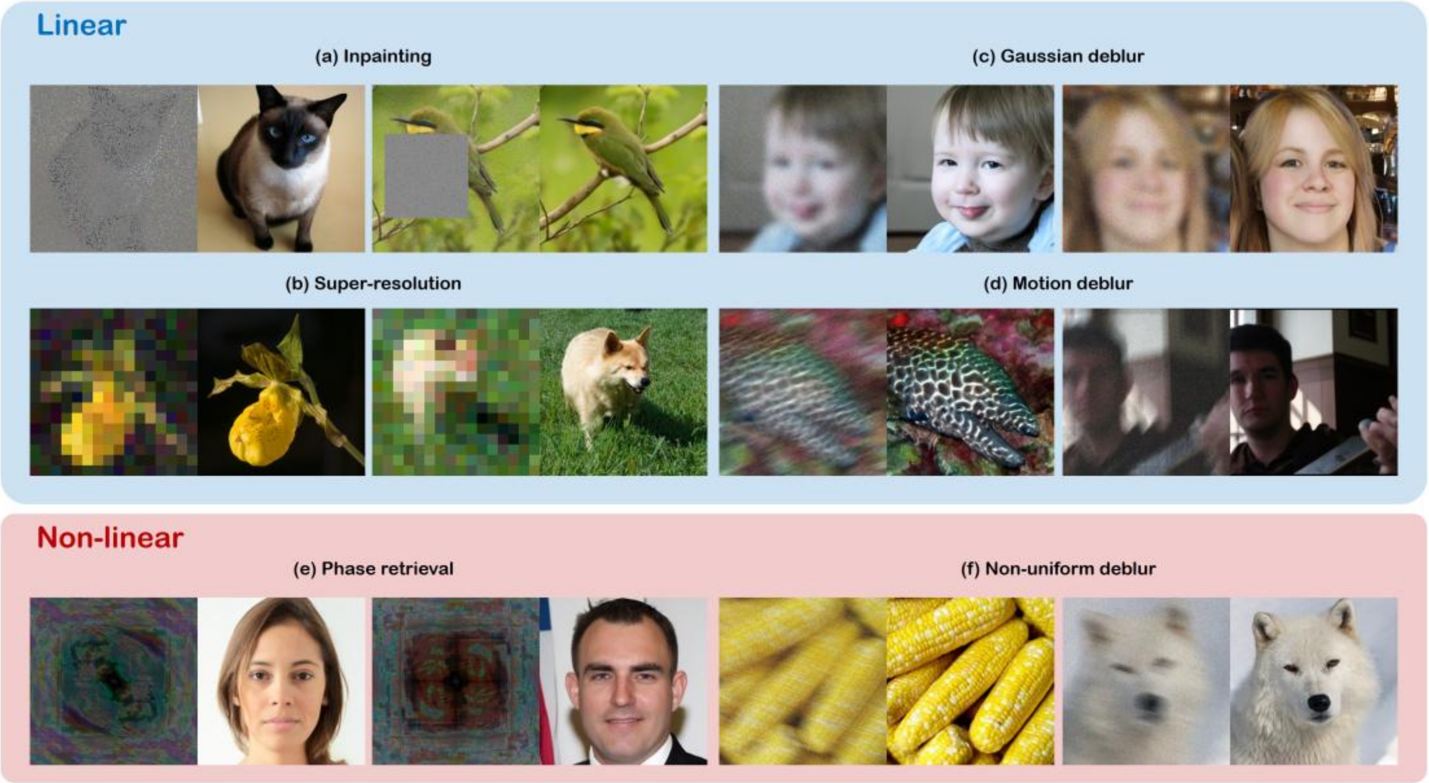


Figure 1: Solving noisy linear, and nonlinear inverse problems with diffusion models. Our reconstruction results (right) from the measurements (left) are shown.

- 图像修复（inpainting）
- 超分辨率（super-resolution）
- 模糊恢复（blur / deblur）
- 相位恢复（phase retrieval）
- 非线性退化恢复（nonlinear deblurring）

DPS做法

DDPM用SDE扩散方程视角表示

SCORE-BASED GENERATIVE MODELING THROUGH STOCHASTIC DIFFERENTIAL EQUATIONS

$$d\boldsymbol{x} = \left[-\frac{\beta(t)}{2}\boldsymbol{x} - \beta(t)\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t)\right]dt + \sqrt{\beta(t)}d\bar{\boldsymbol{w}},$$

$$\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t|\boldsymbol{y}) = \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{y}|\boldsymbol{x}_t).$$

贝叶斯公式

$$d\boldsymbol{x} = \left[-\frac{\beta(t)}{2}\boldsymbol{x} - \beta(t)(\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t) + \underline{\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{y}|\boldsymbol{x}_t)})\right]dt + \sqrt{\beta(t)}d\bar{\boldsymbol{w}},$$

似然函数近似：

不可解——可解

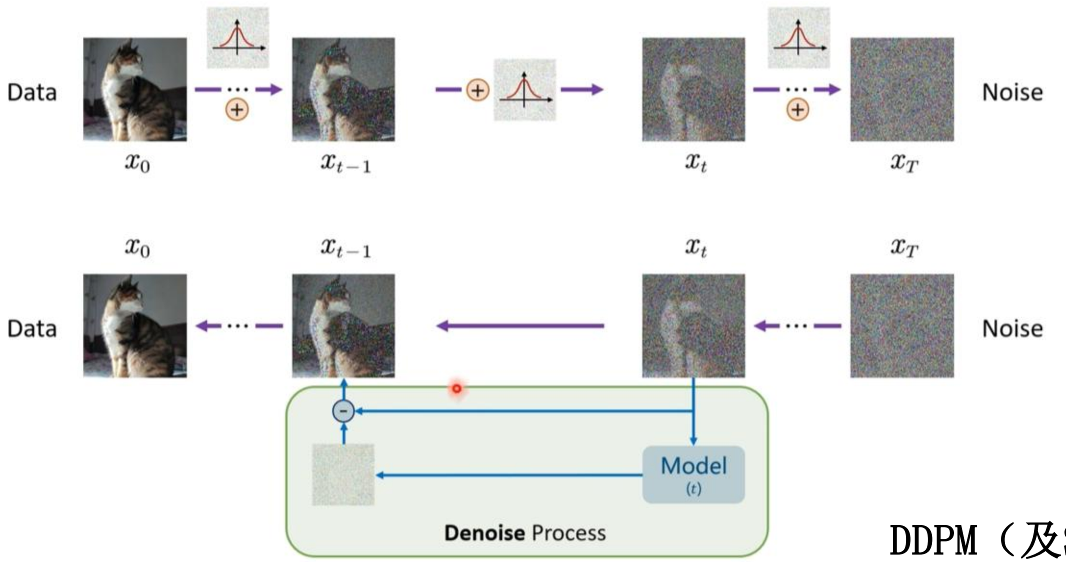
$$p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq p(\boldsymbol{y}|\hat{\boldsymbol{x}}_0), \quad \text{where} \quad \hat{\boldsymbol{x}}_0 := \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t] = \mathbb{E}_{\boldsymbol{x}_0\sim p(\boldsymbol{x}_0|\boldsymbol{x}_t)}[\boldsymbol{x}_0]$$

高斯噪声 $$\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq -\frac{1}{\sigma^2}\nabla_{\boldsymbol{x}_t}\|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0(\boldsymbol{x}_t))\|_2^2$$

泊松噪声 $$\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq -\rho\nabla_{\boldsymbol{x}_t}\|\boldsymbol{y} - \mathcal{A}(\boldsymbol{x}_0)\|_{\boldsymbol{\Lambda}}^2, \quad [\boldsymbol{\Lambda}]_{ii} \triangleq 1/2\boldsymbol{y}_j,$$

高效应对高斯、泊松噪声

Diffusion Model——Score-based generative model



布朗运动 (维纳过程)

$dW=\sqrt{dt}\,Z, Z\sim N(0,1)$

SDE 扩散过程（伊藤过程）

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

drift　　diffusion

(Anderson, 1982)

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

Score

VP-SDE： DDPM 的一般化的形式(DPS的扩散模型)

DDPM（及SMLD）就是SDE方法的特例

$$d\boldsymbol{x}=-\frac{\beta(t)}{2}\boldsymbol{x}dt+\sqrt{\beta(t)}d\boldsymbol{w},$$

$$\boxed{f(X_t,t)=-\frac{1}{2}\beta_t\mathbf{x}, \quad g(t)=\sqrt{\beta(t)}}$$

马尔科夫链　到噪声的过程

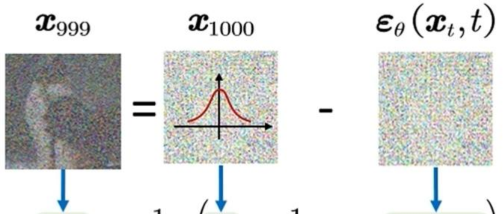
DDPM《一文理清 Diffusion Model 扩散模型 》

$$\boldsymbol{x}_t=\sqrt{\alpha_t}\,\boldsymbol{x}_{t-1}+\sqrt{1-\alpha_t}\,\boldsymbol{\varepsilon}_{t-1}$$

$$\boldsymbol{x}_t=\sqrt{\bar{\alpha}_t}\,\boldsymbol{x}_0+\sqrt{1-\bar{\alpha}_t}\,\boldsymbol{\varepsilon}_t$$

$$d\boldsymbol{x}=\left[-\frac{\beta(t)}{2}\boldsymbol{x}-\beta(t)\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t)\right]dt+\sqrt{\beta(t)}d\bar{\boldsymbol{w}},$$

$$\theta^*=\arg\min_\theta \mathbb{E}_{t\sim U(\varepsilon,1),\boldsymbol{x}(t)\sim p(\boldsymbol{x}(t)|\boldsymbol{x}(0)),\boldsymbol{x}(0)\sim p_{\text{data}}}\left[\|\boldsymbol{s}_\theta(\boldsymbol{x}(t),t)-\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}(t)|\boldsymbol{x}(0))\|_2^2\right],$$



$$\boldsymbol{x}_{t-1}=\frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t-\frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\varepsilon}_\theta(\boldsymbol{x}_t,t)\right)+\sqrt{\beta_t}\,\boldsymbol{\varepsilon}$$

使用SDE作为分布映射模型的好处：很多SDE数值估计方法、更多寻找生成方法

Diffusion Posterior Sampling

$$d\boldsymbol{x} = \left[-\frac{\beta(t)}{2}\boldsymbol{x} - \beta(t)\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t)\right]dt + \sqrt{\beta(t)}d\bar{\boldsymbol{w}},$$

p(x|y)=p(y|x)p(x)/p(y)

$$\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t|\boldsymbol{y}) = \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{y}|\boldsymbol{x}_t).$$

$$d\boldsymbol{x} = \left[-\frac{\beta(t)}{2}\boldsymbol{x} - \beta(t)(\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{y}|\boldsymbol{x}_t))\right]dt + \sqrt{\beta(t)}d\bar{\boldsymbol{w}},$$

Score，由前面的 Measurement matching term
扩散模型已知 （我们的目标）

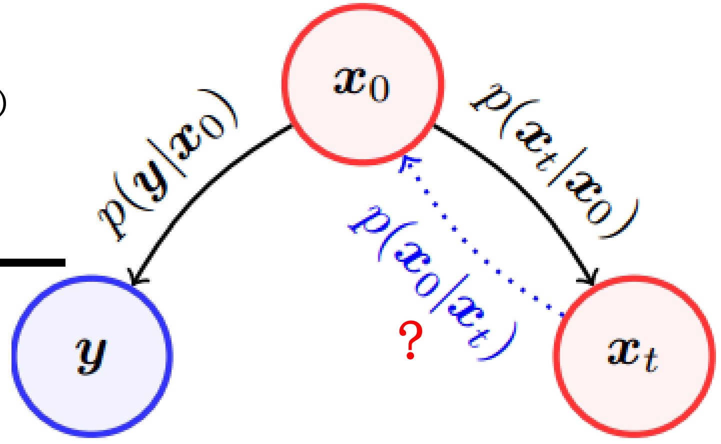
$$\boldsymbol{Y} = \mathcal{A}(\boldsymbol{X}) + \sigma_{\boldsymbol{y}}\boldsymbol{Z},\ \boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{0}, I_m).$$

DPS的公式推导——E[x0|xt]

我们的目标： $$\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t|\boldsymbol{y}) = \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{y}|\boldsymbol{x}_t).$$

Score，由前面的扩散模型已知

Measurement matching term（我们的目标，intractable）



intractable ← $$p(\boldsymbol{y}|\boldsymbol{x}_t) = \int p(\boldsymbol{y}|\boldsymbol{x}_0,\boldsymbol{x}_t)p(\boldsymbol{x}_0|\boldsymbol{x}_t)d\boldsymbol{x}_0$$
$$= \int p(\boldsymbol{y}|\boldsymbol{x}_0)p(\boldsymbol{x}_0|\boldsymbol{x}_t)d\boldsymbol{x}_0, \qquad (7)$$

通过近似

Tractable $$p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq p(\boldsymbol{y}|\hat{\boldsymbol{x}}_0), \quad \text{where} \quad \hat{\boldsymbol{x}}_0 := \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t] = \mathbb{E}_{\boldsymbol{x}_0\sim p(\boldsymbol{x}_0|\boldsymbol{x}_t)}[\boldsymbol{x}_0]$$

描述内容：根据当前xt所能求得最逼近x0的数

$$\boldsymbol{Y} = \mathcal{A}(\boldsymbol{X}) + \sigma_{\boldsymbol{y}}\boldsymbol{Z},$$
$$p(\boldsymbol{y}|\boldsymbol{x}_0) = \frac{1}{\sqrt{(2\pi)^n\sigma^{2n}}}\exp\left[-\frac{\|\boldsymbol{y}-\mathcal{A}(\boldsymbol{x}_0)\|_2^2}{2\sigma^2}\right]$$

Tweedie引理推导——E[x0|xt]

$$p(\boldsymbol{y}|\boldsymbol{\eta}) = p_0(\boldsymbol{y})\exp(\boldsymbol{\eta}^\top T(\boldsymbol{y}) - \varphi(\boldsymbol{\eta})), \longrightarrow (\nabla_{\boldsymbol{y}}T(\boldsymbol{y}))^\top\mathbb{E}[\boldsymbol{\eta}|\boldsymbol{y}] = \nabla_{\boldsymbol{y}}\log p(\boldsymbol{y}) - \nabla_{\boldsymbol{y}}\log p_0(\boldsymbol{y})$$

DDPM: $$\boldsymbol{x}_t = \underbrace{\sqrt{\bar{\alpha}_t}}_{\mu_t}\boldsymbol{x}_0 + \underbrace{\sqrt{1-\bar{\alpha}_t}}_{\sigma_t}\boldsymbol{\varepsilon}_t$$

$$p(\boldsymbol{x}_t|\boldsymbol{x}_0) = \frac{1}{(2\pi(1-\bar{\alpha}(t)))^{d/2}}\exp\left(-\frac{\|\boldsymbol{x}_t-\sqrt{\bar{\alpha}(t)}\boldsymbol{x}_0\|^2}{2(1-\bar{\alpha}(t))}\right) \longrightarrow \hat{\boldsymbol{x}}_0 := \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t] = \frac{1}{\sqrt{\bar{\alpha}(t)}}(\boldsymbol{x}_t + (1-\bar{\alpha}(t))\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t))$$

DPS的公式解释——E[x0|xt]

DPS

$$\hat{\boldsymbol{x}}_0 := \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t] = \frac{1}{\sqrt{\bar{\alpha}(t)}}(\boldsymbol{x}_t + (1-\bar{\alpha}(t))\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t))$$

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{t\sim U(\varepsilon,1),\boldsymbol{x}(t)\sim p(\boldsymbol{x}(t)|\boldsymbol{x}(0)),\boldsymbol{x}(0)\sim p_{\text{data}}}\left[\|\boldsymbol{s}_\theta(\boldsymbol{x}(t),t) - \nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}(t)|\boldsymbol{x}(0))\|_2^2\right],$$

$$\hat{\boldsymbol{x}}_0 \simeq \frac{1}{\sqrt{\bar{\alpha}(t)}}(\boldsymbol{x}_t + (1-\bar{\alpha}(t))\boldsymbol{s}_{\theta^*}(\boldsymbol{x}_t,t)).$$

DDPM

$$\boldsymbol{x}_0 = \frac{\boldsymbol{x}_t - \sqrt{1-\bar{\alpha}_t}\,\boldsymbol{\varepsilon}_t}{\sqrt{\bar{\alpha}_t}}$$

$$\operatorname*{argmin}_{\theta} \frac{1}{2\sigma_q^2(t)}\frac{(1-\alpha_t)^2}{(1-\bar{\alpha}_t)\alpha_t}[\|\boldsymbol{\varepsilon}_t - \underline{\hat{\boldsymbol{\varepsilon}}_\theta(\boldsymbol{x}_t,t)}\|_2^2]$$

DDPM $\boldsymbol{x}_t = \underbrace{\sqrt{\bar{\alpha}_t}}_{\mu_t}\boldsymbol{x}_0 + \underbrace{\sqrt{1-\bar{\alpha}_t}}_{\sigma_t}\boldsymbol{\varepsilon}_t$ $\quad \varepsilon_t = \dfrac{x_t - \sqrt{\bar{\alpha}_t}\,x_0}{\sqrt{1-\bar{\alpha}_t}}$

$$\nabla_{x_t}\log p_t(x_t|x_0) = -\nabla_{x_t}\frac{(x_t-\mu_t)^2}{2\sigma_t^2} = -\frac{2(x_t-\mu_t)}{2\sigma_t^2} = -\frac{x_t-\mu_t}{\sigma_t^2}$$

$$= -\frac{x_t - \sqrt{\bar{\alpha}_t}\,x_0}{1-\bar{\alpha}_t} = -\underline{\frac{\varepsilon_t}{\sqrt{1-\bar{\alpha}_t}}}$$

3.3 ESTIMATING SCORES FOR THE SDE

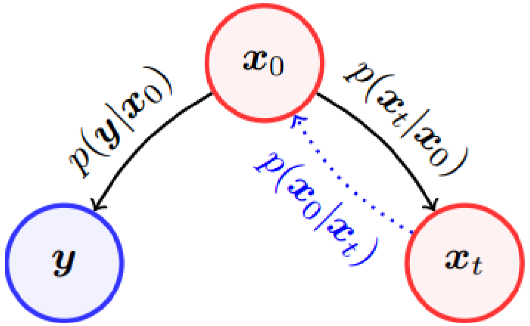
The score of a distribution can be estimated by training a score-based model on samples with score matching (Hyvärinen, 2005; Song et al., 2019a). To estimate $\nabla_{\mathbf{x}}\log p_t(\mathbf{x})$, we can train a time-dependent score-based model $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x},t)$ via a continuous generalization to Eqs. (1) and (3):

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_t\Big\{\lambda(t)\mathbb{E}_{\mathbf{x}(0)}\mathbb{E}_{\mathbf{x}(t)|\mathbf{x}(0)}\big[\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}(t),t) - \nabla_{\mathbf{x}(t)}\log p_{0t}(\mathbf{x}(t)\mid\mathbf{x}(0))\|_2^2\big]\Big\}. \quad (7)$$

Here $\lambda : [0,T] \to \mathbb{R}_{>0}$ is a positive weighting function, t is uniformly sampled over $[0,T]$, $\mathbf{x}(0) \sim p_0(\mathbf{x})$ and $\mathbf{x}(t) \sim p_{0t}(\mathbf{x}(t) \mid \mathbf{x}(0))$. With sufficient data and model capacity, score matching ensures that the optimal solution to Eq. (7), denoted by $\mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x},t)$, equals $\nabla_{\mathbf{x}}\log p_t(\mathbf{x})$ for almost all $\mathbf{x}$ and t. As in SMLD and DDPM, we can typically choose $\lambda \propto 1/\mathbb{E}\big[\|\nabla_{\mathbf{x}(t)}\log p_{0t}(\mathbf{x}(t)\mid\mathbf{x}(0))\|_2^2\big]$. Note that Eq. (7) uses denoising score matching, but other score matching objectives, such as sliced

SCORE-BASED GENERATIVE MODELING THROUGH STOCHASTIC DIFFERENTIAL EQUATIONS

DPS的公式推导——创新点：p(y|xt)近似



$$
\begin{aligned}
p(\boldsymbol{y}|\boldsymbol{x}_t) &= \int p(\boldsymbol{y}|\boldsymbol{x}_0, \boldsymbol{x}_t) p(\boldsymbol{x}_0|\boldsymbol{x}_t) d\boldsymbol{x}_0 \\
&= \int p(\boldsymbol{y}|\boldsymbol{x}_0) p(\boldsymbol{x}_0|\boldsymbol{x}_t) d\boldsymbol{x}_0, \\
&= \mathbb{E}_{\boldsymbol{x}_0 \sim p(\boldsymbol{x}_0|\boldsymbol{x}_t)} \left[p(\boldsymbol{y}|\boldsymbol{x}_0) \right]
\end{aligned}
$$

含义：对给定x0的，对**p(y｜x0)** 进行期望的结果

我们的目标：不需要对所有可能的x0积分，
而直接用$\widehat{x_0}$（期望）去近似

Jensen不等式应用：

$$f(\mathbb{E}[x]) \leq \mathbb{E}[f(x)]$$

$$\longrightarrow p(y|x_t) \geq p\left(y|\mathbb{E}_{x_0|x_t}[x_0]\right) \longrightarrow$$

$$p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq p(\boldsymbol{y}|\hat{\boldsymbol{x}}_0)$$

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y}|\hat{\boldsymbol{x}}_0),$$

量化近似误差：Jensen gap： $\mathcal{J}(f, \boldsymbol{x} \sim p(\boldsymbol{x})) = \mathbb{E}[f(\boldsymbol{x})] - f(\mathbb{E}[\boldsymbol{x}])$

定理给定的Jensen gap上界： $\boldsymbol{Y} = \mathcal{A}(\boldsymbol{X}) + \sigma_{\boldsymbol{y}} \boldsymbol{Z},\ \boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_m).$

$$\mathcal{J} \leq \frac{d}{\sqrt{2\pi\sigma^2}} e^{-1/2\sigma^2} \|\nabla_{\boldsymbol{x}} \mathcal{A}(\boldsymbol{x})\| m_1$$

$\|\nabla_x A(x)\|$ 是测量模型的梯度，它的最大值 $\|\nabla_x A(x)\|$ 是在 x 上的梯度的最大值。
m_1 是一个常数，表示 x_0 **和** $\hat{x}_0$ 之间的距离的加权平均：

$$m_1 = \int \|x_0 - \hat{x}_0\| p(x_0|x_t) dx_0$$

当噪声较大（即 σ^2较大）时，x_0 和 $\widehat{x_0}$的差异变得较小，
近似效果更好。

创新1：在噪声较大的情况下，DPS 方法能够有效工作

DPS measurement models_Gaussian &Poisson Noise

$$\boldsymbol{Y} = \mathcal{A}(\boldsymbol{X}) + \sigma_{\boldsymbol{y}} \boldsymbol{Z},$$

- **Gaussian Noise**

$$p(y|x_0) = \frac{1}{\sqrt{(2\pi)^n \det(\Sigma)}} \exp\left(-\frac{1}{2}(y-\mu)^T \Sigma^{-1}(y-\mu)\right)$$

$$p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq p(\boldsymbol{y}|\hat{\boldsymbol{x}}_0)$$

$$p(y|x_0) = \mathcal{N}(y|A(x_0), \sigma^2 I) = \frac{1}{\sqrt{(2\pi)^n \sigma^{2n}}} \exp\left[-\frac{\|\boldsymbol{y} - \mathcal{A}(\boldsymbol{x}_0)\|_2^2}{2\sigma^2}\right]$$

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq -\frac{1}{\sigma^2} \nabla_{\boldsymbol{x}_t} \|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0(\boldsymbol{x}_t))\|_2^2$$

- **Poisson Noise**

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq -\rho \nabla_{\boldsymbol{x}_t} \|\boldsymbol{y} - \mathcal{A}(\boldsymbol{x}_0)\|_{\boldsymbol{\Lambda}}^2, \quad [\boldsymbol{\Lambda}]_{ii} \triangleq 1/2\boldsymbol{y}_j,$$

$$p(\boldsymbol{y}|\boldsymbol{x}_0) = \prod_{j=1}^{n} \frac{[\mathcal{A}(\boldsymbol{x}_0)]_j^{\boldsymbol{y}_j} \exp\left[[-\mathcal{A}(\boldsymbol{x}_0)]_j\right]}{\boldsymbol{y}_j\,!},$$

$$p(\boldsymbol{y}|\boldsymbol{x}_0) \to \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi[\mathcal{A}(\boldsymbol{x}_0)]_j}} \exp\left(-\frac{(\boldsymbol{y}_j - [\mathcal{A}(\boldsymbol{x}_0)]_j)^2}{2[\mathcal{A}(\boldsymbol{x}_0)]_j}\right)$$

$$p(y_j|x_0) = \frac{[A(x_0)]_j^{y_j} \exp(-[A(x_0)]_j)}{y_j!}$$

$$\simeq \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi \boldsymbol{y}_j}} \exp\left(-\frac{(\boldsymbol{y}_j - [\mathcal{A}(\boldsymbol{x}_0)]_j)^2}{2\boldsymbol{y}_j}\right),$$

- **Reverse SDE**

$$d\boldsymbol{x} = \left[-\frac{\beta(t)}{2}\boldsymbol{x} - \beta(t)(\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y}|\boldsymbol{x}_t))\right] dt + \sqrt{\beta(t)} d\bar{\boldsymbol{w}},$$

$$\theta^* = \arg\min_\theta \mathbb{E}_{t \sim U(\varepsilon,1), \boldsymbol{x}(t) \sim p(\boldsymbol{x}(t)|\boldsymbol{x}(0)), \boldsymbol{x}(0) \sim p_{\text{data}}} \left[\|\boldsymbol{s}_\theta(\boldsymbol{x}(t), t) - \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}(t)|\boldsymbol{x}(0))\|_2^2\right],$$

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y}) \simeq s_{\theta^*}(\boldsymbol{x}_t, t) - \rho \nabla_{\boldsymbol{x}_t} \|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0)\|_2^2$$

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y}) \simeq s_{\theta^*}(\boldsymbol{x}_t, t) - \rho \nabla_{\boldsymbol{x}_t} \|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0)\|_{\boldsymbol{\Lambda}}^2$$

算法流程

Algorithm 1 DPS - Gaussian

Require: $N, \boldsymbol{y}, \{\zeta_i\}_{i=1}^{N}, \{\tilde{\sigma}_i\}_{i=1}^{N}$

$\boldsymbol{x}_N \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
for $i = N-1$ **to** 0 **do**
    $\hat{\boldsymbol{s}} \leftarrow \boldsymbol{s}_\theta(\boldsymbol{x}_i, i)$
    $\hat{\boldsymbol{x}}_0 \leftarrow \frac{1}{\sqrt{\bar{\alpha}_i}}(\boldsymbol{x}_i + (1-\bar{\alpha}_i)\hat{\boldsymbol{s}})$
    $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
    $\boldsymbol{x}'_{i-1} \leftarrow \frac{\sqrt{\alpha_i}(1-\bar{\alpha}_{i-1})}{1-\bar{\alpha}_i}\boldsymbol{x}_i + \frac{\sqrt{\bar{\alpha}_{i-1}}\beta_i}{1-\bar{\alpha}_i}\hat{\boldsymbol{x}}_0 + \tilde{\sigma}_i \boldsymbol{z}$
    $\boldsymbol{x}_{i-1} \leftarrow \boldsymbol{x}'_{i-1} - \zeta_i \nabla_{\boldsymbol{x}_i} \|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0)\|_2^2$
end for
return $\hat{\mathbf{x}}_0$

$$d\boldsymbol{x} = \left[-\frac{\beta(t)}{2}\boldsymbol{x} - \beta(t)(\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{y}|\boldsymbol{x}_t))\right]dt + \sqrt{\beta(t)}d\bar{\boldsymbol{w}},$$

$$\hat{\boldsymbol{x}}_0 := \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t] = \frac{1}{\sqrt{\bar{\alpha}(t)}}(\boldsymbol{x}_t + (1-\bar{\alpha}(t))\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t))$$

$$\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t|\boldsymbol{y}) \simeq s_{\theta^*}(\boldsymbol{x}_t, t) - \rho\nabla_{\boldsymbol{x}_t}\|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0)\|_2^2$$

$$\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t|\boldsymbol{y}) \simeq s_{\theta^*}(\boldsymbol{x}_t, t) - \rho\nabla_{\boldsymbol{x}_t}\|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0)\|_{\boldsymbol{\Lambda}}^2$$

- **DDPM**

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathbf{I}),$$

where $\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t$ and $\tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$

DPS_从上述推导总结好处

1. 噪声 $\mathcal{J} \leq \frac{d}{\sqrt{2\pi\sigma^2}} e^{-1/2\sigma^2} \|\nabla_{\boldsymbol{x}} \mathcal{A}(\boldsymbol{x})\| m_1$

J接近 0 as σ → ∞,

2. 适用于非线性问题：

$$\mathcal{A}(\boldsymbol{x}) \triangleq \boldsymbol{A}\boldsymbol{x} \longrightarrow \mathcal{A}(\cdot)$$

3. 无需重新训练模型（只需扩散模型）

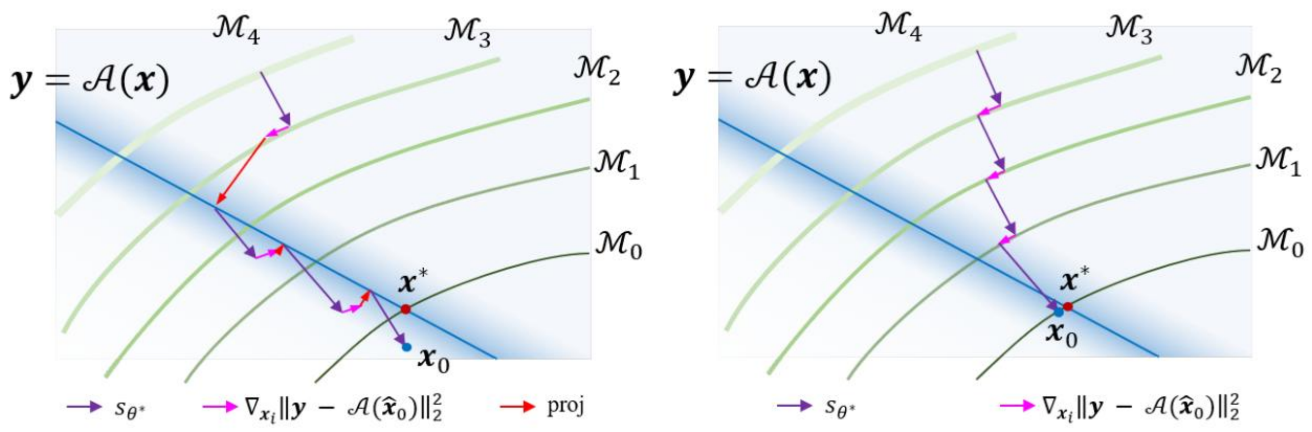
$$p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq p(\boldsymbol{y}|\hat{\boldsymbol{x}}_0)$$

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y}) \simeq s_{\theta^*}(\boldsymbol{x}_t, t) - \rho \nabla_{\boldsymbol{x}_t} \|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0)\|_2^2$$

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y}) \simeq s_{\theta^*}(\boldsymbol{x}_t, t) - \rho \nabla_{\boldsymbol{x}_t} \|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0)\|_{\boldsymbol{\Lambda}}^2$$

$$\hat{\boldsymbol{x}}_0 := \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t] = \frac{1}{\sqrt{\bar{\alpha}(t)}}(\boldsymbol{x}_t + (1 - \bar{\alpha}(t))\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t))$$

可视化梯度传播结果



(a) Geometry of Chung et al. (2022a)

(b) Geometry of DPS

MCG：投影到观测空间，过度强调数据的一致性，导致网络只能处理无噪声的情况。

DPS没有投影到观测空间

DPS的实验过程及结果

一、Noisy Linear inverse problem

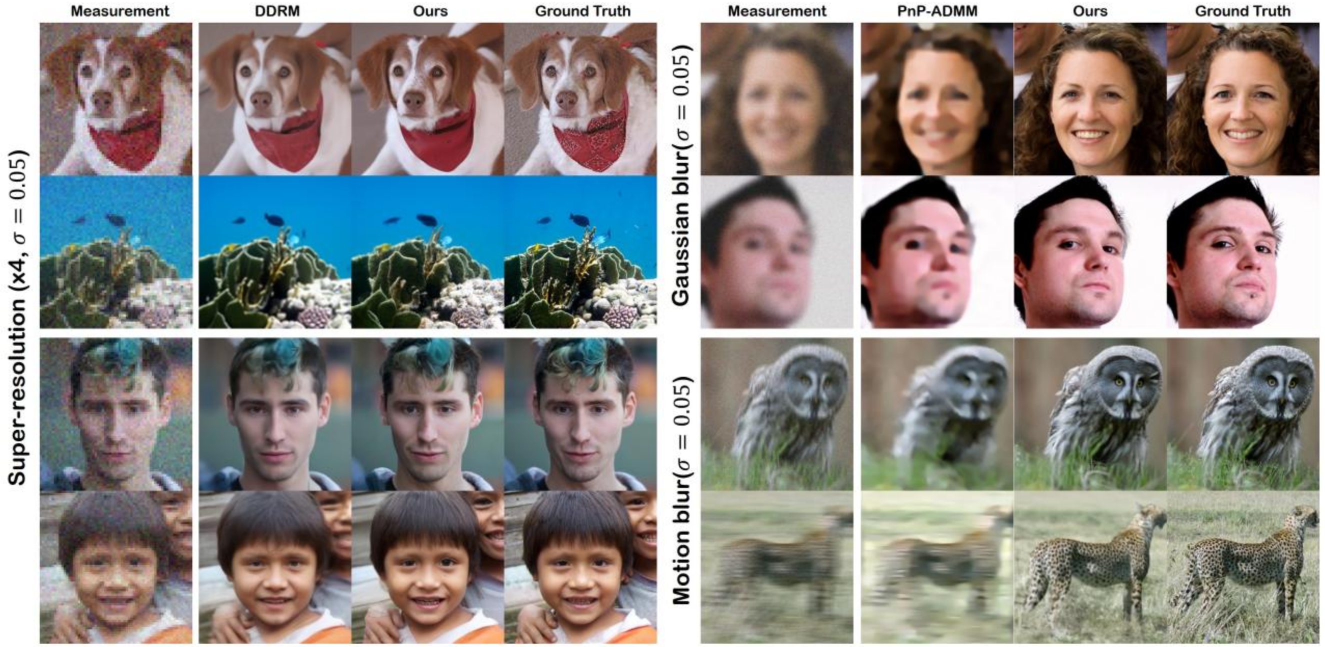


Figure 4: Results on solving linear inverse problems with Gaussian noise ($\sigma = 0.05$).

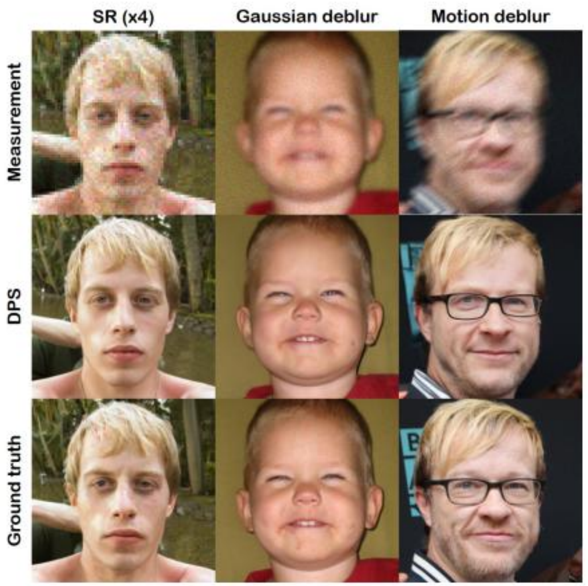


Figure 5: Results on solving linear inverse problems with Poisson noise ($\lambda = 1.0$)

实验方法：

数据集：FFHQ、ImageNet

1.**方框类型修复**（Box-type inpainting）

2.**超分辨率**（Super-resolution）

3.**高斯模糊**

4.**运动模糊**（Motion Blur）

5.**相位恢复**（Phase Retrieval）

二、Nonlinear inverse problem

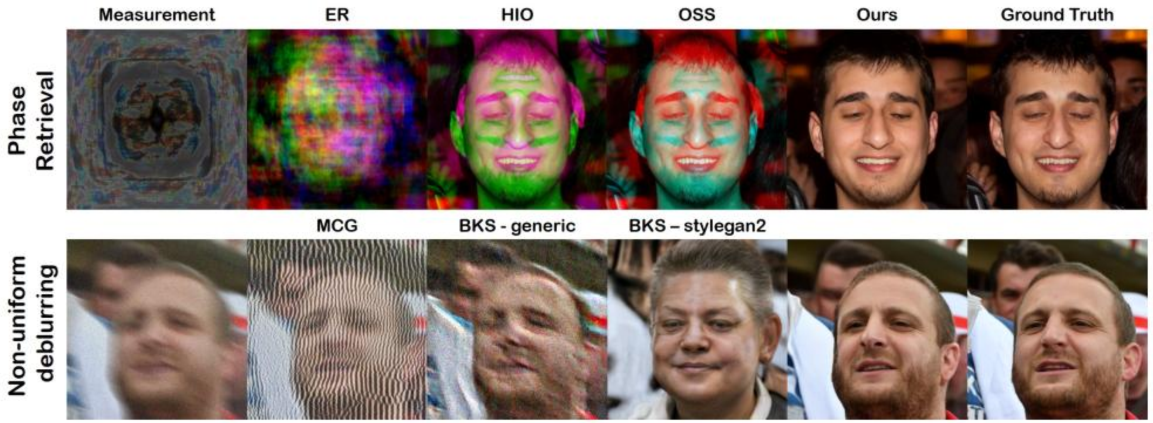


Figure 6: Results on solving nonlinear inverse problems with Gaussian noise ($\sigma = 0.05$).

FFHQ-linear inverse problem

Method	SR (×4)		Inpaint (box)		Inpaint (random)		Deblur (gauss)		Deblur (motion)	
	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓
DPS (ours)	**39.35**	**0.214**	**33.12**	**0.168**	**21.19**	**0.212**	**44.05**	**0.257**	**39.92**	**0.242**
DDRM (Kawar et al., 2022)	<u>62.15</u>	<u>0.294</u>	42.93	<u>0.204</u>	69.71	0.587	<u>74.92</u>	<u>0.332</u>	-	-
MCG (Chung [illegible]	87.64	0.520	<u>40.11</u>	0.309	<u>29.26</u>	<u>0.286</u>	101.2	0.340	310.5	0.702
PnP-ADMM (C[illegible]	[illegible]	0.353	151.9	0.406	123.6	0.692	90.42	0.441	<u>89.08</u>	<u>0.405</u>
Score-SDE (Song et al., 2021b) (ILVR (Choi et al., 2021))	96.72	0.563	60.06	0.331	76.54	0.612	109.0	0.403	292.2	0.657
ADMM-TV	110.6	0.428	68.94	0.322	181.5	0.463	186.7	0.507	152.3	0.508

Table 1: Quantitative evaluation (FID, LPIPS) of solving linear inverse problems on FFHQ 256×256-1k validation dataset. **Bold**: best, <u>underline</u>: second best.

ImageNet-linear inverse problem

Method	SR (×4)		Inpaint (box)		Inpaint (random)		Deblur (gauss)		Deblur (motion)	
	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓
DPS (ours)	**50.66**	**0.337**	**38.82**	<u>0.262</u>	**35.87**	**0.303**	**62.72**	<u>0.444</u>	**56.08**	**0.389**
DDRM (Kawar et al., 2022)	<u>59.57</u>	<u>0.339</u>	45.95	**0.245**	114.9	0.665	<u>63.02</u>	**0.427**	-	-
MCG (Chung et al., 2022a)	144.5	0.637	<u>39.74</u>	0.330	<u>39.19</u>	<u>0.414</u>	95.04	0.550	186.9	0.758
PnP-ADMM (Chan et al., 2016)	97.27	0.433	78.24	0.367	114.7	0.677	100.6	0.519	<u>89.76</u>	<u>0.483</u>
Score-SDE (Song et al., 2021b) (ILVR (Choi et al., 2021))	170.7	0.701	54.07	0.354	127.1	0.659	120.3	0.667	98.25	0.591
ADMM-TV	130.9	0.523	87.69	0.319	189.3	0.510	155.7	0.588	138.8	0.525

Table 2: Quantitative evaluation (FID, LPIPS) of solving linear inverse problems on ImageNet 256×256-1k validation dataset. **Bold**: best, <u>underline</u>: second best.

$$\mathrm{FID}(x,y) = \|\mu_x - \mu_y\|^2 + \mathrm{Tr}(\Sigma_x + \Sigma_y - 2(\Sigma_x\Sigma_y)^{1/2})$$

$$\mathrm{LPIPS}(x,y) = \sum_k w_k \|\phi_k(x) - \phi_k(y)\|^2$$

Phase Retrieval

Method	FID ↓	LPIPS ↓
DPS(ours)	**55.61**	**0.399**
OSS	137.7	0.635
HIO	96.40	0.542
ER	214.1	0.738

Table 3: Quantitative evaluation of the Phase Retrieval task (FFHQ).

Nonlinear deblurring

Method	FID ↓	LPIPS ↓
DPS(ours)	**41.86**	0.278
BKS-styleGAN2	63.18	0.407
BKS-generic	141.0	0.640
MCG	180.1	0.695

Table 4: Quantitative evaluation of the non-uniform deblurring task (FFHQ).

核心创新点

$$p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq p(\boldsymbol{y}|\hat{\boldsymbol{x}}_0), \quad \text{where} \quad \hat{\boldsymbol{x}}_0 := \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t] = \mathbb{E}_{\boldsymbol{x}_0 \sim p(\boldsymbol{x}_0|\boldsymbol{x}_t)}[\boldsymbol{x}_0]$$

- 优点：

不用**SVD**解决噪声、有效处理非线性问题

$$\boldsymbol{Y} = \mathcal{A}(\boldsymbol{X}) + \sigma_{\boldsymbol{y}}\boldsymbol{Z}, \ \boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{0}, I_m).$$

A需已知