Performance of SDEdit, ScoreALD, and DPS for Deconvolution and Inpainting

Braden Becker, Lord Crawford

Abstract—Image deconvolution and inpainting are challenging inverse problems that have benefited greatly from increasingly powerful diffusion models. This paper presents three methods of increasing performance for deconvolution and inpainting: SDEdit, ScoreALD, and DPS. All methods utilize the same pre-trained diffusion model and are tested on a standardized set of test images. Each method has different parameters to tune performance, and this paper presents the optimal parameters found for each method. The results of both deconvolution and inpainting for each method on the test images were evaluated using both PSNR and LPIPS. DPS was found to be the most performant across all tests and metrics and was followed by ScoreALD and SDEdit, respectively.

Index Terms—Deconvolution, Inpainting, Computational Imaging

1 Introduction

The introduction of diffusion models has made many advances in computational imagery possible. In particular, they have allowed for the fast, accurate, and cheap creation of synthetic images. However, many applications require the denoising, restoration, or modification of existing images. These problems can still benefit from diffusion models; however, they require additional methods to make use of them. This paper focuses on two of these inverse problems in particular: deconvolution and inpainting. Deconvolution is the process of removing noise or blur from an image. Inpainting is the process of filling in the empty space of, or adding to, an existing image. In each case, the ground truth image cannot be restored perfectly as information has been lost. The aim is to create the most accurate image possible with the lowest computational cost. Three methods were tested in this research: SDEdit, ScoreALD, and DPS. All methods were tested using the same pre-trained diffusion model. The performance of each method was evaluated by comparing the PSNR and LPIPS values on the same ground truth image set. The performance of each model is also discussed in relation to its computational complexity. Edge devices in particular have a need for fast image processing, and so computational cost must be considered when deciding which algorithm to implement. The research made clear a hierarchy of performance among the three algorithms as well as a performance and cost tradeoff.

2 Related Work

This paper presents several experiments on solving inverse problems using a pre-trained diffusion model. The methods implemented are taken from across existing literature: SDEdit [1], ScoreALD [2], and DPS [3]. Starter code for implementation and guidance was provided by the EE367 Computational Imaging teaching team at Stanford University.

Ho et al. [4] introduced diffusion models as generative models that learn to reverse a gradual noising process, capturing rich priors over natural images. Our work builds on their variance-preserving formulation using a model pre-trained on the FFHQ dataset. SDEdit [1] was among the first to repurpose such a model for image restoration without task-specific retraining. ScoreALD [2], originally developed for compressed sensing MRI, injected measurement gradients at each reverse diffusion step to guide outputs toward the observed data. DPS [3] improved upon this by computing gradients through Tweedie's denoised estimate, improving stability and extending support to nonlinear forward models.

3 Methods

3.1 Diffusion Model Background

We adopt the variance-preserving (VP) forward process from Ho et al. [4], which corrupts a clean image $\mathbf{x}_0$ via the Markov chain

$$\mathbf{x}_t = \sqrt{1-\beta_t}\,\mathbf{x}_{t-1} + \sqrt{\beta_t}\,\mathbf{z}_{t-1}, \quad \mathbf{z}_{t-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{1}$$

where β_t is the noise schedule. Setting $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, one can show by induction (Appendix A) that the marginal at any step t is

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\,\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\,\mathbf{z}, \quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{2}$$

allowing direct noising to any level without simulating the full chain. The pretrained model predicts the score $\mathbf{s}_\theta(\mathbf{x}_t, t) = \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)$, which is related to the noise predictor $\boldsymbol{\epsilon}_\theta$ of Ho et al. [4] via Tweedie's formula (Appendix A):

$$\mathbf{s}_\theta(\mathbf{x}_t, t) = -\frac{\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{1-\bar{\alpha}_t}}. \tag{3}$$

3.2 Unconditional Sampling with Diffusion Models

To generate images unconditionally, we chain 1,000 denoising steps starting from pure noise. At each timestep, we apply the reverse diffusion process by first computing a denoised estimate and then updating the sample. Gradually, the noise sample transforms into a novel image that wasn't observed in training.

3.3 Posterior Sampling with Diffusion Models for Inverse Problems

We evaluate three approaches that use diffusion models as image priors to solve inverse problems. SDEdit [1] partially noises the degraded measurement to an intermediate timestep and runs standard reverse diffusion from that point. ScoreALD [2] augments each reverse step with a likelihood gradient to steer the trajectory toward measurement-consistent solutions. DPS [3] computes likelihood gradients through the denoised estimate and normalizes them for stable convergence, supporting both linear and nonlinear forward operators.

4 Experiments

Each algorithm was implemented in Python using the standard Torch library. Four permutations of tests were run: deconvolution box, deconvolution random, inpainting box, and inpainting random. Each of the deconvolution tests had Gaussian noise added in either a set boxed area or globally. Each of the inpainting tests followed the same pattern. Both the pre-trained Diffusion model and the test images were kept constant for all tests. Each test was evaluated visually, as well as by calculating the PSNR and LPIPS of the restored image.

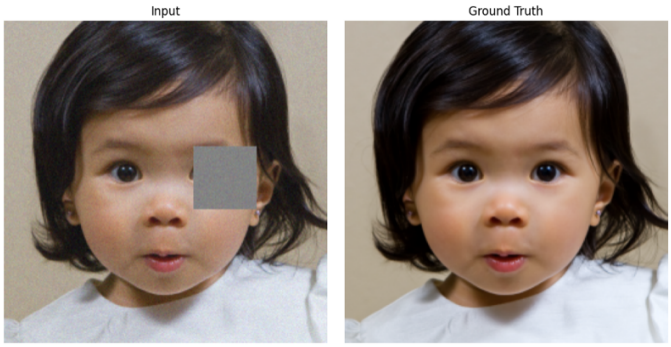

Fig. 1. An example ground truth image and test image with occluded box

Fig. 2. Comparison of inpainting with DPS, ScoreALD, and SDEdit

5 Discussion

The test achieving the highest PSNR of 34.268 and lowest LPIPS of 0.022 used DPS and was tasked with inpainting within a box. The least performant combination was SDEdit attempting to inpaint random noise, resulting in a PSNR of 14.583 and an LPIPS of 0.333. DPS outperforms ScoreALD and SDEdit across all tests and configurations.

Fig. 3. Deconvolution of random noise by DPS, PSNR=28.864, LPIPS=0.066

TABLE 1
Quantitative Results: PSNR (dB) and LPIPS for Each Method and Task

	SDEdit		ScoreALD		DPS	
	PSNR ↑	LPIPS ↓	PSNR ↑	LPIPS ↓	PSNR ↑	LPIPS ↓
Deconv Box	20.296	0.184	25.325	0.128	**28.662**	**0.057**
Deconv Random	20.551	0.212	25.065	0.140	**28.864**	**0.066**
Inpaint Box	19.837	0.223	27.262	0.088	**34.268**	**0.022**
Inpaint Random	14.583	0.333	13.490	0.964	**15.984**	**0.224**

There were several parameters across methods that allowed for additional tuning of the results. ScoreALD uses an annealing method to scale the magnitude of the steps in proportion to the proximity of the manifold. The ideal parameters of the annealing method are specific to deconvolution or inpainting respectively. In contrast DPS uses a scaling parameter to meter the relationship between the ground truth and the output of the diffusion model. If this factor is too high the output image will retain the global structure but have high frequency noise, too low and the output will hallucinate major features. This scale factor was adjusted automatically depending on the type of test being run.

Overall we find that DPS is a more robust choice for solving these inverse problems. The computational cost is significant, but on modern hardware does not represent an insurmountable challenge.

Acknowledgments

Starter code and guidance for this project was provided by the teaching team of EE367: Gordon Wetzstein and Sonia Kim.

References

[1] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon, "SDEdit: Guided image synthesis and editing with stochastic differential equations," in *International Conference on Learning Representations (ICLR)*, 2022.

[2] A. Jalal, M. Arvinte, G. Daras, E. Price, A. G. Dimakis, and J. Tamir, "Robust compressed sensing MRI with deep generative priors," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[3] 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.

[4] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

APPENDIX A
MATHEMATICAL DERIVATIONS

Derivation 1: Closed-Form Marginal of the VP Forward Process

Claim. Given the one-step forward process with $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, the marginal satisfies $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\,\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\,\mathbf{z}$, $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Proof. By induction. The base case $t = 1$ holds directly since $\bar{\alpha}_1 = \alpha_1$. For the inductive step, assume $\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\,\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}}\,\boldsymbol{\epsilon}$. Substituting into the one-step process:

$$\mathbf{x}_t = \sqrt{\alpha_t \bar{\alpha}_{t-1}}\,\mathbf{x}_0 + \underbrace{\sqrt{\alpha_t(1-\bar{\alpha}_{t-1})}\,\boldsymbol{\epsilon} + \sqrt{1-\alpha_t}\,\mathbf{z}'_t}_{\sim \mathcal{N}(\mathbf{0},\, \sigma^2 \mathbf{I})}.$$

The combined noise variance is $\sigma^2 = \alpha_t(1-\bar{\alpha}_{t-1}) + (1-\alpha_t) = 1 - \bar{\alpha}_t$, completing the induction. □

Derivation 2: Equivalence of the Two Reverse Diffusion Forms

Claim. Substituting $\hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t + (1-\bar{\alpha}_t)\mathbf{s}_\theta)$ into the posterior mean update

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

yields $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t + (1-\alpha_t)\,\mathbf{s}_\theta(\mathbf{x}_t, t))$.

Proof. Using $\bar{\alpha}_t = \alpha_t \bar{\alpha}_{t-1}$, the coefficient of $\mathbf{x}_t$ after substitution simplifies to:

$$\frac{\alpha_t(1-\bar{\alpha}_{t-1}) + (1-\alpha_t)}{\sqrt{\alpha_t}(1-\bar{\alpha}_t)} = \frac{1-\bar{\alpha}_t}{\sqrt{\alpha_t}(1-\bar{\alpha}_t)} = \frac{1}{\sqrt{\alpha_t}},$$

and the $\mathbf{s}_\theta$ term contributes $(1-\alpha_t)/\sqrt{\alpha_t}$, giving the result. □

Derivation 3: Equivalence of Score and Noise Parameterizations

Claim. The score and noise predictors satisfy $\mathbf{s}_\theta(\mathbf{x}_t, t) = -\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)/\sqrt{1-\bar{\alpha}_t}$, so that

$$\frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t + (1-\alpha_t)\,\mathbf{s}_\theta) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\,\boldsymbol{\epsilon}_\theta\right).$$

Proof. From Tweedie's formula applied to the VP marginal, the true score is $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) = -\boldsymbol{\epsilon}/\sqrt{1-\bar{\alpha}_t}$. The learned score therefore satisfies $\mathbf{s}_\theta = -\boldsymbol{\epsilon}_\theta/\sqrt{1-\bar{\alpha}_t}$, and direct substitution yields the claim, recovering Algorithm 2 of Ho et al. [4]. □