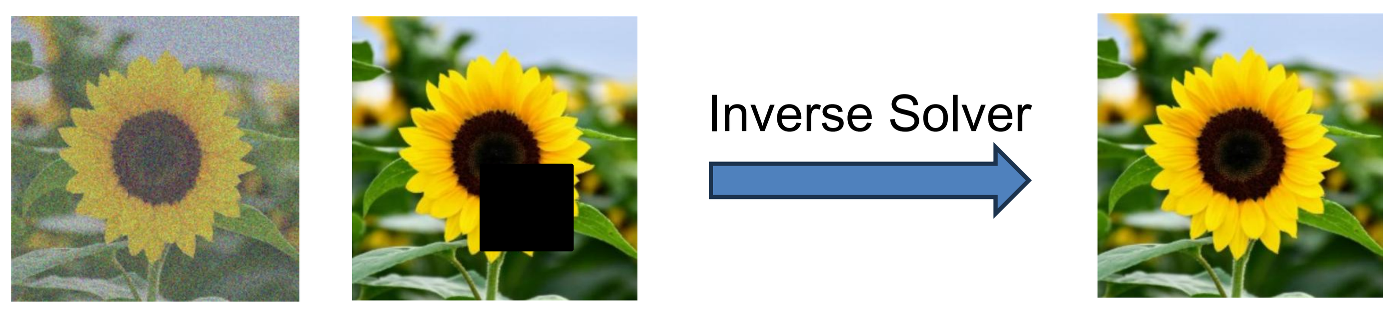
# Solving Inverse Problems with Diffusion Model-based Priors

Julia Feldhaus
EE367: Computational Imaging, Stanford University

## Motivation

Restoring blurry or incomplete images is an ill-posed problem, with infinite solutions. The challenge is knowing which solution is best.




Diffusion models tackle this challenge by iteratively denoising an image while balancing both a learned prior and observed data. This project explores five methods for using pretrained diffusion models to solve inverse problems and compares the results.

## Related Work

Key approaches to inverse problems:

(1) Maximum Likelihood – Simple to compute but poorly suited to ill-posed problems
(2) Maximum A Posteriori – Considers prior and physical model, but gives single point estimate (lacks detail)
(3) Posterior Sampling – Samples from feasible solution set, best for highly ill-posed problems

## References

[1] Chung, Kim, Mccann, Klasky, and Ye (2023). Diffusion posterior sampling for general noisy inverse problems. In *ICLR*.
[2] Ho, Jain, and Abbeel (2020). Denoising diffusion probabilistic models. In *NeurIPS*.
[3] Jalal, Arvinte, Daras, Price, Dimakis, and Tamir (2021). Robust compressed sensing mri with deep generative priors.
[4] Meng, He, Y. Song, J. Song, Wu, Zhu, and Ermon (2022). Sdedit: Guided image synthesis and editing with stochastic differential equations.

## Techniques

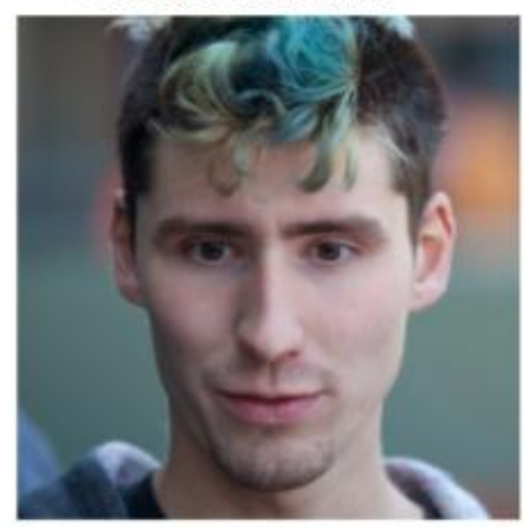

Train Model → Pretrained weights (prior)

Score: $\nabla_{x_t} \log p_t(x_t) = \dfrac{\sqrt{\bar{\alpha}_t}\mathbb{E}[x_0|x_t] - x_t}{1-\bar{\alpha}_t}$

Prediction: $\hat{x}_0^{(\mathrm{VP})} = \mathbb{E}[\mathrm{x}_0|\mathrm{x}_t] = \dfrac{1}{\sqrt{\bar{\alpha}_t}}\left(x_t + (1-\bar{\alpha}_t)\nabla_{x_t}\log p_t(x_t)\right)$

### Single-Step Image Denoising

Ground Truth

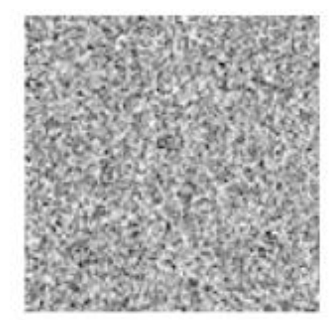


Add Noise ($x_t$)

Predict clean image in single step ($\hat{x}_0$)

### Unconditional Image Generation (DDPM)

Gaussian Noise



Predict clean image ($\hat{x}_0$)

Step towards prediction and add some noise back in

$$x_{t-1} = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}\hat{x}_0 + \sqrt{1-\alpha_t}\mathbf{z}$$

### Score-Distillation Editing

Corrupted Image

Add in t timesteps worth of gaussian noise

Predict clean image ($\hat{x}_0$)

Step towards prediction and add some noise back in (Same $x_{t-1}$ as previous method

### Score ALD

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0},\mathbf{I})$
**for** $t = T,\ldots,1$ **do**
$\mathbf{z} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ if $t>1$, else $\mathbf{z}=\mathbf{0}$
$\hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t + (1-\bar{\alpha}_t)\mathbf{s}_\theta(x_t,t))$
$\mathbf{x}_{t-1} = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}\hat{\mathbf{x}}_0 + \sqrt{1-\alpha_t}\mathbf{z}$
$\mathbf{x}_{t-1} = \mathbf{x}_{t-1} - \frac{1}{2(\sigma^2+\gamma_t^2)}\nabla_{\mathbf{x}_t}\left\|\mathscr{A}(\mathbf{x}_t)-\mathbf{y}\right\|^2$
**end for**
**return** $\mathbf{x}_0$

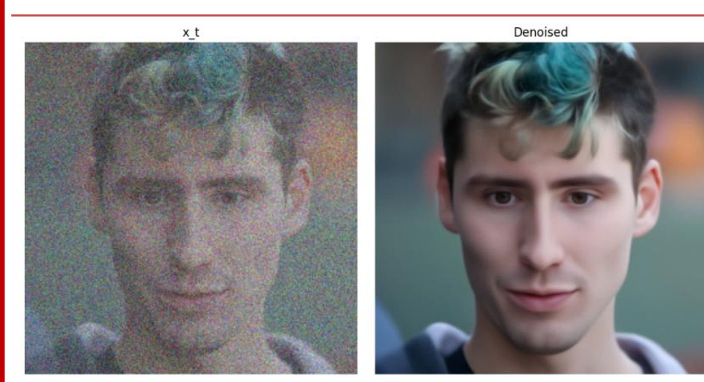
$\gamma_t$ = Annealing Schedule

### Diffusion Posterior Sampling

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0},\mathbf{I})$
**for** $t = T,\ldots,1$ **do**
$\mathbf{z} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ if $t>1$, else $\mathbf{z}=\mathbf{0}$
$\hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t + (1-\bar{\alpha}_t)\mathbf{s}_\theta(x_t,t))$
$\mathbf{x}'_{t-1} = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}\hat{\mathbf{x}}_0 + \sqrt{1-\alpha_t}\mathbf{z}$
$\mathbf{x}_{t-1} = \mathbf{x}'_{t-1} - \zeta_t\nabla_{\mathbf{x}_t}\left\|\mathscr{A}(\hat{\mathbf{x}}_0)-\mathbf{y}\right\|^2$
**end for**
**return** $\mathbf{x}_0$

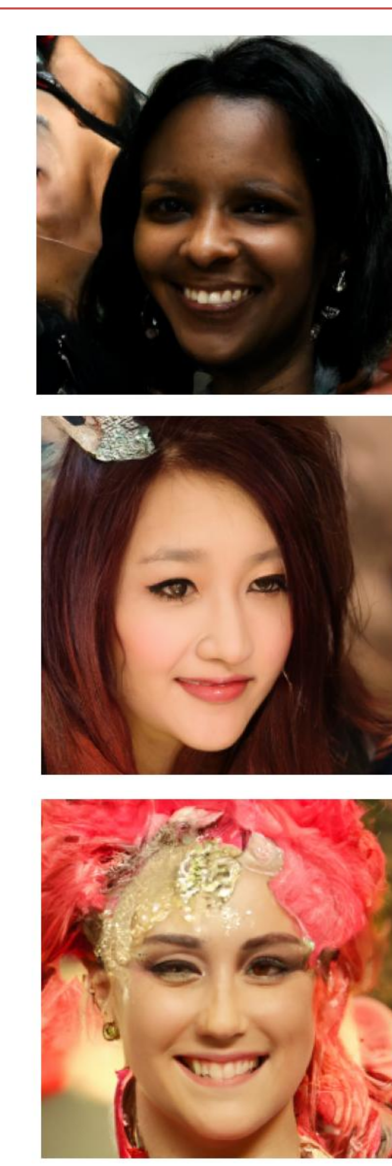
$$\zeta_t = \frac{\zeta}{\left\|\nabla_{\mathbf{x}_t}\left\|\mathscr{A}(\hat{\mathbf{x}}_0)-\mathbf{y}\right\|^2\right\|}$$

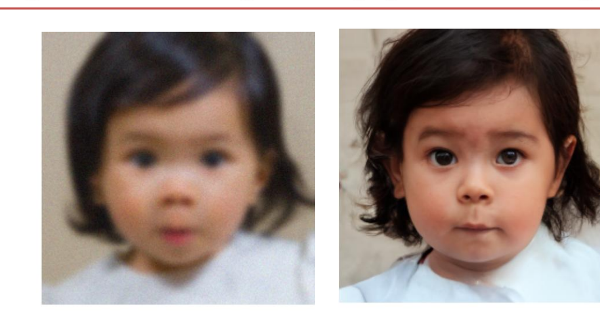
## Experimental Results



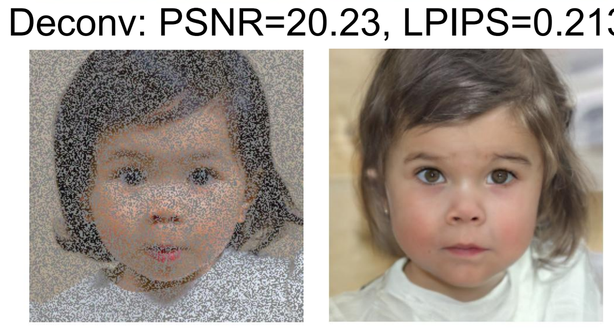
T=100: Low noise
PSNR = 27.69
LPIPS = 0.1771

T=500: High noise
PSNR = 23.17
LPIPS = 0.3514

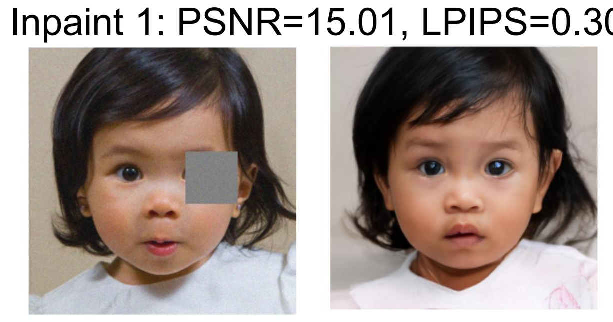




Deconv: PSNR=20.23, LPIPS=0.213

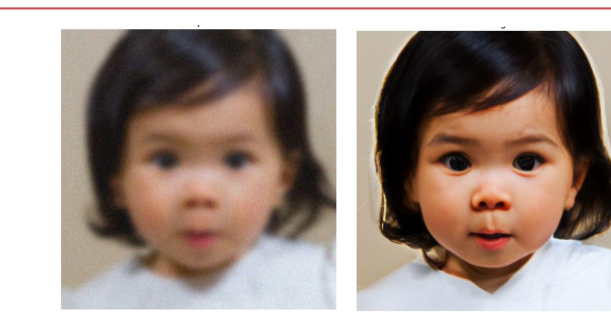


Inpaint 1: PSNR=15.01, LPIPS=0.300

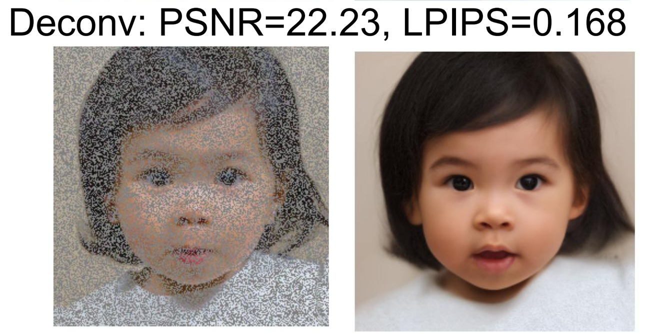


Inpaint 2: PSNR=20.80, LPIPS=0.172

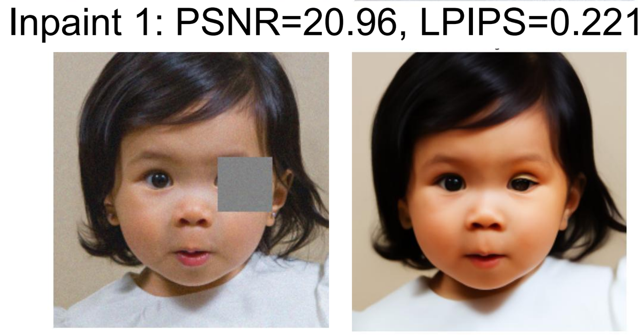
Less noise: Partial reconstruction
More noise: Hallucination



Deconv: PSNR=22.23, LPIPS=0.168

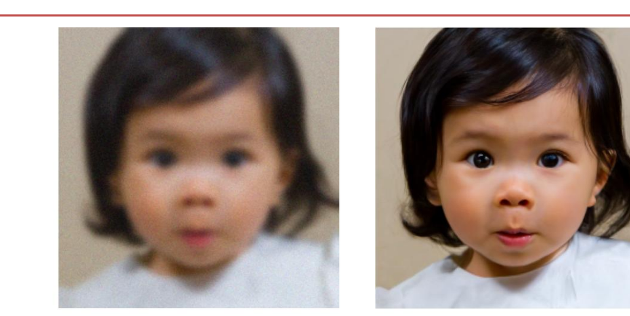


Inpaint 1: PSNR=20.96, LPIPS=0.221



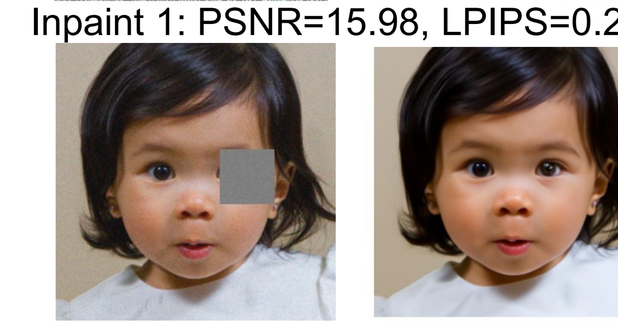
Inpaint 2: PSNR=23.15, LPIPS=0.119

Higher annealing factor, more reliance on measurements



Deconv: PSNR=28.42, LPIPS=0.058

Inpaint 1: PSNR=15.98, LPIPS=0.234



Inpaint 2: PSNR=34.05, LPIPS=0.023

Higher scale, more reliance on measurements