Solving Inverse Problems with Diffusion Model-based Priors

Julia Feldhaus

Abstract—Inverse problems in computational imaging require recovering clean signals from corrupted measurements, a task that is often ill-posed. Diffusion models have recently emerged as powerful generative models that serve as effective priors for these image reconstruction tasks. In this work, we explore several approaches for solving inverse imaging problems using a pretrained diffusion model. The methods investigated include single-step denoising, unconditional image generation, Score-Distillation Editing (SDEdit), Score-based Annealed Langevin Dynamics (ScoreALD), and Diffusion Posterior Sampling (DPS). These approaches are evaluated on several reconstruction tasks including denoising, deconvolution, and inpainting. Reconstruction quality is assessed using quantitative metrics such as PSNR and LPIPS, along with qualitative visual analysis. The results demonstrate that diffusion-based posterior sampling methods can effectively balance learned image priors with measurement consistency, producing realistic reconstructions across multiple inverse problems.

Index Terms—Computational Imaging, Inverse Problems, Diffusion Models, Posterior Sampling

1 Introduction

The restoration of corrupted photographs is a frequent challenge in computational imaging. These corruptions can arise from many sources, including damaged hardware, downsampling loss, sensor noise, motion or defocus blur, and physical obstructions. Recovering the original signal from corrupted measurements is an ill-posed inverse problem, as there is no unique solution.

While many techniques exist for solving inverse problems, approaches that incorporate strong statistical priors are often more successful at selecting a plausible reconstruction. Diffusion models provide one such approach, iteratively denoising an image while balancing a learned prior with the observed measurements [1]. These models have recently emerged as powerful AI tools with applications in image and video synthesis. These models are trained by gradually adding noise to many sample images and learning to reverse this process using a deep neural network. Once trained, the model captures the structure and statistical trends of the training dataset and can serve as a strong prior for solving inverse problems. By combining the diffusion prior with measurement constraints, these models can be adapted to a wide range of image reconstruction and generation tasks.

In this project, we explore several methods for solving inverse image problems using a pretrained diffusion model. The methods include single-step image denoising, unconditional image generation, score-distillation editing (SDEdit), ScoreALD, and Diffusion Posterior Sampling (DPS). The pretrained diffusion model is downloaded from the "Diffusion Posterior Sampling for General Noisy Inverse Problems" repository [2]. It uses a variance-preserving formulation with a score-predicting network and is trained on the Flickr-Faces-HQ (FFHQ) dataset. Each of the explored methods is evaluated both qualitatively and quantitatively using reconstruction metrics such as PSNR and LPIPS to compare their ability to recover realistic images from corrupted measurements.

• *J. Feldhaus is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305.*
E-mail: jcfeld@stanford.edu

2 Related Work

Solving inverse problems is a central challenge in computational imaging, with a wide range of proposed solutions. Among the most common approaches are maximum likelihood estimation (MLE), maximum a posteriori estimation (MAP), and posterior sampling, each offering different strengths and limitations.

Maximum likelihood estimation (MLE) involves choosing the model parameters that are most likely to generate the observed data. This method is popular due to its relative computational efficiency and ability to derive closed-form solutions. Despite these advantages, MLE is often insufficient for ill-posed inverse problems, where multiple plausible solutions exist, because it considers only the likelihood of the observed measurements without accounting for prior knowledge [3]. Maximum a posteriori (MAP) estimation extends MLE by incorporating prior information about the solution, reflecting any previously observed data or domain-specific knowledge. MAP is typically solved using convex optimization techniques to produce a single most probable reconstructed solution. Although this approach effectively balances observed measurements and prior beliefs, MAP may fail to capture the full uncertainty and breadth of solutions present in highly ill-posed inverse problems [4].

In contrast to MLE and MAP, which produce single-solution estimates, posterior sampling generates samples from the full set of feasible solutions. This approach produces multiple image reconstructions that are consistent with both the measurements and prior beliefs, which can optionally be synthesized to a single image if one solution is desired. By representing the range of possible solutions, this approach captures the full uncertainty of the problem and is

particularly effective for highly ill-posed tasks. As a result, posterior sampling has become a powerful tool for complex image reconstruction challenges such as deblurring, inpainting, and other tasks where a single estimate may not be sufficient [5].

3 Proposed Method

This project considers the variance-preserving (VP) formulation of diffusion models introduced in the *Denoising Diffusion Probabilistic Models* paper [6]. In this formulation, the forward diffusion process progressively corrupts an image by adding Gaussian noise through a Markov chain. The transition from step $t-1$ to t is defined as

$$x_t = \sqrt{1-\beta_t}\, x_{t-1} + \sqrt{\beta_t}\, z_{t-1}, \quad t = 1, 2, \ldots, T \tag{1}$$

where $z_{t-1} \sim \mathcal{N}(0, I)$ are independent Gaussian noise variables and β_t defines the noise schedule. Using the definitions $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, the forward diffusion process can also be expressed directly in terms of the initial image x_0 as

$$x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, z, \quad z \sim \mathcal{N}(0, I). \tag{2}$$

The proof of this equivalence is provided in Appendix A. This closed-form representation simplifies calculations by allowing samples from any diffusion step to be generated directly from the original image without iteratively applying all intermediate steps.

The reverse diffusion process aims to invert this added corruption by gradually denoising the image using a learned neural network. Using the score-based formulation, the reverse step can be written as

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(x_t + (1-\alpha_t) s_\theta(x_t, t) \right), \tag{3}$$

where $s_\theta(x_t, t)$ is the score output for a given image. This single reverse diffusion step can also be written as

$$\hat{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}} \left(x_t + (1-\bar{\alpha}_t) s_\theta(x_t, t) \right) \tag{4}$$

$$x_{t-1} = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t} \hat{x}_0, \tag{5}$$

which is proved in Appendix A. In the original DDPM formulation, the reverse process is instead written using a noise-predicting network $\epsilon_\theta(x_t, t)$, which can be shown to be mathematically equivalent to the score-based formulation in Appendix A. Using the score formulation is convenient for inverse problems, since the score represents the gradient of the log data density and can be easily combined with measurement gradients during posterior sampling [7].

3.1 Single-Step Image Denoising

The first application of the diffusion model explored in this paper is single-step image denoising. In this technique, Gaussian noise is added to a clean ground-truth image to obtain a noisy image x_t. The model then predicts the corresponding clean image $\hat{x}_0$ in a single step using Equation (4). This method is evaluated at varying levels of added noise in order to assess the model's ability to recover the underlying clean signal from increasingly corrupted observations.

3.2 Unconditional Image Generation

In unconditional image generation, the process is initialized with an image consisting entirely of Gaussian noise. The Denoising Diffusion Probabilistic Models (DDPM) reverse process is then used to iteratively generate an approximate clean image. At each timestep, the model predicts an estimate of the clean image $\hat{x}_0$ from the current noisy sample, and the sample is updated by taking a small step toward this prediction. A small amount of Gaussian noise is added at each timestep to prevent the process from collapsing to a single average solution and instead encourage the model to explore multiple possible solution paths. This method does not use reconstruction metrics such as PSNR or LPIPS, since there is no ground-truth reference image for comparison.

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \ldots, 1$ **do**
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
  $\hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t + (1-\bar{\alpha}_t)\boldsymbol{s}_\theta(x_t, t))$
  $\boldsymbol{x}_{t-1} = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\boldsymbol{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}\hat{\mathbf{x}}_0 + \sqrt{1-\alpha_t}\boldsymbol{z}$
end for
return $\mathbf{x}_0$

Fig. 1. $\hat{x}_0$ and score-based formulation of the DDPM reverse process

3.2.1 Score-Distillation Editing

Score-Distillation Editing (SDEdit) also uses the DDPM reverse process described in Fig. 1. However, instead of starting from pure noise, the method begins with a corrupted input image that must be restored. The forward diffusion process (Equation 2) is first applied to add t timesteps of Gaussian noise to the image. The reverse DDPM process is then run for t steps to gradually remove the noise and recover a refined version of the original image. This method is evaluated at varying levels of added noise to examine how the diffusion prior can improve corrupted observations while preserving the overall structure of the input image.

3.3 Posterior Sampling

The first posterior sampling method explored in this project is ScoreALD (Annealed Langevin Dynamics). ScoreALD follows a similar iterative process to unconditional image generation, but incorporates an additional gradient term that enforces consistency with the observed corrupted image (Figure 2). Specifically, the update step includes a scaled gradient of the squared difference between a corrupted version of the current prediction and the true corrupted measurement.

This approach produces reconstructions that balance the diffusion model prior with the physical measurement model, resulting in images that are both realistic face reconstructions and consistent with details in the observed data. As the sampling process progresses, the scaling of the gradient term changes. Early in the process the noise level σ is high, so the gradient term is reduced and the model relies more heavily on the diffusion prior. Later in the process, the

image is less noisy, so the scaling term increases and the model is able to fine-tune details to better match the ground truth corrupted image. The annealing parameter γ_t can also be adjusted to control the balance between the prior and the measurement terms.

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

Fig. 2. ScoreALD reverse iterative process

Diffusion Posterior Sampling (DPS) is the second posterior sampling method explored in this project. Like ScoreALD, DPS combines the diffusion prior with a measurement consistency term to guide the reconstruction process. However, instead of computing the gradient using a corrupted estimate of the current iterate x_t, DPS computes the gradient using a corrupted estimate of the predicted clean image $\hat{x}_0$ (Fig. 3). The gradient scaling term is also modified in DPS. Instead of using a fixed annealing factor, DPS introduces the normalization term

$$\xi_t = \frac{\xi}{\|\nabla_{x_t}\|\mathcal{A}(\hat{x}_0) - y\|^2\|} \tag{6}$$

which is defined as an arbitrary scaling parameter ξ divided by the magnitude of the gradient. This normalization helps stabilize the update step.

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

Fig. 3. DPS reverse iterative process

4 Experimental Results

Experiments were conducted on several image reconstruction tasks using the pretrained diffusion model described in the methods section. Each method was evaluated on its ability to reconstruct corrupted face images under different noise levels and reconstruction settings.

4.1 Single-Step Image Denoising

The images produced by the single-step denoising process are included in Fig. 4. For t=100 (lowest noise), the PSNR was 32.54 and LPIPS was 0.0989. For t=300, the PSNR was 27.69 and LPIPS was 0.1771. For t=500 (highest noise), the PSNR was 23.17 and LPIPS was 0.3514.

As noise level increases, the PSNR decreases and LPIPS increases. This is consistent with the images produced, in which the reconstructions become blurrier and less similar to the target image as noise level increases.

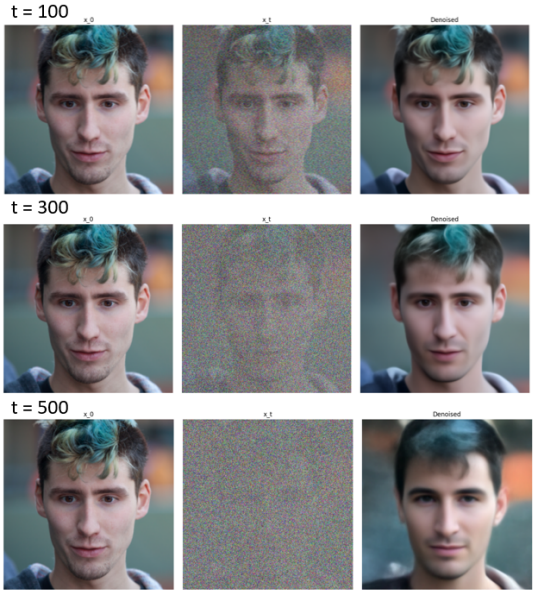

Fig. 4. Target, noisy, and denoised images for single-step image denoising at varying levels of noise (t)

4.2 Unconditional Image Generation

As the unconditional image generation method begins with pure Gaussian noise, there is no target image to compare to. Each generation is a unique face. Thus, there are no quantitative statistics to present. Qualitatively, each generated face image is not blurry or noisy. They look like real human faces, as defined by the diffusion model prior. However, the images contain slight imperfections that make them identifiable as generated, such as random backgrounds, clothing, and hair. These images are provided in Fig. 5.

Fig. 5. Unconditionally generated faces

4.3 Score Distillation Editing

In the SDEdit experiment, the SDEdit process was conducted for deconvolution and inpainting on a missing

square. This test was repeated three times, for t=100, t=500, and t=800. The varying t value represents the number of noising steps, with a higher number of steps representing more initial noise in the image. Figures 6 and 7 show the generated images for each noise level on the two image inverse problems.

The reconstruction metrics for deconvolution and inpainting are given in tables 1 and 2, respectively. The tables show that generally PSNR decreases and LPIPS increases as the noise level increases. This is because SDEdit does not have a data fidelity term, so as more noise is added, the reconstruction looks less like the original measurement. However, based on qualitative analysis, less noise is not always a better reconstruction. At t=100, the resulting images are still partially corrupted (blurry/missing data). At t=800, the results are clear and complete, but the face produced does not look like the original image. This demonstrates a tradeoff between hallucination at high noise and lack of reconstruction at low noise that must be balanced.

TABLE 1
Reconstruction metrics for deconvolution at different noise levels.

t	PSNR	LPIPS
100	25.060	0.242
500	20.230	0.213
800	12.967	0.364

TABLE 2
Reconstruction metrics for square inpainting at different noise levels.

t	PSNR	LPIPS
100	25.294	0.088
500	20.803	0.172
800	13.567	0.386

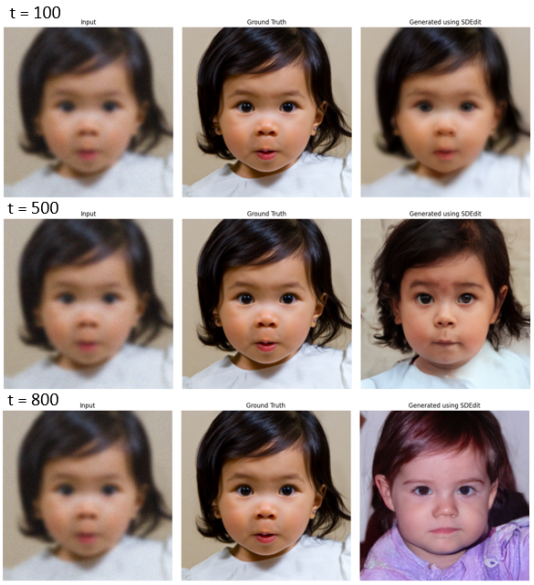

Fig. 6. Deconvolution with SDEdit at varying noise levels

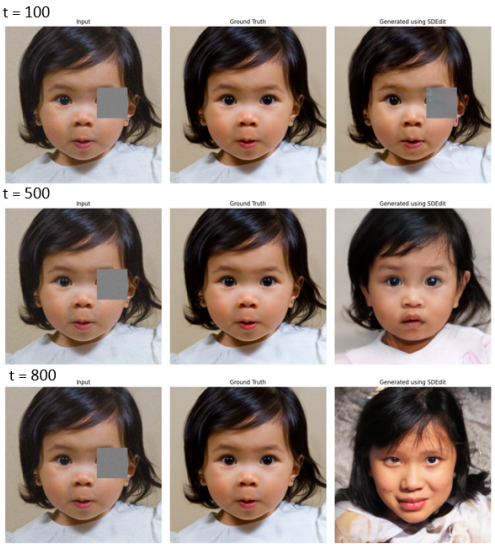

Fig. 7. Square inpainting with SDEdit at varying noise levels

4.4 ScoreALD

The ScoreALD experiment was also completed on two inverse image problems: deconvolution and inpainting. In this experiment, the annealing factor γ_t is changed over time in order to adjust the weighting between prior and measurement. The annealing factor is defined to increase linearly over time, starting from a lowest value and increasing an equal amount for each time step until arriving at the final reconstruction. This is repeated for three γ_t sequences. For inpainting tasks, the annealing factor bounds are 10 to 15 (decreased), 15 to 20 (default), and 20 to 25 (increased). For deconvolution, the annealing factor bounds are 5 to 10 (decreased), 10 to 15 (default), and 15 to 20 (increased).

TABLE 3
Reconstruction metrics for deconvolution at different annealing levels.

Annealing Setting	PSNR	LPIPS
Decreased	25.194	0.106
Default	22.230	0.168
Increased	18.485	0.230

TABLE 4
Reconstruction metrics for inpainting at different annealing levels.

Annealing Setting	PSNR	LPIPS
Decreased	26.289	0.093
Default	23.145	0.119
Increased	19.925	0.183

Tables 3 and 4 report the reconstruction metrics for each task at varied levels of annealing. In both reconstruction problems, increasing annealing leads to decreased PSNR and increased LPIPS. This indicates that a higher annealing factor produces a result that is less true to the ground truth uncorrupted image. However, this does not capture the full scope of the annealing factor's effect. In the generated images in (8) and (9), a lower annealing factor corresponds

to a less complete reconstruction with present corruption artifacts. This is because a low annealing factor indicates a high reliance on measurement fidelity, so the model output is closer to the original corrupted image. As annealing increases, the model relies more on the diffusion model prior. Thus, the resulting images have less corruption, but appear less similar to the ground truth face.

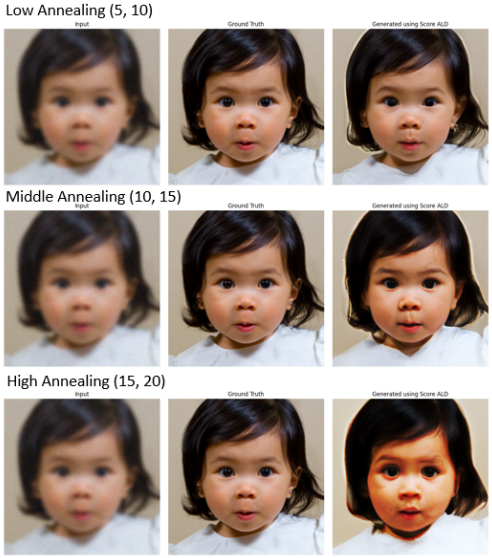

Fig. 8. Deconvolution using ScoreALD at varied annealing levels

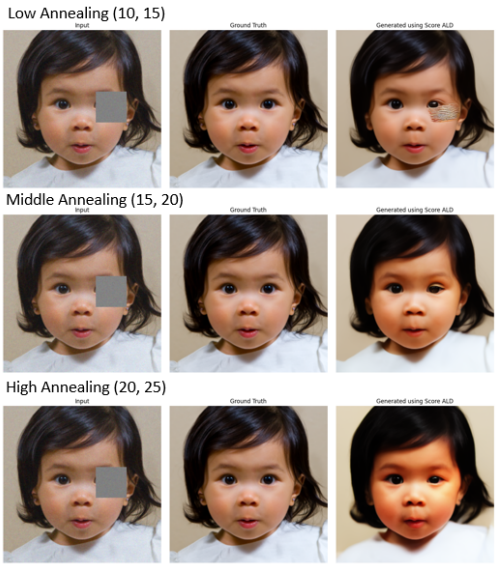

Fig. 9. Inpainting using ScoreALD at varied annealing levels

4.5 Diffusion Posterior Sampling

In the Diffusion Posterior Sampling experiment, the DPS process was applied to deconvolution, square inpainting, and random inpainting with scaling parameters $\xi = 0.1, 0.3, 1.0$. The generated images for each method are given in Figures 10-12, and quantitative evaluation metrics for each task are given in Tables 5–7. For deconvolution, the default scaling value $\xi = 0.3$ produced the best reconstruction, achieving the highest PSNR and lowest LPIPS. For square inpainting, larger scaling parameters performed best, and for random inpainting, a lower scaling parameter performed best.

Qualitatively, it is clear in the images that a higher scale factor corresponds to generated images that more closely visually resemble the corrupted measurement. For this reason, random inpainting results look much better at lower scales. The higher the scale, the more corruption pattern becomes integrated into the image. The opposite trend occurs for square inpainting, since most of the observed image outside the missing region is identical to the ground truth. A higher scale factor is able to preserve details from those parts of the image to reconstruct the clean estimate more accurately. Overall, these results show that the scaling parameter ξ controls the balance between measurement consistency and the diffusion prior, and that the optimal value depends on the structure of the inverse problem.

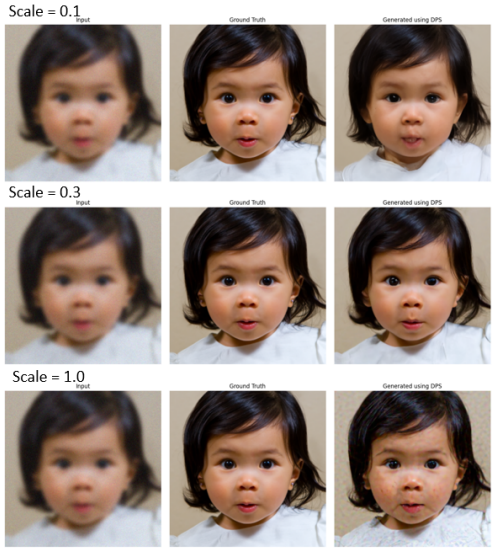

Fig. 10. Deconvolution with DPS at varying scale levels

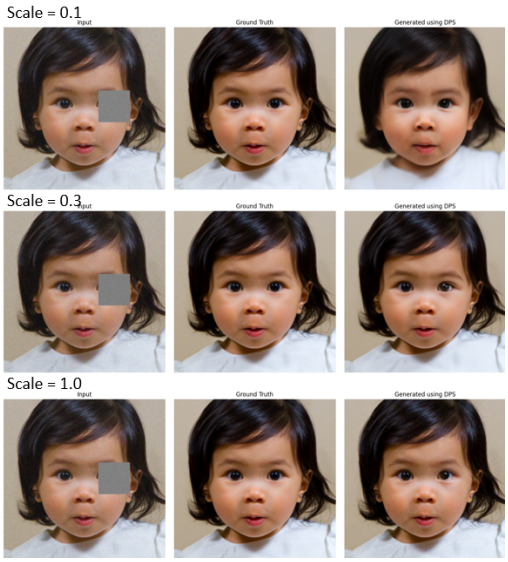

Fig. 11. Square inpainting with DPS at varying scale levels

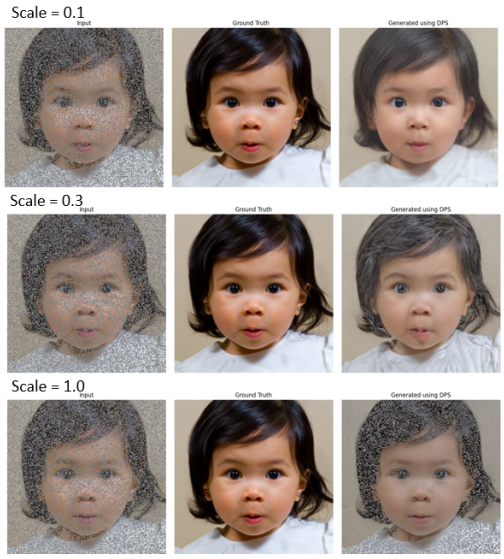

Fig. 12. Random inpainting with DPS at varying scale levels

TABLE 5
Reconstruction metrics for deconvolution using Diffusion Posterior Sampling (DPS) at different scaling parameters ξ.

Scaling Parameter ξ	PSNR	LPIPS
0.1	24.723	0.114
0.3 (default)	28.424	0.058
1.0	28.260	0.112

TABLE 6
Reconstruction metrics for square inpainting using Diffusion Posterior Sampling (DPS) at different scaling parameters ξ.

Scaling Parameter ξ	PSNR	LPIPS
0.1	26.440	0.110
0.3 (default)	34.054	0.023
1.0	34.677	0.022

TABLE 7
Reconstruction metrics for random inpainting using Diffusion Posterior Sampling (DPS) at different scaling parameters ξ.

Scaling Parameter ξ	PSNR	LPIPS
0.1	16.048	0.208
0.3 (default)	15.978	0.235
1.0	14.491	0.700

5 Discussion

The experiments demonstrate that diffusion model priors can effectively solve several image reconstruction tasks, including denoising, deconvolution, and inpainting. Methods that incorporate measurement consistency, such as ScoreALD and DPS, generally produced more accurate reconstructions than approaches that rely purely on the diffusion prior. The results also show that hyperparameters such as the annealing factor and scaling parameter strongly influence the balance between measurement fidelity and prior guidance.

However, this study has several limitations. First, experiments were conducted only on face images from the FFHQ dataset using a single pretrained diffusion model, which may limit generalization to other image domains. Additionally, only a small number of inverse problems and hyperparameter settings were explored, and reconstruction quality was evaluated using a limited set of metrics.

Future work could extend this study by testing additional datasets, exploring more inverse problems, and investigating adaptive strategies for selecting parameters such as the annealing and scaling factors. Further analysis of computational efficiency and uncertainty in posterior samples could also provide deeper insight into the strengths and limitations of diffusion-based reconstruction methods. Recent work on improving the efficiency of diffusion-based inverse solvers by combining diffusion sampling with classical optimization, would also be a promising path to explore [8].

6 Conclusion

This paper explored several diffusion-model-based approaches for solving inverse imaging problems, including single-step denoising, SDEdit, ScoreALD, and Diffusion Posterior Sampling. Experimental results showed that diffusion priors are capable of producing realistic reconstructions while incorporating measurement constraints. Among the methods tested, posterior sampling approaches demonstrated the strongest performance, with reconstruction quality depending on the balance between the diffusion prior and measurement consistency. These results highlight the potential of diffusion models as flexible tools for solving challenging inverse problems in computational imaging.

Acknowledgments

The authors would like to thank the EE367 teaching staff for all of their support throughout the course of this project.

References

[1] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
[2] H. Chung, J. Kim, M. T. McCann, M. L. Klasky, and J. C. Ye, "Diffusion posterior sampling for general noisy inverse problems," in *International Conference on Learning Representations (ICLR)*, 2023.
[3] Y. Pawitan, *In All Likelihood: Statistical Modelling and Inference Using Likelihood*. Oxford University Press, 2001.
[4] J. Dupont *et al.*, "Bayesian inverse problems and convexity," *arXiv preprint arXiv:1602.08590*, 2016.
[5] J. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems*. Springer, 2005.
[6] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *NeurIPS*, 2020.
[7] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations (ICLR)*, 2021.
[8] H. Chung, S. Lee, and J. C. Ye, "Decomposed diffusion sampler for accelerating large-scale inverse problems," in *International Conference on Learning Representations (ICLR)*, 2024.

APPENDIX

Consider the variance preserving formulation of diffusion models:

$$x_t = \sqrt{1-\beta_t}\, x_{t-1} + \sqrt{\beta_t}\, z_{t-1}, \quad t = 1, 2, \ldots, T, \quad z_{t-1} \sim \mathcal{N}(0, I)$$

Prove that $x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, z$, where $\alpha_t = 1-\beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, $z \sim \mathcal{N}(0, I)$

Let $x_t = \sqrt{1-\beta_t}\, x_{t-1} + \sqrt{\beta_t}\, z_{t-1}$

Then $x_t = \sqrt{1-\beta_t}\left(\sqrt{1-\beta_{t-1}}\, x_{t-2} + \sqrt{\beta_{t-1}}\, z_{t-2}\right) + \sqrt{\beta_t}\, z_{t-1} = \sqrt{1-\beta_t}\sqrt{1-\beta_{t-1}}\, x_{t-2} + \sqrt{1-\beta_t}\sqrt{\beta_{t-1}}\, z_{t-2} + \sqrt{\beta_t}\, z_{t-1}$

$= \sqrt{1-\beta_t}\sqrt{1-\beta_{t-1}}\sqrt{1-\beta_{t-2}}\, x_{t-3} + \sqrt{\beta_t}\, z_{t-1} + \sqrt{1-\beta_t}\sqrt{\beta_{t-1}}\, z_{t-2} + \sqrt{1-\beta_t}\sqrt{1-\beta_{t-1}}\sqrt{\beta_{t-2}}\, z_{t-3}$

$= \sqrt{1-\beta_t}\sqrt{1-\beta_{t-1}} \cdots \sqrt{1-\beta_1}\, x_0 + \sqrt{\beta_t}\, z_{t-1} + \sqrt{1-\beta_t}\sqrt{\beta_{t-1}}\, z_{t-2} + \ldots + \sqrt{1-\beta_t}\sqrt{1-\beta_{t-1}} \cdots \sqrt{1-\beta_2}\sqrt{\beta_1}\, z_0$

$= \sqrt{\alpha_t}\sqrt{\alpha_{t-1}} \cdots \sqrt{\alpha_1}\, x_0 + \sqrt{\beta_t}\, z_{t-1} + \sqrt{1-\beta_t}\sqrt{\beta_{t-1}}\, z_{t-2} + \ldots + \sqrt{1-\beta_t}\sqrt{1-\beta_{t-1}} \cdots \sqrt{1-\beta_2}\sqrt{\beta_1}\, z_0$

$= \sqrt{\prod_{i=1}^{t} \alpha_i}\, x_0 + \sqrt{\beta_t}\, z_{t-1} + \sqrt{1-\beta_t}\sqrt{\beta_{t-1}}\, z_{t-2} + \ldots + \sqrt{1-\beta_t}\sqrt{1-\beta_{t-1}} \cdots \sqrt{1-\beta_2}\sqrt{\beta_1}\, z_0$

$= \sqrt{\bar{\alpha}_t}\, x_0 + \underbrace{\sqrt{\beta_t}\, z_{t-1} + \sqrt{1-\beta_t}\sqrt{\beta_{t-1}}\, z_{t-2} + \ldots + \sqrt{1-\beta_t}\sqrt{1-\beta_{t-1}} \cdots \sqrt{1-\beta_2}\sqrt{\beta_1}\, z_0}_{\bigstar}$

$\bigstar$ This term is the sum of many normal gaussian distributions z_i, scaled by the given coefficients. Since all $z_i \sim \mathcal{N}(0, I)$, the sum $Z = C_1 z_{t-1} + C_2 z_{t-2} + \ldots + C_t z_0$

has mean $C_1 \mu_{t-1} + C_2 \mu_{t-2} + \ldots + C_t \mu_0 = C_1(0) + C_2(0) + \ldots + C_t(0) = 0$

and variance $C_1^2 \sigma_{t-1}^2 + C_2^2 \sigma_{t-2}^2 + \ldots + C_t^2 \sigma_0 = C_1^2 I + C_2^2 I + \ldots + C_t^2 I = (C_1^2 + C_2^2 + \ldots + C_t^2) I$

Thus, $Z \sim \mathcal{N}(0, (C_1^2 + C_2^2 + \ldots + C_t^2) I) = \mathcal{N}(0, \sigma_z^2 I)$

Now we have $\sigma_z^2 = \beta_t + (1-\beta_t)\beta_{t-1} + (1-\beta_t)(1-\beta_{t-1})\beta_{t-2} + \ldots + (1-\beta_t)(1-\beta_{t-1})\ldots(1-\beta_2)\beta_1$

Expand all β_i terms to $(1-(1-\beta_i))$

$\sigma_z^2 = 1 - (1-\beta_t) + (1-\beta_t)\beta_{t-1} + (1-\beta_t)(1-\beta_{t-1})\beta_{t-2} + \ldots + (1-\beta_t)(1-\beta_{t-1})\ldots(1-\beta_2)\beta_1$

$= 1 - (1-\beta_t) + (1-\beta_t)(1-(1-\beta_{t-1})) + (1-\beta_t)(1-\beta_{t-1})(1-(1-\beta_{t-2})) + \ldots + (1-\beta_t)(1-\beta_{t-1})\ldots(1-\beta_2)(1-(1-\beta_1))$

$= 1 - (1-\beta_t) + (1-\beta_t) - (1-\beta_t)(1-\beta_{t-1}) + (1-\beta_t)(1-\beta_{t-1}) - (1-\beta_t)(1-\beta_{t-1})(1-\beta_{t-2}) + \ldots$

$+ (1-\beta_t)(1-\beta_{t-1})\ldots(1-\beta_2) - (1-\beta_t)(1-\beta_{t-1})\ldots(1-\beta_2)(1-\beta_1)$

Each negative term cancels with the next positive term, except the first and last.

This leaves $\sigma_z^2 = 1 - (1-\beta_t)(1-\beta_{t-1})\ldots(1-\beta_2)(1-\beta_1) = 1 - \alpha_t\alpha_{t-1}\ldots\alpha_2\alpha_1 = 1 - \prod_{i=1}^{t} \alpha_i = 1 - \bar{\alpha}_t$

We now have $z \sim \mathcal{N}(0, (1-\bar{\alpha}_t) I)$, or $\sqrt{1-\bar{\alpha}_t}\, z$ where $z \sim \mathcal{N}(0, I)$

Now $x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, z$, $z \sim (0, I)$

Fig. 13. Handwritten derivation showing the forward diffusion process $x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}\epsilon$.

Prove equivalence between the two forms of the DDPM reverse diffusion process:

(1) $\hat{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(x_t + (1-\bar{\alpha}_t)\, s_\theta(x_t, t)\right)$

$$x_{t-1} = \frac{\sqrt{\alpha_t}\,(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\,(1-\alpha_t)}{1-\bar{\alpha}_t}\,\hat{x}_0 \quad \longleftrightarrow \quad (2)\; x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t + (1-\alpha_t)\, s_\theta(x_t, t)\right)$$

Let $x_{t-1} = \frac{\sqrt{\alpha_t}\,(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}\hat{x}_0$, where $\hat{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}\left[x_t + (1-\bar{\alpha}_t) s_\theta(x_t, t)\right)$

we know $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, so $\bar{\alpha}_t = \alpha_t \bar{\alpha}_{t-1}$

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

$= \frac{\alpha_t - \bar{\alpha}_t}{\sqrt{\alpha_t}(1-\bar{\alpha}_t)} x_t + \frac{\sqrt{\bar{\alpha}_t}\,(1-\alpha_t)}{\sqrt{\alpha_t}\,(1-\bar{\alpha}_t)} \cdot \frac{1}{\sqrt{\bar{\alpha}_t}}\left(x_t + (1-\bar{\alpha}_t) s_\theta(x_t, t)\right)$

$= x_t\left(\frac{\alpha_t - \bar{\alpha}_t}{\sqrt{\alpha_t}(1-\bar{\alpha}_t)} + \frac{\sqrt{\bar{\alpha}_t}\,(1-\alpha_t)}{\sqrt{\bar{\alpha}_t}\sqrt{\alpha_t}\,(1-\bar{\alpha}_t)}\right) + \frac{\sqrt{\bar{\alpha}_t}\,(1-\alpha_t)}{\sqrt{\bar{\alpha}_t}\sqrt{\alpha_t}\,(1-\bar{\alpha}_t)}(1-\bar{\alpha}_t)\, s_\theta(x_t, t)$

$= x_t\left(\frac{1}{\sqrt{\alpha_t}}\right)\left(\frac{\alpha_t - \bar{\alpha}_t + (1-\alpha_t)}{1-\bar{\alpha}_t}\right) + \left(\frac{1}{\sqrt{\alpha_t}}\right)(1-\alpha_t)\, s_\theta(x_t, t)$

$= \frac{1}{\sqrt{\alpha_t}}\left(\frac{1-\bar{\alpha}_t}{1-\bar{\alpha}_t} x_t + (1-\alpha_t)\, s_\theta(x_t, t)\right)$

This leaves $x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t + (1-\alpha_t) s_\theta(x_t, t)\right)$, as desired.

Fig. 14. Handwritten derivation showing equivalence between two forms of the DDPM reverse diffusion process.

Prove equivalence between noise-based and score-based reverse diffusion process formulations:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t,t)\right) \Leftrightarrow x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t + (1-\alpha_t)s_\theta(x_t,t)\right)$$

Let $x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t + (1-\alpha_t)s_\theta(x_t,t)\right)$

From the VP forward diffusion process, $x_t = \sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\,\epsilon$, $\epsilon \sim N(0,I)$

From Tweedie's formula, $\hat{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t + (1-\bar{\alpha}_t)\nabla_{x_t}\log p_t(x_t))$,

where $s_\theta(x_t,t) = \nabla_{x_t}\log p(x_t)$

Since $x_t = \sqrt{\bar{\alpha}_t}\cdot x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ and $\epsilon \sim N(0,I)$, this implies $x_t \sim N(\sqrt{\bar{\alpha}_t}\,x_0, (1-\bar{\alpha}_t)I)$

Then from the standard gaussian pdf,

$$\log p(x_t) = -\frac{1}{2(1-\bar{\alpha}_t)}\|x_t - \sqrt{\bar{\alpha}_t}\,x_0\|^2 + c$$

$$\nabla_{x_t}\log p(x_t) = -\frac{1}{2(1-\bar{\alpha}_t)}(2)(x_t - \sqrt{\bar{\alpha}_t}\,x_0) = -\frac{x_t - \sqrt{\bar{\alpha}_t}\,x_0}{1-\bar{\alpha}_t}$$

$x_t - \sqrt{\bar{\alpha}_t}\,x_0 = \sqrt{1-\bar{\alpha}_t}\,\epsilon$, so $s_\theta(x_t,t) = -\frac{\sqrt{1-\bar{\alpha}_t}\,\epsilon}{1-\bar{\alpha}_t} = -\frac{\epsilon}{\sqrt{1-\bar{\alpha}_t}} = -\frac{\epsilon_\theta(x_t,t)}{\sqrt{1-\bar{\alpha}_t}}$

now $x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t + (1-\alpha_t)\left(\frac{-\epsilon_\theta(x_t,t)}{\sqrt{1-\bar{\alpha}_t}}\right)\right) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t,t)\right)$

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t,t)\right), \text{ as desired.}$$

Fig. 15. Handwritten derivation showing equivalence between score-based and noise-based formulations of reverse diffusion process.