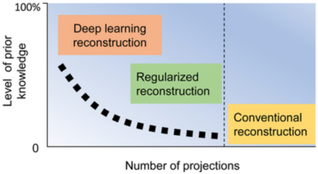
Solving Inverse Problems with Latent Diffusion Models via Hard Data Consistency

Bowen Song, Soo Min Kwon, Zecheng Zhang, Xinyu Hu,
Qing Qu, Liyue Shen

April 20, 2024

Deep Learning for Inverse Problems



Learn the data-driven prior by deep learning when the measurement is sparse [6] [1]

- Inverse problems arise from a wide range of applications across many domains, including computational imaging, remote sensing, and so on.
- The goal is to reconstruct an unknown signal x_{true} given the observed measurements y of the form $y = A(x_{true}) + \epsilon$, where ϵ can be an additive noise.
- Deep learning models that learn the data prior (distribution of $p(x_{clean,y})$) help reconstruct the clean images from very **sparse** measurements.

[1] Shen, Liyue et al., Patient-specific reconstruction of volumetric computed tomography images from a single projection view via deep learning, Nature biomedical engineering

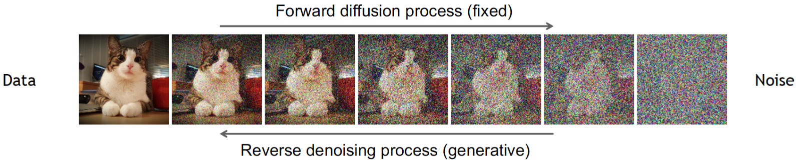
Deep Learning for Inverse Problems

- **Supervised** Approaches (assuming that the x_{clean}, y pair is available during training, train a network that maps y to x_{clean}) [8]
  - Need to retrain for a different inverse problem
  - Generalization capabilities may be limited in the presence of noise/modality shift [7]
  - Need paired data for training
- **Unsupervised** Approaches (assuming that only x_{clean} is available during training) [8]
  - Easily adapt to a new inverse problem in a zero-shot manner.
  - Do not need paired data for training.

Both approaches are widely reported in the literature [8].

Diffusion (Score-based) Models

Denoising diffusion models consist of two processes



An illustration of the diffusion pipeline [9]

- A forward process in which gradually add noise to x_{clean}
- A reverse denoising process that remove noise from x_t to recover x_{clean}

Specifically, the reverse process is governed by the score function $\nabla \log p(x_t)$. Training a neural network that approximates $\nabla \log p(x_t)$ would enable data generation capability.

Mathematical Formulation of Diffusion Process

- As $x_t \approx x_{t-1} - \frac{\beta_t \Delta t}{2} x_{t-1} + \sqrt{\beta_t} \Delta t \omega$ where $\omega \in N(0,1)$
- As $\Delta t \to 0$, then $dx_t = -\frac{1}{2}\beta_t x_t dt + \sqrt{\beta_t} d\omega_t$

Forward Diffusion SDE:

$$\mathrm{d}\mathbf{x}_t = -\frac{1}{2}\beta(t)\mathbf{x}_t\,\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\boldsymbol{\omega}_t$$

Reverse Generative Diffusion SDE:

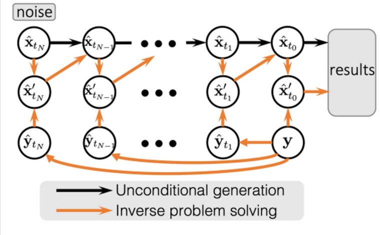
$$\mathrm{d}\mathbf{x}_t = \overbrace{\left[-\frac{1}{2}\beta(t)\mathbf{x}_t - \beta(t)\underbrace{\nabla_{\mathbf{x}_t}\log q_t(\mathbf{x}_t)}_{\text{“Score Function”}}\right]}^{\text{drift term}}\mathrm{d}t + \overbrace{\sqrt{\beta(t)}\,\mathrm{d}\bar{\boldsymbol{\omega}}_t}^{\text{diffusion term}}$$

The mathematical formulation of diffusion process [1]

- The solution of the stochastic differential equation can be utilized by the score function
- we can use a neural network to approximate it, such as $s_\theta(x_t) \approx \nabla_{x_t} \log p(x_t)$

Solving Inverse Problem with Diffusion Models

Song, Yang & Shen, Liyue 2022 [4]

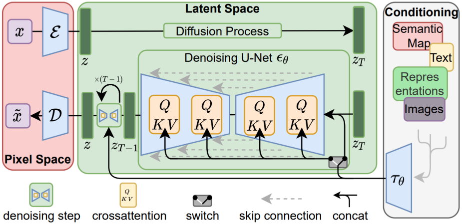


the flowchart of Score SDE [4]. Each $\hat{\mathbf{x}}_t$ is modified through optimization

To solve linear inverse problems with diffusion model priors, we can use

- **hard consistency**: modify $\mathbf{x}_t$ with optimization, such as with the objective $\text{argmin}_z \lambda ||\hat{\mathbf{x}}_t - z||_2^2 + (1-\lambda)||\hat{\mathbf{y}}_t - Az||_2^2$ [4, 3]
- **soft consistency**: change $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)$ to $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|y)$ via Bayesian rule [1, 5]. We have $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|y) = \nabla_{\mathbf{x}_t} \log p(y|\mathbf{x}_t) + \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)$, with $\nabla_{\mathbf{x}_t} \log p(y|\mathbf{x}_t)$ can be approximated through $\nabla_{\mathbf{x}_t} \log p(y|\hat{\mathbf{x}}_0(\mathbf{x}_t))$ which is the score of the likelihood of the predicted ground truth image[1].

Training diffusion models to model $p(x_{clean})$ can be costly, we can save training time and memory usage by training diffusion models in the latent space [3, 10], while enabling conditioning on multimodal inputs.



the flowchart of Latent Diffusion Models [2]

Brute-Force Approaches

- Learn the conditional score function with the loss $\min ||\epsilon - \epsilon_\theta(z_t, t, y)||$. But this needs to retrain for each different forward functions [2].
- Train another score function to model the diffusion of y_t, this also needs to retrain for each different measurement functions [3].
- Use DPS formulation [1] as $\nabla_{\boldsymbol{z}_t} \log p(\boldsymbol{z}_t|y) \approx \nabla_{\boldsymbol{z}_t} \log p(y|\hat{\boldsymbol{z}}_{\mathbf{0}}(\boldsymbol{z}_t)) + \nabla_{\boldsymbol{z}_t} \log p(\boldsymbol{x}_t)$, where $\nabla_{\hat{\boldsymbol{z}}_{\mathbf{0}}(\boldsymbol{z}_t)} \log p(y|\hat{\boldsymbol{z}}_{\mathbf{0}}(\boldsymbol{z}_t)) \propto ||y - A(D(z))||_2^2$

Challenges in Solving Inverse Problems with Latent Diffusion Models

- the forward model with latent diffusion model is given by $A(D(.))$, which is a highly non-convex and non-linear operator (a deep neural network) [10]
- Soft-consistency methods fail to have measurement consistency and generate blurry or noisy results [10]
- Most hard-consistency methods can only handle linear inverse problems [10]. Many need a new diffusion sequence y_t such that $y_t = Ax + \epsilon_t$ [4], but $y_t|y$ becomes intractable when A is nonlinear.

Our Approach

Key Observation

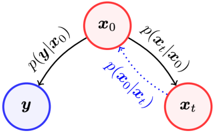


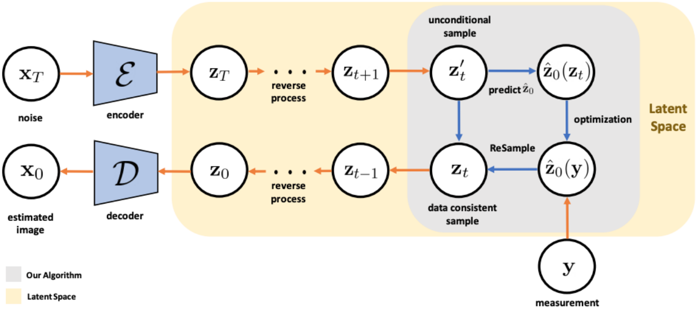
Figure 2: Probabilistic graph. Black solid line: tractable, blue dotted line: intractable in general.

Probablistic Graphical Model of conditional forward diffusion

The conditional forward process has $p(z_t|z_0, y) = p(z_t|z_0)$. Also $p(z_t|z_{t-1}, y) = p(z_t|z_{t-1})$. So if we can construct an estimation of z_0 call it $\hat{z}_0(y)$ that is consistent to the measurement, we can then sample $z_{t'}$ from $p(z_t|\hat{z}_0(y))$, so that $z_{t'}$ encodes information from y.

An Overview of Our Approach

The entire sampling process is conducted in the latent space upon starting from a random noise. The proposed algorithm predict the $\hat{\mathbf{z}}_0(\mathbf{z}_t)$ at $t = 0$, and then performs hard data consistency optimization at some time steps t via a skipped-step mechanism. ReSample is performed afterwards to map $\hat{\mathbf{z}}_0(y)$ back to time t



An overview of our method [2]

[2] Song, Bowen and Kwon, Soo Min et al., Solving inverse problems with latent diffusion models via hard data consistency, International Conference on Learning Representations (ICLR). 2024 **Spotlight**

Algorithm PseudoCode

Algorithm 1 ReSample: Solving Inverse Problems with Latent Diffusion Models

Require: Measurements $\boldsymbol{y}$, $\mathcal{A}(\cdot)$, Encoder $\mathcal{E}(\cdot)$, Decoder $\mathcal{D}(\cdot)$, Score function $\boldsymbol{s}_\theta(\cdot, t)$, Pretrained LDM Parameters β_t, $\bar{\alpha}_t$, η, δ, Hyperparameter γ to control σ_t^2, Time steps to perform resample C

$\boldsymbol{z}_T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ ▷ Initial noise vector

for $t = T-1, \ldots, 0$ **do**

  $\boldsymbol{\epsilon}_1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$

  $\hat{\boldsymbol{\epsilon}}_{t+1} = \boldsymbol{s}_\theta(\boldsymbol{z}_{t+1}, t+1)$ ▷ Compute the score

  $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_{t+1}) = \frac{1}{\sqrt{\bar{\alpha}_{t+1}}}(\boldsymbol{z}_{t+1} - \sqrt{1-\bar{\alpha}_{t+1}}\hat{\boldsymbol{\epsilon}}_{t+1})$ ▷ Predict $\hat{\boldsymbol{z}}_0$ using Tweedie's formula

  $\boldsymbol{z}_t' = \sqrt{\bar{\alpha}_t}\hat{\boldsymbol{z}}_0(\boldsymbol{z}_{t+1}) + \sqrt{1-\bar{\alpha}_t - \eta\delta^2}\hat{\boldsymbol{\epsilon}}_{t+1} + \eta\delta\boldsymbol{\epsilon}_1$ ▷ Unconditional DDIM step

  if $t \in C$ **then** ▷ ReSample time step

    $\hat{\boldsymbol{z}}_0(\boldsymbol{y}) \in \arg\min_{\boldsymbol{z}} \frac{1}{2}\|\boldsymbol{y} - \mathcal{A}(\mathcal{D}(\boldsymbol{z}))\|_2^2$ ▷ Solve with initial point $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_{t+1})$

    $\boldsymbol{z}_t = \text{StochasticResample}(\hat{\boldsymbol{z}}_0(\boldsymbol{y}), \boldsymbol{z}_t', \gamma)$ ▷ Map back to t

  else

    $\boldsymbol{z}_t = \boldsymbol{z}_t'$ ▷ Unconditional sampling if not resampling

$\boldsymbol{x}_0 = \mathcal{D}(\boldsymbol{z}_0)$ ▷ Output reconstructed image

Key Idea 1

Basic ReSample: optimized at $t = 0$ and resample back

- Given an x_t, I first use Tweedie's formula to obtain $\hat{z}_0 = \mathbb{E}[z_0|z_t] = \frac{z_t - (1-\alpha_t)\epsilon_\theta(z_t, t)}{\sqrt{\alpha_t}}$
- Let D be the decoder, and E be the encoder. The closest point $\hat{z}_0(y)$ that is measurement-consistent can be approximated by $E((I - A^+A)D(\hat{z}_0) + A^+y)$ if lossless autoencoding, where A^+ can be psuedo-inverse for simplicity.
  - Easily derived through null space decomposition, I can demonstrate the proof if interested.
- Then I can sample from $p(z_t|\hat{z}_0(y))$ to get $z_{t'}$ to replace the original z_t[2].

Problem of this approach: Too much noise. Everytime $z_{t'}$ has a variance of $1 - \alpha_t$, which is significantly larger than the noise level for each step of diffusion sampling.

[2] Song, Bowen and Kwon, Soo Min et al., Solving inverse problems with latent diffusion models via hard data consistency, International Conference on Learning Representations (ICLR). 2024 **Spotlight**

Key Idea 2

Posterior ReSample + Skip Step

- For computational efficiency, we do not to resample for every t, but only perform resample once every N steps[2].
- To mitigate the large variance issue, I propose to add a prior to $z_{t'}$ to be centered at the given z_t, and use that $p(z_t|\hat{z}_0(y)) \in N(\sqrt{\alpha_t}\hat{z}_0(y), 1-\alpha)$ to get a less noisy $z_{t'}$.
  - Then, $z_{t'} \in N(\frac{\sigma^2\sqrt{\alpha_t}\hat{z}_0(y)+(1-\alpha_t)z_t}{\sigma^2+1-\alpha_t}, \frac{1}{\frac{1}{\sigma^2}+\frac{1}{1-\alpha_t}})$. It is a weighted average between original given x_t and the mean of resampled[2].

[2] Song, Bowen and Kwon, Soo Min et al., Solving inverse problems with latent diffusion models via hard data consistency, International Conference on Learning Representations (ICLR). 2024 **Spotlight**

Theoretical Analysis

- ReSample reduces variance compared to stochastic encoding[2]
- if $\hat{\mathbf{z}}_0(y)$ is consistent to measurement, such that $y = A(D(\hat{\mathbf{z}}_0(y)))$, then the expectation of the unconditional sample is equal to the expectation of the sample after stochastic resampling (**Unbiasness**)[2].
- The predicted z_0 converges to the ground truth z_0 in probability as t decreases assuming the second order score is bounded[2].

  More details about the theoretical analysis can be found in [10] and the appendix in this slide.

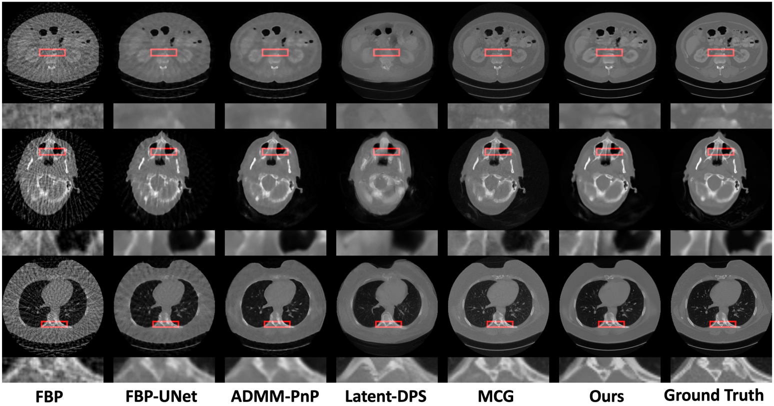
[2] Song, Bowen and Kwon, Soo Min et al., Solving inverse problems with latent diffusion models via hard data consistency, International Conference on Learning Representations (ICLR). 2024 **Spotlight**

Experimental Settings

- For linear inverse problems, we consider the following tasks:
  - Gaussian deblurring, we use a kernel with size 61×61 with standard deviation 3.0.
  - For super resolution, we use 4x bicubic downsampling
  - For inpainting, we use a random mask with varying levels of missing pixels.

  All images are $256 \times 256 \times 3$
- For nonlinear deblurring, we apply the kernel as proposed by Chung et al. [1].
- For CT reconstruction, we simulate CT measurements (sinograms) on 256×256 full-dose CT images with a parallel-beam geometry using 25 projection angles equally distributed across 180 degrees.

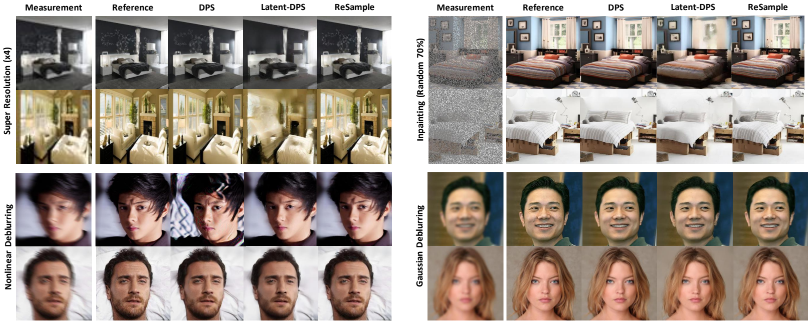
Result shows that ReSample gives sharp images and high quality reconstructions for both natural images and medical images.



Reconstruction on sparse (25) simulated CT parallel-beam projections

Results

Result shows that ReSample gives sharp images and high quality reconstructions for both natural images and medical images.



Reconstruction on various inverse problems, such as inpainting, deblurring and superresolution

Quantitative Results

ReSample achieves SOTA or comparable performance on a variety of inverse problems on natural images on a variety of datasets.

Method	Nonlinear Deblurring			Gaussian Deblurring		
	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑
DPS [11]	0.230 ± 0.065	$\underline{26.81} \pm 2.84$	$\underline{0.720} \pm 0.077$	0.175 ± 0.03	28.36 ± 2.12	0.772 ± 0.07
MCG [12]	-	-	-	0.517 ± 0.06	15.85 ± 1.08	0.536 ± 0.08
ADMM-PnP [29]	0.499 ± 0.073	16.17 ± 4.01	0.359 ± 0.140	0.289 ± 0.04	20.98 ± 4.51	0.602 ± 0.15
DDRM [13]	-	-	-	0.193 ± 0.04	26.88 ± 1.96	0.747 ± 0.07
DMPS [16]	-	-	-	0.206 ± 0.04	26.45 ± 1.83	0.726 ± 0.07
Latent-DPS	$\underline{0.225} \pm 0.04$	26.18 ± 1.73	0.703 ± 0.07	0.205 ± 0.04	27.42 ± 1.84	0.729 ± 0.07
PSLD [21]	-	-	-	0.360 ± 0.15	23.07 ± 3.91	0.494 ± 0.22
ReSample (Ours)	$\mathbf{0.153} \pm 0.03$	$\mathbf{30.18} \pm 2.21$	$\mathbf{0.828} \pm 0.05$	$\mathbf{0.148} \pm 0.04$	$\mathbf{30.69} \pm 2.14$	$\mathbf{0.832} \pm 0.05$

Table 2: **Quantitative results of Gaussian and nonlinear deblurring on the CelebA-HQ dataset.** Input images have an additive Gaussian noise with $\sigma_{\boldsymbol{y}} = 0.01$. Best results are in bold and second best results are underlined. For nonlinear deblurring, some baselines are omitted, as they can only solve *linear* inverse problems.

Quantitative Results

ReSample achieves SOTA or comparable performance on sparse-view CT reconstruction

Method	Abdominal		Head		Chest	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Latent-DPS	26.80 ±1.09	0.870 ±0.026	28.64 ±5.38	0.893 ±0.058	25.67±1.14	0.822 ±0.033
MCG (Chung et al., 2022)	29.41 ±3.14	0.857 ±0.041	28.28 ±3.08	0.795 ±0.116	27.92 ±2.48	0.842 ±0.036
DPS (Chung et al., 2023a)	27.33 ±2.68	0.715 ± 0.031	24.51 ±2.77	0.665 ±0.058	24.73 ±1.84	0.682 ±0.113
PnP-UNet (Gilton et al., 2021)	<u>32.84</u> ±1.29	<u>0.942</u> ±0.008	<u>33.45</u> ±3.25	<u>0.945</u> ±0.023	29.67 ±1.14	<u>0.891</u> ±0.011
FBP	26.29 ±1.24	0.727 ±0.036	26.71 ±5.02	0.725 ±0.106	24.12 ±1.14	0.655 ±0.033
FBP-UNet (Jin et al., 2017)	32.77 ±1.21	0.937 ±0.013	31.95 ±3.32	0.917 ±0.048	<u>29.78</u> ±1.12	0.885 ±0.016
ReSample (Ours)	**35.91** ±1.22	**0.965** ±0.007	**37.82** ±5.31	**0.978** ±0.014	**31.72** ±0.912	**0.922** ±0.011

Table 3: **Quantitative results of CT reconstruction on the LDCT dataset**. Best results are in bold and second best results are underlined.

Conclusion

We propose ReSample, an algorithm that can effectively leverage LDMs to solve general inverse problems. Our contributions and limitations are summarized below:

- Our algorithm has a high impact in both the industry and academia since we are the first to enable a strong pre-trained prior (LDM) for image restoration with measurement consistency
- The applications of ReSample with high-dimensional data are of high interest to the both the industry and the academia.
- One limitation of our method lies in the computational overhead of hard data consistency, which we leave as a significant challenge for future work to address and improve upon.

Future Works

Latent diffusion models are found to be most effective in 1) **High dimensional data** 2) **Multimodal data such as text**

- It is imperative to apply ReSample for high-dimensional data, such as 3D or video inverse problems.
- It is very important to accelerate the inference time of ReSample, as now the algorithm takes 500-1000 NFEs.
- It is crucial to utilize more advanced multimodal latent diffusion models for solving inverse problems such as stable diffusion. Developing a method that can utilize radiology report for solving inverse problems would be very impactful for medical image reconstruction.

Thanks for listening

Thank you for listening!

References

[1] Chung, H., Kim, J., Mccann, M. T., Klasky, M. L., Ye, J. C. (2022). Diffusion posterior sampling for general noisy inverse problems. arXiv preprint arXiv:2209.14687.

[2] Rombach, Robin, et al. "High-resolution image synthesis with latent diffusion models." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.

[3] Fabian, Zalan, Berk Tinaz, and Mahdi Soltanolkotabi. "DiracDiffusion: Denoising and Incremental Reconstruction with Assured Data-Consistency." arXiv preprint arXiv:2303.14353 (2023).

[4] Song, Yang, et al. "Solving inverse problems in medical imaging with score-based generative models." International Conference on Learning Representations. 2022

[5] Dhariwal, Prafulla, and Alexander Nichol. "Diffusion models beat gans on image synthesis." Advances in Neural Information Processing Systems 34 (2021): 8780-8794.

[6] Shen, Liyue, Wei Zhao, and Lei Xing. "Patient-specific reconstruction of volumetric computed tomography images from a single projection view via deep learning." Nature biomedical engineering 3.11 (2019): 880-888.

[7] Song, Bowen, Liyue Shen, and Lei Xing. "PINER: Prior-informed Implicit Neural Representation Learning for Test-time Adaptation in Sparse-view CT Reconstruction." Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2023.

[8] Ongie, Gregory, et al. "Deep learning techniques for inverse problems in imaging." IEEE Journal on Selected Areas in Information Theory 1.1 (2020): 39-56.

[9] Ho, Jonathan, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models." Advances in neural information processing systems 33 (2020): 6840-6851.

[10] Song, Bowen, et al. "Solving Inverse Problems with Latent Diffusion Models via Hard Data Consistency." International Conference on Learning Representations. 2024.

Proposition 1 (Stochastic Encoding). *Since the sample $\hat{\boldsymbol{z}}_t$ given $\hat{\boldsymbol{z}}_0(\boldsymbol{y})$ and measurement $\boldsymbol{y}$ is conditionally independent of $\boldsymbol{y}$, we have that*

$$p(\hat{\boldsymbol{z}}_t|\hat{\boldsymbol{z}}_0(\boldsymbol{y}), \boldsymbol{y}) = p(\hat{\boldsymbol{z}}_t|\hat{\boldsymbol{z}}_0(\boldsymbol{y})) = \mathcal{N}(\sqrt{\bar{\alpha}_t}\hat{\boldsymbol{z}}_0(\boldsymbol{y}), (1-\bar{\alpha}_t)\boldsymbol{I}). \tag{12}$$

This proposition[2] accounts for Algorithm 1: Basic ReSample

Proposition 2 (Stochastic Resampling). *Suppose that $p(\boldsymbol{z}'_t|\hat{\boldsymbol{z}}_t, \hat{\boldsymbol{z}}_0(\boldsymbol{y}), \boldsymbol{y})$ is normally distributed such that $p(\boldsymbol{z}'_t|\hat{\boldsymbol{z}}_t, \hat{\boldsymbol{z}}_0(\boldsymbol{y}), \boldsymbol{y}) = \mathcal{N}(\boldsymbol{\mu}_t, \sigma_t^2)$. If we let $p(\hat{\boldsymbol{z}}_t|\hat{\boldsymbol{z}}_0(\boldsymbol{y}), \boldsymbol{y})$ be a prior for $\boldsymbol{\mu}_t$, then the posterior distribution $p(\hat{\boldsymbol{z}}_t|\boldsymbol{z}'_t, \hat{\boldsymbol{z}}_0(\boldsymbol{y}), \boldsymbol{y})$ is given by*

$$p(\hat{\boldsymbol{z}}_t|\boldsymbol{z}'_t, \hat{\boldsymbol{z}}_0(\boldsymbol{y}), \boldsymbol{y}) = \mathcal{N}\left(\frac{\sigma_t^2\sqrt{\bar{\alpha}_t}\hat{\boldsymbol{z}}_0(\boldsymbol{y}) + (1-\bar{\alpha}_t)\boldsymbol{z}'_t}{\sigma_t^2 + (1-\bar{\alpha}_t)}, \frac{\sigma_t^2(1-\bar{\alpha}_t)}{\sigma_t^2 + (1-\bar{\alpha}_t)}\boldsymbol{I}\right). \tag{13}$$

We refer to this new mapping technique as *stochastic resampling*. Since we do not have access to σ_t^2, it serves as a hyperparameter that we tune in our algorithm. The choice of σ_t^2 plays a role of controlling the tradeoff between prior consistency and data consistency. If $\sigma_t^2 \to 0$, then we recover unconditional sampling, and if $\sigma_t^2 \to \infty$, we recover stochastic encoding. We observe that this new technique also has several desirable properties, for which we rigorously prove in the next section.

This proposition accounts for Algorithm 2: Posterior ReSample

[2] Song, Bowen and Kwon, Soo Min et al., Solving inverse problems with latent diffusion models via hard data consistency, International Conference on Learning Representations (ICLR). 2024 **Spotlight**

Lemma 1. *Let $\tilde{z}_t$ and $\hat{z}_t$ denote the stochastically encoded and resampled image of $\hat{z}_0(y)$, respectively. If $VAR(z'_t) > 0$, then we have that $VAR(\hat{z}_t) < VAR(\tilde{z}_t)$.*

Theorem 1. *If $\hat{z}_0(y)$ is measurement-consistent such that $y = \mathcal{A}(\mathcal{D}(\hat{z}_0(y)))$, i.e. $\hat{z}_0 = \hat{z}_0(z_{t+1}) = \hat{z}_0(y)$, then stochastic resample is unbiased such that $\mathbb{E}[\hat{z}_t|y] = \mathbb{E}[z'_t]$.*

These two results, Lemma 1 and Theorem 1, prove the benefits of stochastic resampling. At a high-level, these proofs rely on the fact the posterior distributions of both stochastic encoding and resampling are Gaussian and compare their respective means and variances. In the following result, we characterize the variance induced by stochastic resampling, and show that as $t \to 0$, the variance decreases, giving us a reconstructed image that is of better quality.

Theorem 2. *Let z_0 denote a sample from the data distribution and z_t be a sample from the noisy perturbed distribution at time t. Then,*

$$Cov(z_0|z_t) = \frac{(1-\bar{\alpha}_t)^2}{\bar{\alpha}_t}\nabla^2_{z_t}\log p_{z_t}(z_t) + \frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}\boldsymbol{I}.$$

By Theorem 2, notice that since as α_t is an increasing sequence that converges to 1 as t decreases, the variance between the ground truth z_0 and the estimated $\hat{z}_0$ decreases to 0 as $t \to 0$, assuming that $\nabla^2_{z_t}\log p_{z_t}(z_t) < \infty$. Following our theory, we empirically show that stochastic resampling can reconstruct signals that are less noisy than stochastic encoding, as shown in the next section.

Theoretical analysis from [10][2]

[2] Song, Bowen and Kwon, Soo Min et al., Solving inverse problems with latent diffusion models via hard data consistency, International Conference on Learning Representations (ICLR). 2024 **Spotlight**

UNIVERSITY OF MICHIGAN

Algorithm 1 ReSample: Solving Inverse Problems with Latent Diffusion Models

Require: Measurements $\boldsymbol{y}$, $\mathcal{A}(\cdot)$, Encoder $\mathcal{E}(\cdot)$, Decoder $\mathcal{D}(\cdot)$, Score function $\boldsymbol{s}_\theta(\cdot, t)$, Pretrained LDM Parameters β_t, $\bar{\alpha}_t$, η, δ, Hyperparameter γ to control σ_t^2, Time steps to perform resample C

$\boldsymbol{z}_T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ ▷ Initial noise vector

for $t = T-1, \ldots, 0$ **do**

  $\boldsymbol{\epsilon}_1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$

  $\hat{\boldsymbol{\epsilon}}_{t+1} = \boldsymbol{s}_\theta(\boldsymbol{z}_{t+1}, t+1)$ ▷ Compute the score

  $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_{t+1}) = \frac{1}{\sqrt{\bar{\alpha}_{t+1}}}(\boldsymbol{z}_{t+1} - \sqrt{1-\bar{\alpha}_{t+1}}\hat{\boldsymbol{\epsilon}}_{t+1})$ ▷ Predict $\hat{\boldsymbol{z}}_0$ using Tweedie's formula

  $\boldsymbol{z}_t' = \sqrt{\bar{\alpha}_t}\hat{\boldsymbol{z}}_0(\boldsymbol{z}_{t+1}) + \sqrt{1-\bar{\alpha}_t - \eta\delta^2}\hat{\boldsymbol{\epsilon}}_{t+1} + \eta\delta\boldsymbol{\epsilon}_1$ ▷ Unconditional DDIM step

  if $t \in C$ **then** ▷ ReSample time step

    $\hat{\boldsymbol{z}}_0(\boldsymbol{y}) \in \arg\min_{\boldsymbol{z}} \frac{1}{2}\|\boldsymbol{y} - \mathcal{A}(\mathcal{D}(\boldsymbol{z}))\|_2^2$ ▷ Solve with initial point $\hat{\boldsymbol{z}}_0(\boldsymbol{z}_{t+1})$

    $\boldsymbol{z}_t = \text{StochasticResample}(\hat{\boldsymbol{z}}_0(\boldsymbol{y}), \boldsymbol{z}_t', \gamma)$ ▷ Map back to t

  else

    $\boldsymbol{z}_t = \boldsymbol{z}_t'$ ▷ Unconditional sampling if not resampling

$\boldsymbol{x}_0 = \mathcal{D}(\boldsymbol{z}_0)$ ▷ Output reconstructed image