Sony AI

GibbsDDRM: A Partially Collapsed Gibbs Sampler for Solving Blind Inverse Problems with Denoising Diffusion Restoration

Naoki Murata[1], Koichi Saito[1], Chieh-Hsin Lai[1], Yuhta Takida[1], Toshimitsu Uesaka[1], Yuki Mitsufuji[1,2], Stefano Ermon[3]

[1]Sony AI, [2]Sony Group Corporation, [3]Stanford University

Overview of GibbsDDRM

Unsupervised approach to blind linear inverse problems utilizing a pre-trained diffusion model

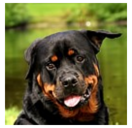
Problem: Posterior sampling in blind linear inverse problems.

Measurement model

$$\mathbf{y} = \mathbf{H}_\varphi * \mathbf{x}_0 + \mathbf{z}$$

$\mathbf{H}_\varphi$, $\mathbf{x}_0$ → **Unknown**

Sampling from the joint posterior

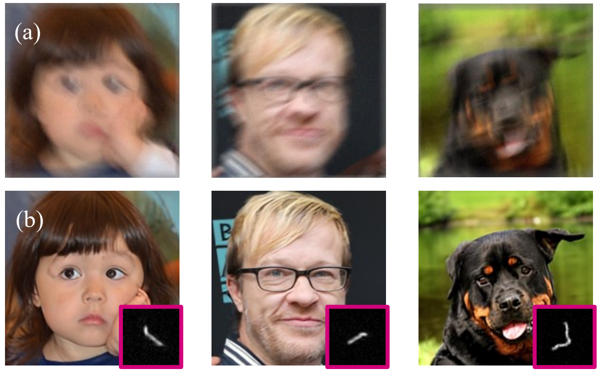
( , ) $\sim p(\mathbf{x}_0, \varphi | \mathbf{y})$

Approach: Efficient posterior sampling using a variant of the Gibbs sampler

The proposed Gibbs sampler consists of tractable conditional distributions that enable sampling using a pre-trained diffusion model.

Advantage and results:

- ☞ Problem-agnostic: A pretrained diffusion model can be applied to different inverse problems without requiring fine-tuning.
- ☞ Success even when simple generic priors (e.g., Laplacian prior) are used for the linear operators.



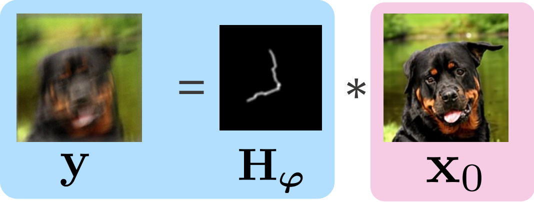
N. Murata, “GibbsDDRM: A Partially Collapsed Gibbs Sampler for Solving Blind Inverse Problems with Denoising Diffusion Restoration”

Diffusion models for inverse problems

Methods that employ rich generative models as priors to solve inverse problems

Problem setting: (non-blind) Inverse problems

Measurement Degradation Clean data

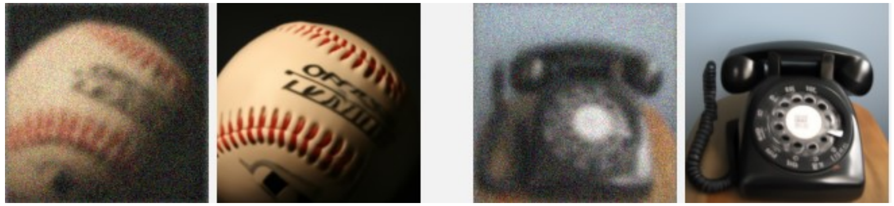


$\mathbf{y} = \mathbf{H}_{\varphi} * \mathbf{x}_0 + \mathbf{z}$

➡ **Known** ➡ **Unknown**

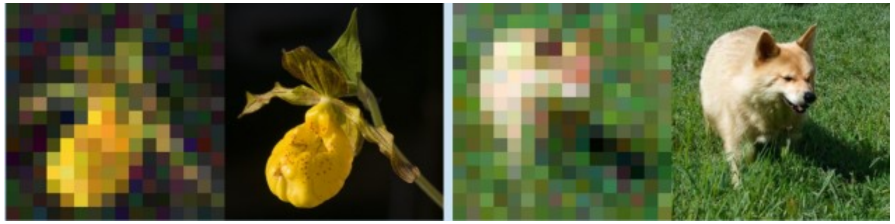
- ☞ Inverse problems are ill-posed because of the degradation process.
- ☞ An assumption on prior data is necessary to narrow down plausible solutions.
- ☞ Recent methods utilize **pre-trained diffusion models** as data priors.

Denoising Diffusion Restoration Models (DDRM) [Kawar, NeurIPS2022]



Deblurring

Diffusion Posterior Sampling (DPS) [Chung, ICLR2023]



Super resolution

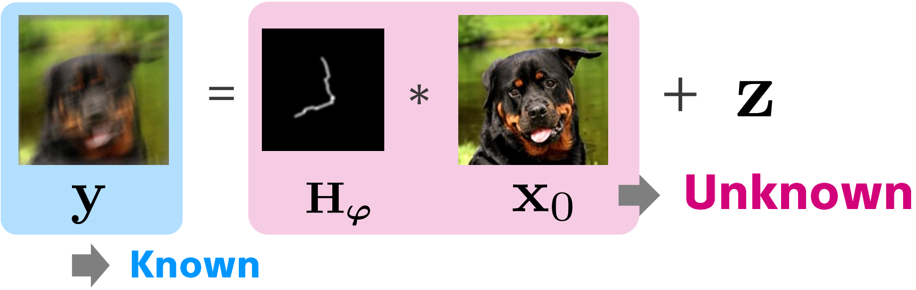
Both methods assume that corruption processes are known.

☞ **We consider a blind setting, where the degradation process is also unknown.**

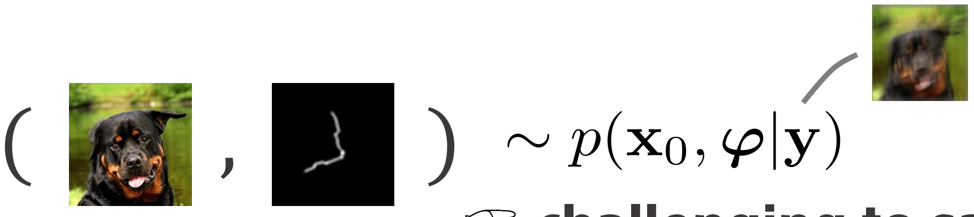
Problem setting and sampling from the posterior

Problem: Posterior sampling in blind linear inverse problems

Blind linear inverse problems

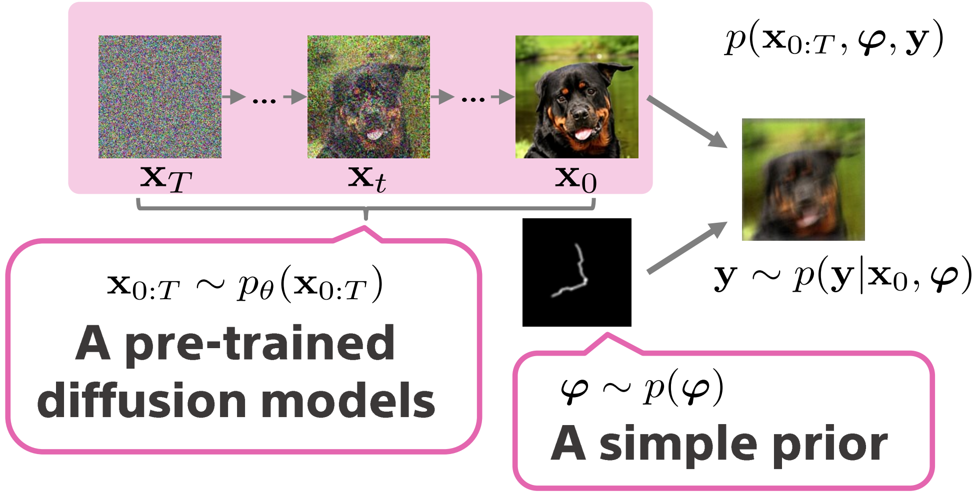


Task: Posterior sampling from the joint distribution

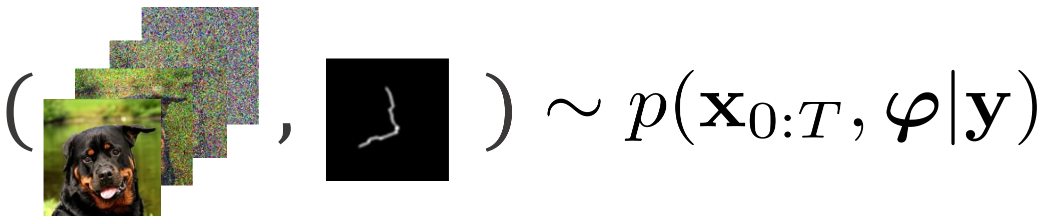


☞ **challenging to sample**

Joint distribution for the data, latents, measurements, and linear operator's parameter



Posterior sampling from the joint distribution



☞ **Can be sampled with a variant of the Gibbs sampler**

GibbsDDRM: A Partially Collapsed Gibbs Sampler

Our objective : sampling from the joint posterior

( , ) $\sim p(\mathbf{x}_{0:T}, \boldsymbol{\varphi} | \mathbf{y})$

Naïve Gibbs Sampler? ☞ it is difficult to obtain all the conditional distribution for the sampler.

Naïve Blocked Gibbs sampler? ☞ It will take long inference time.

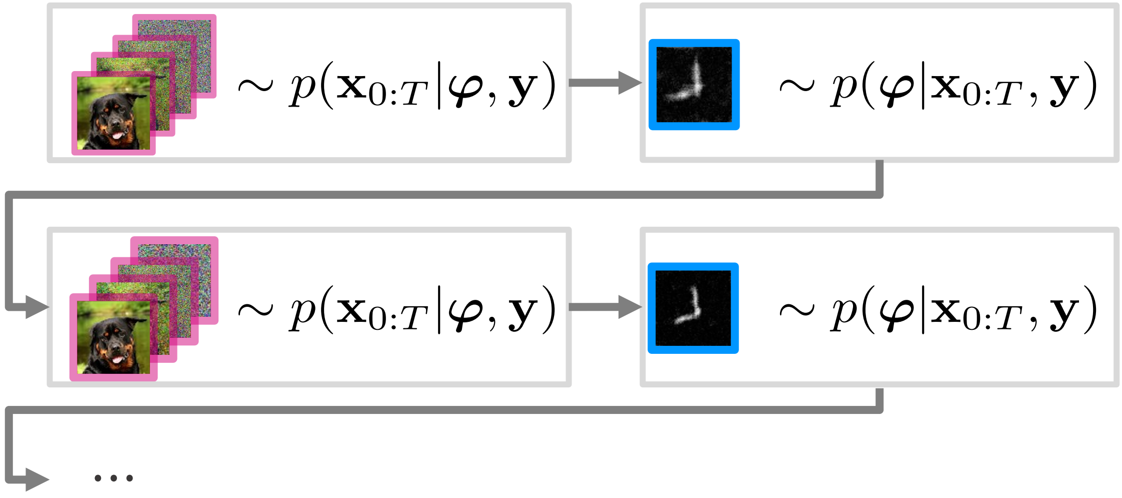
Our approach: A partially collapsed Gibbs sampler (PCGS) [Van Dyk & Park, 2008]

- A **computationally feasible** Gibbs sampler is obtained that has the approximated target posterior as its stationary distribution.
- If no approximation is made in each sampling step, the proposed sampler achieves sampling from the true posterior. (Prop. 3.1.)

Naïve Gibbs sampler and Proposed PCGS

Naïve Gibbs Sampler? 👉it is difficult to obtain all the conditional distribution for the sampler.
Naïve Blocked Gibbs sampler?👉It will take long inference time.

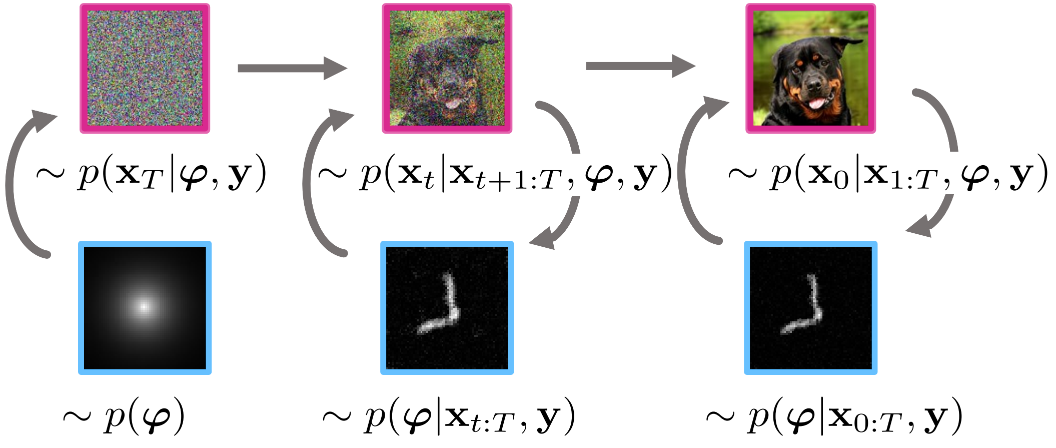
Naïve Blocked Gibbs sampler



To sample $\boldsymbol{\varphi}$ once, the entire diffusion model's sampling for obtaining $\mathbf{x}_{0:T}$ is necessary.

Our approach

A partially collapsed Gibbs sampler (PCGS)



The data $\mathbf{x}_0$, latent variables $\mathbf{x}_{1:T}$,and linear operator's parameter $\boldsymbol{\varphi}$ can be sampled **within a cycle of diffusion-based sampling**.

N. Murata, "GibbsDDRM: A Partially Collapsed Gibbs Sampler for Solving Blind Inverse Problems with Denoising Diffusion Restoration"

GibbsDDRM: Sampling of $\mathbf{x}_t$ (Data and latent variables)

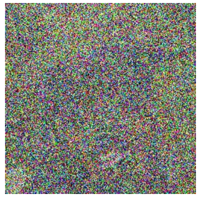
Algorithm of GibbsDDRM

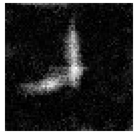
for $n = 1$ **to** N **do**
  $\varphi^{(n,0)} \leftarrow \varphi^{(n-1,K)}, K \leftarrow 0$
  Sample $\mathbf{x}_T^{(n,0)} \sim p(\mathbf{x}_T|\varphi^{(n,K)}, \mathbf{y})$
  // $\uparrow$ approximated by $p_\theta(\mathbf{x}_T|\varphi, \mathbf{y})$.
  for $t = T-1$ **to** 0 **do**
    $\chi_t \leftarrow \{\mathbf{x}_{t+1}^{(n,M_{t+1})}, \mathbf{x}_{t+2}^{(n,M_{t+2})}, \cdots, \mathbf{x}_T^{(n,0)}\}$
    Sample $\mathbf{x}_t^{(n,0)} \sim p(\mathbf{x}_t|\varphi^{(n,K)}, \chi_t, \mathbf{y})$
    // $\uparrow$ approximated by $p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1}, \varphi, \mathbf{y})$.
    for $m = 1$ **to** M_t **do**
      Sample $\varphi^{(n,K+1)} \sim p(\varphi|\mathbf{x}_t^{(n,m-1)}, \chi_t, \mathbf{y})$
      // $\uparrow$ Langevin sampling with the approximated score $\nabla_\varphi \log p(\mathbf{y}|\mathbf{x}_{\theta,t}, \varphi)$.
      $K \leftarrow K + 1$
      Sample $\mathbf{x}_t^{(n,m)} \sim p(\mathbf{x}_t|\varphi^{(n,K)}, \chi_t, \mathbf{y})$
      // $\uparrow$ approximated by $p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1}, \varphi, \mathbf{y})$.
    end for
  end for
end for

Sampling of $\mathbf{x}_t$ (Data and latent variables)

Approximated by the DDRM [Kawar, NeurIPS2022], which is diffusion-based method for inverse problems.

$$\mathbf{x}_t \sim p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1}, \varphi, \mathbf{y})$$





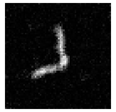
GibbsDDRM: Sampling of φ (Linear operator's parameter)

Algorithm of GibbsDDRM

for $n = 1$ **to** N **do**
  $\varphi^{(n,0)} \leftarrow \varphi^{(n-1,K)}$, $K \leftarrow 0$
  Sample $\mathbf{x}_T^{(n,0)} \sim p(\mathbf{x}_T | \varphi^{(n,K)}, \mathbf{y})$
  // ↑ approximated by $p_\theta(\mathbf{x}_T | \varphi, \mathbf{y})$.
  for $t = T-1$ **to** 0 **do**
    $\chi_t \leftarrow \{\mathbf{x}_{t+1}^{(n,M_{t+1})}, \mathbf{x}_{t+2}^{(n,M_{t+2})}, \cdots, \mathbf{x}_T^{(n,0)}\}$
    Sample $\mathbf{x}_t^{(n,0)} \sim p(\mathbf{x}_t | \varphi^{(n,K)}, \chi_t, \mathbf{y})$
    // ↑ approximated by $p_\theta(\mathbf{x}_t | \mathbf{x}_{t+1}, \varphi, \mathbf{y})$.
    for $m = 1$ **to** M_t **do**
      Sample $\varphi^{(n,K+1)} \sim p(\varphi | \mathbf{x}_t^{(n,m-1)}, \chi_t, \mathbf{y})$
      // ↑ Langevin sampling with the approximated score $\nabla_\varphi \log p(\mathbf{y} | \mathbf{x}_{\theta,t}, \varphi)$.
      $K \leftarrow K + 1$
      Sample $\mathbf{x}_t^{(n,m)} \sim p(\mathbf{x}_t | \varphi^{(n,K)}, \chi_t, \mathbf{y})$
      // ↑ approximated by $p_\theta(\mathbf{x}_t | \mathbf{x}_{t+1}, \varphi, \mathbf{y})$.
    end for
  end for
end for

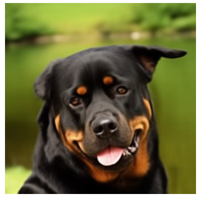
Sampling of φ (Linear operator's parameter)

Langevin sampling with the approximated score



$$\nabla_\varphi \log p(\varphi | \mathbf{x}_{t:T}, \mathbf{y}) \simeq -\frac{1}{2\sigma_\mathbf{y}^2} \nabla_\varphi \|\mathbf{y} - \mathbf{H}_\varphi \mathbf{x}_{\theta,t}\|_2^2 + \nabla_\varphi \log p(\varphi)$$

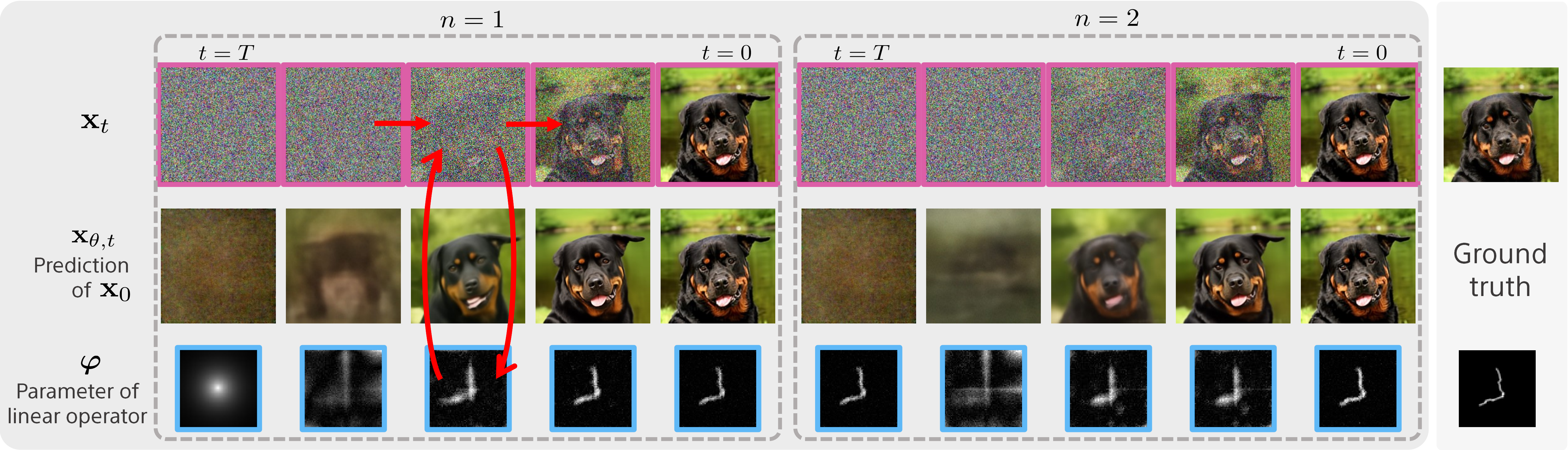
Theorem 3.2.
$p(\mathbf{y} | \mathbf{x}_{t:T}, \varphi) \simeq p(\mathbf{y} | \mathbf{x}_{\theta,t}, \varphi)$



Simple generic prior

Estimated clean data from $\mathbf{x}_t$

Experimental results: Blind image deblurring



Sony AI

Blind image deblurring results

Quantitative results

Method	FFHQ (256 × 256) FID↓	LPIPS ↓	PSNR ↑
GibbsDDRM (ours)	38.71	**0.115**	25.80
MPRNet (Zamir et al., 2021)	62.92	0.211	**27.23**
DeblurGANv2 (Kupyn et al., 2019)	141.55	0.320	19.86
Pan-DCP (Pan et al., 2017)	239.69	0.653	14.20
SelfDeblur (Ren et al., 2020)	283.69	0.859	10.44
BlindDPS (Chung et al., 2023a)*	**29.49**	0.281	22.24
DDRM (Kawar et al., 2022) with GT kernel	33.97	0.062	30.64

Faithfulness

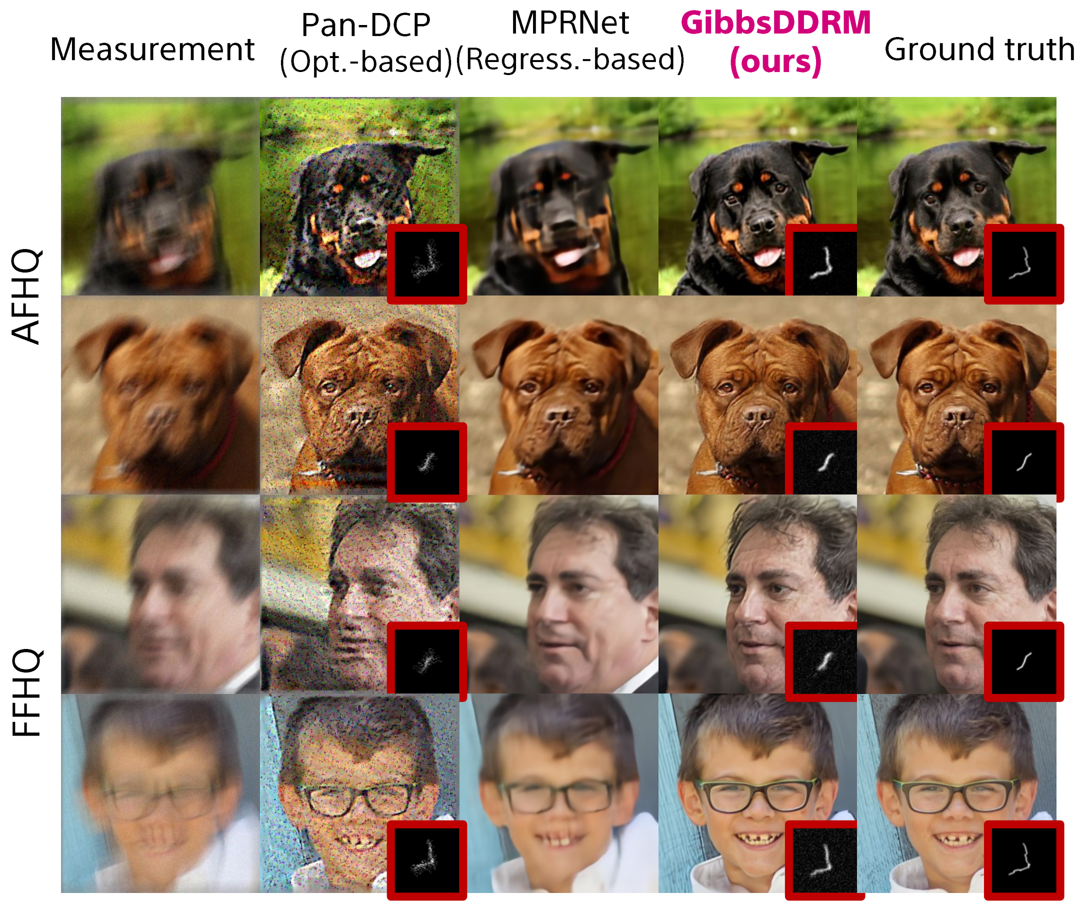
☞ **GibbsDDRM achieved the lowest LPIPS score, while also showing a lower FID score.**

Image quality

☞ The images estimated by GibbsDDRM are perceptually similar to the ground truth.

☞ GibbsDDRM utilizes a generative model to restore lost components.

Qualitative results



Quantitative results: Vocal dereverberation

Table 2. Vocal dereverberation results. **Bold**: Best.

Method	FAD ↓	SI-SDR ↑ improvement	SRMR ↑
Wet (unprocessed)	5.74	–	7.11
Reverb Conversion (Koo et al., 2021)	5.69	0.02	7.23
Music Enhancement (Kandpal et al., 2022)	7.51	−23.9	7.92
Unsupervised Dereverberation(Saito et al., 2023)	4.99	0.37	7.94
GibbsDDRM	**4.21**	**0.59**	**8.40**

- ☞ The task: restore the original dry vocal from a noisy, reverberant (wet) vocal.
- ☞ **GibbsDDRM achieves the highest speech-to-reverberation modulation energy ratio (SRMR), while showing the lowest FAD.**

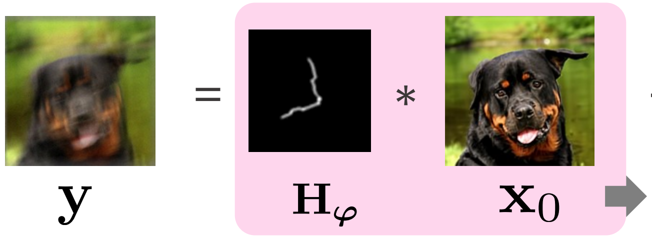
Please listen to the restored samples ☞

Summary of GibbsDDRM

Unsupervised approach to blind linear inverse problems utilizing a pre-trained diffusion model

Problem: Posterior sampling in blind linear inverse problems.

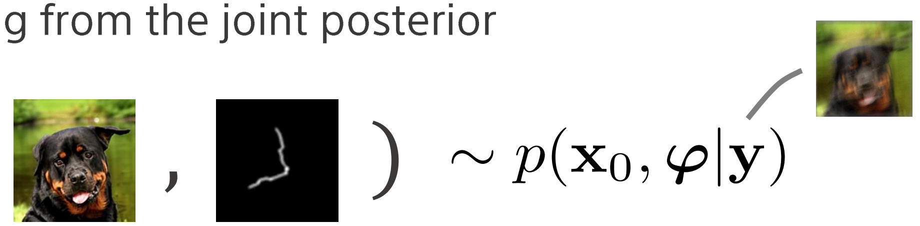
Measurement model



$$\mathbf{y} = \mathbf{H}_\varphi * \mathbf{x}_0 + \mathbf{z}$$

$\mathbf{H}_\varphi$, $\mathbf{x}_0$ → **Unknown**

Sampling from the joint posterior



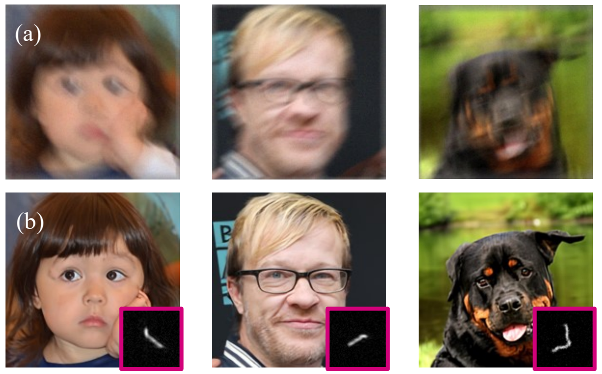
$$(\cdot , \cdot) \sim p(\mathbf{x}_0, \varphi | \mathbf{y})$$

Approach: Efficient posterior sampling using a variant of the Gibbs sampler

The proposed Gibbs sampler consists of tractable conditional distributions that enable sampling using a pre-trained diffusion model.

Advantage and results:

- ☞ Problem-agnostic: A pretrained diffusion model can be applied to different inverse problems without requiring fine-tuning.
- ☞ Success even when simple generic priors (e.g., Laplacian prior) are used for the linear operators.



N. Murata, "GibbsDDRM: A Partially Collapsed Gibbs Sampler for Solving Blind Inverse Problems with Denoising Diffusion Restoration"

SONY