Learning to Sample in Computational Imaging: Measurement Acquisition and Posterior Estimation

Thesis by
Zihui Wu

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2026
Defended July 8th, 2025

Zihui Wu
ORCID: 0000-0002-7622-3548

ACKNOWLEDGEMENTS

As I look back on my five years of Ph.D. study at Caltech, I am deeply grateful to a long list of people who made this journey an unforgettable experience.

First and foremost, I would like to sincerely thank my advisor, Prof. Katie Bouman, for taking me as her student and giving me the opportunities to work on so many exciting projects. Without her guidance and mentorship, this thesis would not have been possible. Katie believed in me even when I doubted myself, and her perseverance pushed me to achieve what I once thought impossible. Working with her has shaped me both academically and personally, and I am truly a better version of myself because of her mentorship.

I am also indebted to my thesis committee members: Prof. Yisong Yue, Prof. Pietro Perona, and Dr. Yang Song. Even though Yisong is not my official advisor, many of the works in this thesis were done with him. His advice and support—both on research and on navigating Ph.D. life—have been invaluable. I am grateful to Pietro for cultivating a vibrant vision and imaging community and for his insightful feedback during group meetings. I also appreciate the generosity of Dr. Song for serving on my committee. This thesis owes much to the knowledge and inspiration I drew from his insightful papers and blog posts on diffusion models.

I would also like to thank my undergraduate research advisor, Prof. Ulugbek Kamilov, back at WashU and my mentor, Dr. Yu Sun, now a professor at Johns Hopkins University. I first met Prof. Kamilov in his class on optimization during my sophomore year, where his Ph.D. student Yu was a teaching assistant. The class sparked my curiosity and led me to join Prof. Kamilov's research group, where they taught me everything from coding and running experiments to writing papers and making figures. Their mentorship laid the foundation for my research career, and without them, I would not have pursued a Ph.D. in computational imaging. It was also a pleasure to work with Yu again during his postdoc in Katie's group, which helped me transition smoothly between the early and later stages of my Ph.D.

I am grateful for the opportunities to collaborate with many brilliant researchers during my Ph.D. Thank you to Prof. Adrian Dalca, Prof. Robert Frost and Prof. Andre van de Kouwe from MGH for hosting me over two summers and teaching me MRI from the basics, which was instrumental to translating my research on MRI in Part I into real-world impact. Thank you to Dr. David Ouyang from Cedars Sinai

for offering me an opportunity to learn about medical imaging in clinical practice, which was eye-opening to me. Thank you to Prof. Liam Connor for mentoring me on radio interferometry and introducing me to the beauty of astronomy. Thank you to Prof. Al Barr for attending both my candidacy exam and thesis defense and for offering encouragement and constructive feedback. Additionally, I would like to thank Dr. Tianwei Yin for collaborating with me on MRI projects—his motivation and technical skill, even as an undergraduate back then, were inspiring and set a high bar for my own work. I am also grateful to Dr. Yu Sun, Hongkai Zheng, Bingliang Zhang, and Wenda Chu—not only as collaborators on the diffusion model projects for Part II but also as friends. I learned so much from our stimulating discussions and was lucky to share this journey with them.

This thesis was made possible by generous support from the Kortschak Scholars Program, the Amazon AI4Science Fellowship, the National Science Foundation, the National Institutes of Health, the Heritage Medical Research Institute, Beyond Limits, Rockley Photonics, and Schmidt Sciences.

Beyond research, I have been fortunate to form deep and lasting friendships at Caltech. I am grateful to my lab mates in the Computational Cameras (Bouman) Group and the Caltech Vision (Perona) Lab, as well as friends across the CMS department and the broader Caltech community. My social life at Caltech was enriched particularly by friends from the Wednesday-Friday basketball group, where I very much enjoyed both the competitive basketball games and the camaraderie. I also appreciate the help from our department and group admins—Maria Lopez, Roberta Carvalho, Monica Nolasco, and Nuvia Alvarez—for their efforts in making my Ph.D. life smooth outside of research.

I am immensely grateful to my parents, grandparents, and other family members for their unwavering love and care throughout my Ph.D. years. Their support has been a constant source of strength during difficult times and will continue to guide me in the years ahead.

Finally, I would like to express my deepest gratitude to my girlfriend, Xinyi Wu. Her love, encouragement, and companionship have made my Ph.D. years truly cherishable. She has been by my side through every high and low, offering support in so many ways that I never expected but greatly appreciated—both personally and academically. I cannot imagine completing this journey without her, and I look forward to all the chapters of life ahead together.

ABSTRACT

Many problems in science and engineering require visualizing objects that are not directly observable—such as black holes that are millions of light-years away from the Earth or internal anatomical structures hidden within the human body. Computational imaging is a powerful paradigm that combines sensor design with advanced computational algorithms to make the invisible visible. The typical computational imaging pipeline involves first collecting indirect measurements of the target object and then solving a reconstruction problem. This thesis focuses on two core challenges about *sampling* along this pipeline: (1) optimizing the sampling process for measurement acquisition, and (2) sampling the posterior distribution of possible reconstructions given noisy measurements.

The first part of the thesis investigates how to design adaptive and task-specific acquisition strategies for computational imaging systems, with a focus on compressed sensing magnetic resonance imaging (CS-MRI). We propose a sequential sampling method that learns to select measurements in multiple stages, and an approach that tailors sampling patterns for specific downstream tasks such as region-of-interest reconstruction, segmentation, and classification. These methods enable a better selection of measurements taken during acquisition, leading to improved performance compared to conventional baselines. We have also implemented our learned sequences on a real MRI scanner and verified their improvement in practice.

The second part of the thesis develops a principled framework for posterior sampling using diffusion models (DMs)—a state-of-the-art class of generative models. By revealing a key connection between DMs and the Split Gibbs Sampling, we introduce a posterior sampling method that rigorously incorporates pre-trained DMs as image priors for solving inverse problems, which exhibits strong performance on a variety of applications. We then show that this framework can be naturally extended into a series of instantiations for solving more general inverse problems, addressing topics like text conditioning, video inverse problems, non-differentiable forward models, and discrete-space sampling. We also present a comprehensive benchmark for systematically evaluating state-of-the-art DM-based posterior estimation methods.

By leveraging machine learning to address challenges in both data acquisition and posterior estimation, this thesis provides new possibilities for building more intelligent and reliable imaging systems across science and engineering.

PUBLISHED CONTENT AND CONTRIBUTIONS

[1] Brady Bhalla*, Zachary Huang*, Elyes Serghine*, Bingliang Zhang, Zihui Wu, and Katherine L. Bouman. "Text-Guided Image Restoration via a Unified Plug-and-Play Diffusion Framework." In: *Computational Cameras and Displays (CCD) Workshop, CVPR 2025* (2025).
Z.W. mentored the first authors, led the conception of the project, designed and supervised all the experiments, and led the writing (especially the figures) of the manuscript.

[2] Wenda Chu, Zihui Wu, Yifan Chen, Yang Song, and Yisong Yue. *Split Gibbs Discrete Diffusion Posterior Sampling*. 2025. arXiv: 2503.01161 [cs.LG]. URL: https://arxiv.org/abs/2503.01161.
Z.W. led the theoretrical analysis and contributed to writing the manuscript.

[3] Austin Wang, Hongkai Zheng, Zihui Wu, Ricardo Baptista, Daniel Zhengyu Huang, and Yisong Yue. "Ensemble Kalman Sampling and Diffusion Prior in Tandem: A Split Gibbs Framework." In: *Frontiers in Probabilistic Inference: Learning Meets Sampling, ICLR 2025*. 2025. URL: https://openreview.net/forum?id=3DfCxd0yxo.
Z.W. participated in the conception and theoretical part of the project, and contributed to writing the manuscript.

[4] Bingliang Zhang*, Zihui Wu*, Berthy T. Feng, Yang Song, Yisong Yue, and Katherine L. Bouman. *STeP: A Framework for Solving Scientific Video Inverse Problems with Spatiotemporal Diffusion Priors*. 2025. arXiv: 2504.07549 [cs.CV]. URL: https://arxiv.org/abs/2504.07549.
Z.W. participated in the conception of the project, prepared the dataset for dynamic MRI, and co-led the writing (especially the figures) of the manuscript.

[5] Hongkai Zheng*, Wenda Chu*, Bingliang Zhang*, Zihui Wu*, Austin Wang, Berthy Feng, Caifeng Zou, Yu Sun, Nikola Borislavov Kovachki, Zachary E Ross, Katherine Bouman, and Yisong Yue. "InverseBench: Benchmarking Plug-and-Play Diffusion Models for Scientific Inverse Problems." In: *The Thirteenth International Conference on Learning Representations*. 2025. URL: https://openreview.net/forum?id=U3PBITXNG6.
Z.W. co-led the conception of the project, conducted all the experiments on compressed sensing MRI, and led the writing (especially the figures) of the manuscript.

[6] Zihui Wu, Yu Sun, Yifan Chen, Bingliang Zhang, Yisong Yue, and Katherine Bouman. "Principled Probabilistic Imaging Using Diffusion Models as Plug-and-Play Priors." In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024. URL: https://openreview.net/forum?id=Xq9HQf7VNV.

Z.W. led the conception of the project, conducted all the experiments, proved the main theorem, and wrote the manuscript.

[7] Zihui Wu, Tianwei Yin, Yu Sun, Robert Frost, Andre van der Kouwe, Adrian V. Dalca, and Katherine L. Bouman. "Learning Task-Specific Strategies for Accelerated MRI." In: *IEEE Transactions on Computational Imaging* 10 (2024), pp. 1040–1054. DOI: 10.1109/TCI.2024.3410521.
Z.W. led the conception of the project, conducted all the experiments, and wrote the manuscript.

[8] Tianwei Yin*, Zihui Wu*, He Sun, Adrian V. Dalca, Yisong Yue, and Katherine L. Bouman. "End-to-End Sequential Sampling and Reconstruction for MRI." In: *Proceedings of Machine Learning for Health*. Vol. 158. Proceedings of Machine Learning Research. PMLR, Dec. 2021, pp. 261–281. URL: https://proceedings.mlr.press/v158/yin21a.html.
Z.W. participated in the conception of the project, conducted the experiments, and led the writing (especially the figures) of the manuscript.

$*$ indicates co-first authors.

TABLE OF CONTENTS

Acknowledgements . . . iii
Abstract . . . v
Published Content and Contributions . . . vi
Table of Contents . . . vii
List of Illustrations . . . xi
List of Tables . . . xxviii
Chapter I: Introduction . . . 1
1.1 Computational Imaging as Inverse Problems . . . 2
1.2 Computational Imaging Pipeline and Some Key Challenges . . . 2
1.3 Thesis Overview . . . 4

I Sampling: Measurement Acquisition 6
Chapter II: Overview and Preliminaries . . . 7
2.1 Magnetic Resonance Imaging (MRI) . . . 7
2.2 Compressed Sensing MRI (CS-MRI) . . . 7
2.3 Part Outline . . . 9
Chapter III: Learning Sequential Sampling and Reconstruction Strategies . . . 10
3.1 Introduction . . . 10
3.2 Related Work . . . 12
3.3 Method . . . 14
3.4 Experiments . . . 19
3.5 Conclusion . . . 23
Chapter IV: Learning End-to-End Strategies for the Downstream Task of Interest 25
4.1 Introduction . . . 26
4.2 Related Work . . . 27
4.3 Method . . . 28
4.4 Experiments on Large-Scale Datasets . . . 33
4.5 Validation on an Experimentally Collected Out-of-Distribution Dataset 40
4.6 Ablation Studies . . . 43
4.7 Limitations . . . 46
4.8 Conclusion . . . 47

II Sampling: Posterior Estimation 56
Chapter V: Overview and Preliminaries . . . 57
5.1 Bayesian Inverse Problems . . . 57
5.2 Diffusion Models . . . 59
5.3 Solving Inverse Problems with Diffusion Models . . . 60

5.4 Part Outline 62
Chapter VI: PnP-DM: A Principled Framework for Posterior Estimation Using Diffusion Models 64
6.1 Introduction 64
6.2 Preliminaries 66
6.3 Method 68
6.4 Convergence Analysis 71
6.5 Experiments 74
6.6 Conclusion 82
Chapter VII: Beyond PnP-DM: Towards a Unified Framework for General Posterior Estimation 84
7.1 Limitations of PnP-DM 84
7.2 A Unified Diffusion-Based Posterior Estimation Framework 85
7.3 Thrust 1: Incorporating More Information as Prior 86
7.4 Thrust 2: Harnessing Higher Dimension for Video Inverse Problems 97
7.5 Thrust 3: Accommodating Black-Box Forward Models 110
7.6 Thrust 4: Tackling Inverse Problems in Discrete Spaces 116
7.7 Conclusion 122
Chapter VIII: InverseBench: Benchmarking Diffusion-Based Methods for Scientific Inverse Problems 124
8.1 Introduction 124
8.2 Plug-and-Play Diffusion Priors for Inverse Problems 126
8.3 InverseBench 128
8.4 Experiments 131
8.5 Conclusion 137
Chapter IX: Conclusion 139
9.1 Key Takeaways 139
9.2 Futures Directions 141
9.3 Concluding Remarks 142

III Appendices **143**

Appendix A: Appendix for Chapter 3 144
A.1 Model Architectures 144
A.2 Baseline Details 145
A.3 Further Analyses 147
A.4 Additional Reconstruction Examples 148
Appendix B: Appendix for Chapter 4 152
B.1 Implementation Details 152
B.2 Subsampling Setup and Implementation 156
B.3 Additional Validation on Out-of-Distribution Data 157
B.4 Additional Results 157
Appendix C: Appendix for Chapter 6 160
C.1 Theory 160
C.2 Inverse Problem Setup 166

C.3 Technical Details of PnP-DM . . . 171
C.4 Implementation Details of Baseline Methods . . . 178
C.5 Additional Related Works . . . 179
C.6 Additional Experimental Results . . . 181
Appendix D: Appendix for Chapter 7 . . . 189
D.1 Appendix for Section 7.3 . . . 189
D.2 Appendix for Section 7.4 . . . 191
D.3 Appendix for Section 7.5 . . . 201
D.4 Appendix for Section 7.6 . . . 217
Appendix E: Appendix for Chapter 8 . . . 228
E.1 Tables of Main Results . . . 228
E.2 Inverse Problem Details . . . 231
E.3 Pre-Trained Diffusion Model Details . . . 236
Bibliography . . . 239

LIST OF ILLUSTRATIONS

Number *Page*

1.1 **Pipeline of computational imaging.** In Stage 1 (Measurement Acquisition), a physical sensor acquires indirect, noisy measurements of some unknown ground truth $\boldsymbol{x}_0$. Mathematically, this can be modeled by $\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}_0) + \boldsymbol{n}$ where $\boldsymbol{n}$ represents measurement noise. In Stage 2 (Image Reconstruction), an algorithm $\mathcal{A}^\dagger(\cdot)$ is designed to obtain a reconstruction $\widehat{\boldsymbol{x}}$ from the measurements. 3

3.1 **We propose a sequential sampling and reconstruction co-design framework for accelerated MRI that adapts to a target during acquisition.** Here, we visualize the sampling policy and final reconstruction of rotated knees in a single-coil imaging setting with 8× acceleration (8× subsampling). The first four columns show the cumulative k-space measurements selected by the proposed learned sampler (pink) in acquisition steps 1 through 4 (during a 4-step acquisition). The fifth column shows the final image recovered by the proposed learned reconstructor, and the last column is the ground truth. This example illustrates how our model has learned to adapt to different k-space distributions: the final sampling patterns in the fourth column contain visible directional structure that aligns with the k-space power spectrum. Rotated anatomical images, such as these rotated knee images, were *not* included in the training set (or quantitatively evaluated test set). 11

3.2 **Overview of the proposed sequential sampling framework.** Low-frequency samples are pre-selected and measured in k-space. The subsampled k-space is transformed into a zero-filled image, which is fed into a reconstructor $\mathsf{R}_{\boldsymbol{\theta}}(\cdot)$ to produce an intermediate image reconstruction (Equation (3.1)). The intermediate reconstruction and measurements are passed into a sampler network $\mathsf{S}_{\boldsymbol{q}}(\cdot)$, which outputs a discrete probability distribution representing suggested samples for the next iteration. An action is sampled from this distribution (Equation (3.2)), and the corresponding k-space measurements are acquired. The sampling and reconstruction process is repeated for T steps. The sampler and reconstructor are neural networks learned via end-to-end training with a loss on the final reconstructed image. Weights are shared across all T acquisition steps. 15

3.3 **Visualizations of two types of k-space sampling patterns: 1D line sampling and 2D point sampling.** White regions are sampled from a uniform distribution over the space of possible actions. The center low-frequency samples are pre-selected in all experiments before any further sampling. DC corresponds to the $(0, 0)$ frequency. 17

3.4 **Visualizations of example reconstructions with an 4× acceleration for 1D line sampling.** Two zoomed-in image patches are shown along with the cumulative k-space measurements selected by each policy. Our sequential approach often provides more accurate reconstructions with detailed local structures. More visualizations are included in the Appendix A.4. 19

3.5 **Histograms of pair-wise SSIM differences between our sequential models and LOUPE [12] on all 1,851 test images.** Positive numbers indicate improvement over LOUPE. The results are acquired by averaging three runs of 4× accelerated 2D point subsampling. More sequential steps lead to a bigger advantage over LOUPE, with the 4-step sequential model outperforming LOUPE on 96.96% of samples. This performance pattern holds for the 1D line scenario and other acceleration factors as well, as shown in Appendix A.3. More quantitative results are given in Table 3.3. 19

3.6 **Comparison between our sequential model and the LOUPE model on the fastMRI knee test set.** Our sequential model outperforms LOUPE for all acceleration ratios with an improvement comparable to 25% of the benefit of doubling the number of k-space measurements. The performance of our sequential model in the 1D line sampling case significantly outperforms LOUPE but plateaus after 2 sequential sampling steps, possibly due to the restricted action space of 1D line sampling. 20

3.7 **Histograms of pair-wise SSIM comparison on all 1,851 test images with a different number of sequential steps (T), using 2D point sampling with a 4× acceleration factor.** The relative error between the 4-step and 1-step (left) or 2-step(right) demonstrates that additional sequential steps help to boost performance, but with diminishing returns as T increases. 21

4.1 **Comparison between (a) traditional CS-MRI, (b) a naïve approach to task-specific CS-MRI, and (c) the proposed TACKLE framework.** Compared with panel (a) which separately deals with reconstruction and task prediction, panel (b) is a simple extension of co-design methods for solving downstream tasks by adding a learnable mapping from measurements to task predictions. However, this naïve approach leads to a suboptimal performance and can even lead to a worse task prediction accuracy, as shown in the example above. On the other hand, we introduce TACKLE for effectively learning task-specific CS-MRI strategies. TACKLE is first pre-trained for generic reconstruction, and then all three modules are fine-tuned for a more specific downstream task. We find that this training schedule allows TACKLE to robustly learn generalizable task-specific strategies. In the above knee segmentation example, all three approaches are trained with the same architectures for the reconstructor (second module) and predictor (third module). Nevertheless, TACKLE significantly outperforms the two baseline approaches. 48

4.2 **Block diagram of the proposed framework TACKLE and a summary of the investigated datasets and settings.** TACKLE uses a task-specific loss to jointly optimize a sampler, a retriever, and an optional predictor, ranging from scanner-level sampling to human-level diagnosis. A summary of the investigated settings is presented in the bottom left panel. FSE, GRE, DESS, and FLAIR stand for fast spin echo, gradient echo, double-echo steady-state, and fluid-attenuated inversion recovery, respectively. We comprehensively investigate multiple CS-MRI tasks on a variety of common MRI settings with six datasets. 49

4.3 **Visual examples of two Meniscus Tear samples reconstructed by different methods in the 16× acceleration single-coil setting.** For each reconstruction, the full-FOV PSNR is labeled in white, and the local PSNR for the ROI is in orange. Note how $\text{TACKLE}_{\text{ROI}}$ recovers the structure and details of the ROI more accurately than the two baselines, as indicated by the red arrows. The better recovery of $\text{TACKLE}_{\text{ROI}}$ over the ROI leads to a more accurate diagnosis of the Meniscus Tear. We emphasize that the location of the ROI is not an input to any of these models and is only used for evaluating the accuracy of each method on the region that contains the pathology. . 49

4.4 **Comparison of a subsampling PSF optimized for full-FOV reconstruction and another optimized for the reconstruction of meniscus tear (MT) ROIs.** Optimizing for MT ROI reconstruction leads to around 40% improvement on the vertical resolution in terms of the *full width at half maximum (FWHM)*, as shown by the PSF profiles in the bottom panel. This improved vertical resolution leads to a better reconstruction of the meniscus that has horizontal anatomy. 50

4.5 **Box plots of the knee tissue segmentation results under 16× (a) and 64× (b).** Within the rectangle between each pair of methods, the top number is the percentage of samples that get improved, and the bottom number is the p-value given by the paired samples t-test. A higher percentage and a lower p-value indicate a more significant improvement. We also provide the 95% confidence intervals for all methods below their names. For both acceleration ratios, $\text{TACKLE}_{\text{seg.}}$ outperforms other baselines in terms of all the statistical measures. 50

4.6 **Comparison of segmentation results under 16× acceleration on one sample of the SKM-TEA dataset.** We show the input of the predictor in the first row, a zoom-in on the region that contains the tissues to be segmented in the second row, and the output of the predictor in the third row. Note that TACKLE$_{\text{seg.}}$ circumvents the typical "reconstruction" in terms of pixel-wise similarity with the ground truth image. Instead, it learns a feature map that accurately localizes the anatomy, leading to better segmentation prediction than other baselines both for this sample and on average over the test set (Table 4.2). 51

4.7 **Box plots of the brain tissue segmentation results under 16× (a) and 64× (b) accelerations.** Within the rectangle between each pair of methods, the top number is the percentage of samples that get improved, and the bottom number is the p-value given by the paired samples t-test. A higher percentage and a lower p-value indicate a more significant improvement. We also provide the 95% confidence intervals for all methods below their names. Similar to the knee segmentation results, the proposed method TACKLE outperforms other baselines in terms of all the statistical measures for both acceleration ratios. 52

4.8 **Comparison of segmentation results under 16× acceleration on one sample from the OASIS dataset.** Similar to the knee segmentation results, TACKLE$_{\text{seg.}}$ circumvents the typical "reconstruction" in terms of pixel-wise similarity with the ground truth image. Instead, it learns an anatomically accurate feature map, which enables better segmentation prediction than other baselines both for this sample and on average over the test set (Table 4.3). The zoom-in panels highlight a region where TACKLE$_{\text{seg.}}$ more accurately predicts the outline of white matter (in yellow) than other methods. This improvement leads to a more precise estimation of the thickness of the cortex (in orange), an important task for studying human cognition and neurodegeneration [10]. 53

4.9 **Confusion matrices of the classification results by LOUPE$_{\text{recon.}}$ and TACKLE$_{\text{class.}}$.** Overall, TACKLE$_{\text{class.}}$ achieves greater accuracy in terms of both classification accuracy and F_1 score than LOUPE$_{\text{recon.}}$. TACKLE$_{\text{class.}}$ also has a significantly lower number of false negatives (bottom left) compared to LOUPE$_{\text{recon.}}$, which could lead to more patients receiving early treatment. 53

4.10 **Per-slice difference histograms.** (a): TACKLE$_{\text{FOV}}$ over LOUPE$_{\text{FOV}}$ on the full-FOV reconstruction task and (b): TACKLE$_{\text{ROI}}$ over LOUPE$_{\text{FOV}}$ on the ROI-oriented reconstruction task. The 95% confidence intervals are given in the top left corner of each plot. In both cases, the vast majority of slices improve, and the p-values given by the paired samples t-test are highly significant. 54

4.11 **Reconstruction comparison of two samples in the experimentally collected dataset (top: from subject 1; bottom: from subject 2) by different methods under 4× acceleration.** The sampling mask, a zoom-in on the ROI, and the error map are presented for each method. By sampling more frequencies along the vertical direction in k-space, TACKLE$_{\text{ROI}}$ has a higher vertical resolution in the image space and thus outperforms other baselines optimized for full-FOV reconstruction on the ROIs with directional anatomical structure. . . 55

4.12 **Reconstruction comparison between the implemented prospective subsampling sequence and the retrospective subsampling sequence.** Our learned sequence can be implemented on an MRI scanner and generates images of quality indistinguishable from those recovered from retrospectively sampled data. Compared to the ground truth image, our prospectively subsampled reconstruction recovers important features around the meniscus region, which is the ROI it is trained to enhance. 55

6.1 **A schematic diagram of our method.** Our method alternates between a likelihood step that enforces data consistency and a prior step that solves a denoising posterior sampling problem by leveraging the Split Gibbs Sampler [296]. An annealing schedule controls the strength of the two steps at each iteration to facilitate efficient and accurate sampling. A crucial part of our design is the prior step, where we identify a key connection to a general diffusion model framework called the EDM [151]. This connection allows us to easily incorporate a family of state-of-the-art diffusion models as priors to conduct posterior sampling in a principled way without additional training. Our method demonstrates strong performance on a variety of linear and nonlinear inverse problems. 68

6.2 **A conceptual illustration of the non-stationary and stationary time-continuous processes as interpolations of K discretize iterations of PnP-DM.** 71

6.3 **Results on a synthetic problem with the ground truth posterior available.** PnP-DM can sample it more accurately that DPS [64]. . . 73

6.4 **Visual examples for the motion deblur problem ($\sigma_y = 0.05$).** We visualize one sample generated by each sampling algorithm. 75

6.5 **Comparison of uncertainty quantification (UQ) for the motion deblur.** Left 3 columns: absolute error ($|\widehat{\boldsymbol{x}} - \boldsymbol{x}_0|$), standard deviation (std), and absolute z-score ($|\widehat{\boldsymbol{x}} - \boldsymbol{x}_0|/\text{std}$) with the outlier pixels in red. Right column: scatter plot of $|\widehat{\boldsymbol{x}} - \boldsymbol{x}_0|$ versus std. Note that PnP-DM leads to a better UQ performance than the baselines by having the lowest percentage of outliers while avoiding having overestimated per-pixel standard deviations. 76

6.6 **Results of the Fourier phase retrieval problem.** (a) PnP-DM provides both upright and rotated reconstructions (two modes given by the invariance of the forward model to 180° rotation) with high fidelity, while the baseline methods cannot. (b) We visualize the percentages of upright and rotated reconstructions out of 90 runs for a test image with two samples for each orientation. 77

6.7 **Results on the black hole imaging problem with simulated data.** Due to severe noise corruption and measurement sparsity, this problem is non-convex and highly ill-posed, leading to a bi-modal posterior distribution as previously found in [269]. Here we compare our method, PnP-DM, with the DPS baseline [64]. A metric quantifying the mismatch with the observed measurements is labeled for each sample, which should be around 2 for an ideal measurement fit. Samples generated by PnP-DM exhibit two distinct modes with sharp details and a consistent ring structure, while samples given by DPS display inconsistent ring sizes and sometimes fail to capture the black hole structure entirely, with samples having poor measurement fit. 78

6.8 **Results on the black hole imaging problem with the real data for the M87 black hole from April 6$^{\text{th}}$, 2017 [71].** The posterior samples from PnP-DM contain fine-grained features that align with the prior distribution; see left for a few samples generated by the pre-trained diffusion model from the prior. Besides having high visual quality, our posterior samples accurately capture key features of the official reconstruction by EHT as well, such as the bright spot location and ring diameter. 79

6.9 **Visual examples of the DSA deconvolution problem in radio astronomy.** All images undergo a nonlinear transformation to visualize the weaker galaxies. (a): The mean image of PnP-DM (based on 20 samples) significantly outperforms the reconstruction of the classic CLEAN algorithm [134]. (b): We visualize three posterior samples and the per-pixel standard deviation map computed from all 20 samples. Zoom-in regions highlight areas with notable sample variability. 81

6.10 **Comparison of galaxy property estimation accuracy between CLEAN (blue) and PnP-DM (orange).** Each scatter plot shows predicted versus true values for semi-major axis θ_A (left), semi-minor axis θ_B (middle), and flux (right), across all detected sources. The dashed line indicates perfect prediction. PnP-DM produces estimates that lie closer to the diagonal, indicating more accurate recovery of galaxy shapes and fluxes. Notably, PnP-DM avoids the strong overestimation biases seen in CLEAN. 82

7.1 **Illustration of the prior step with different text inputs.** Starting from a noisy image $\boldsymbol{v}^{(k)}$, the prior step performs text-guided denoising through reverse diffusion to generate a cleaner sample $\boldsymbol{u}^{(k+1)}$ on the clean image manifold. The images are visualized after decoding. Different text prompts steer the denoising process toward distinct yet plausible modes of the image. 89

7.2 **Visual examples for the super-resolution (16×), gaussian deblur ($\sigma = 6$), and box inpainting tasks with matching prompts.** The proposed instantiations, DAPS and DCDP in particular, produce more detailed and perceptually coherent reconstructions, while the baseline methods suffer from artifacts and a decrease in quality. 91

7.3 **Visual examples of DAPS on the 16× super-resolution task using different prompts.** The examples show that varying descriptions lead to semantically different yet feasible reconstructions. Despite the severe degradation of the input measurements, the restored images are of high quality and closely align with the information from the provided text prompts, while still consistent with the original measurement. 94

7.4 **Visualization of the image generation process of DAPS.** The top row shows outputs from the prior steps ($\boldsymbol{u}^{(k+1)}$ for $k \in \{9, 19, 29, 39\}$), which denoise and integrate text conditioning to guide the sample toward the target distribution, while the bottom row shows outputs from the likelihood steps ($\boldsymbol{v}^{(k)}$ for $k \in \{9, 19, 29, 39\}$), which enforce data consistency with added noise. Initialization starts from pure noise, and the two processes alternate to progressively generate a realistic image. The images are visualized after decoding. 95

7.5 **Effect of prompt specificity on posterior samples generated by DAPS for 16× super-resolution.** We repeatedly generate samples using DAPS with prompts of varying specificity. As prompts become more generic, the samples exhibit greater semantic diversity—e.g., several yellow scoops in the "a plate of food" panel look like mashed potatoes, which are not seen in the more specific "a plate of desserts" panel. However, the overall diversity in appearance remains similar—i.e., the bottom two panels appear to be equally diverse at first glance. Modes with highly specific details, such as "two scoops of ice cream with a cannoli" or "a plate of macarons" (top right), are only recovered when given the corresponding prompts. These results underscore the importance of text conditioning in uncovering rare modes and improving mode coverage in posterior estimation. 96

7.6 **An overview of our proposed framework with spatiotemporal diffusion priors, STeP, for scientific video inverse problems.** Left: STeP combines the physics model of the target problem with a spatiotemporal diffusion prior that directly characterizes the video distribution. We show that such a prior can be efficiently obtained by fine-tuning a pre-trained image diffusion model with limited video data. Right: STeP can generate diverse solutions to a black hole video reconstruction problem that exhibit equally good fidelity with the measurements. 99

7.7 **A schematic comparison between prior works (top) and our STeP framework (bottom) for video inverse problems.** The bold texts highlight the key differences between them. While prior works use an image diffusion model and enforce temporal consistency using simple heuristics or warping noise with optical flow, we directly learn a spatiotemporal diffusion prior. 100

7.8 **Architecture of the spatiotemporal module.** Given a pre-trained image diffusion U-Net, we add a zero-initialized temporal module with an `ON/OFF` switch to each 2D spatial module and initialize the additive weight α to zero. Thus, it will have no effect at the start of fine-tuning and gradually learn from the video training data. The number of frames, height, and width are denoted by n_f, n_h, and n_w, respectively. The numbers of channels for input features (f_{in}) and output features (f_{out}) are denoted by n_{in} and n_{out}, respectively. 101

7.9 **Visual examples of STEP (bottom left) and baselines for black hole video reconstruction.** To facilitate analysis of the reconstructed spatiotemporal structures, we present results in three ways: (1) a single frame to illustrate spatial fidelity, (2) an x-t slice depicting temporal evolution of a vertical line to evaluate temporal consistency, and (3) the averaged optical flow visualized using the standard color scheme from [278] to assess spatiotemporal coherence jointly. Compared to baselines, STEP exhibits clearer alignment with ground truth videos across all aspects. 107

7.10 **Visual examples of STEP (bottom left) and baselines for dynamic MRI.** We visualize a representative frame along with two zoomed-in regions for each method to better illustrate spatial fidelity. Benefiting from its robust spatiotemporal prior, STEP provide reconstructions with fewer structural artifacts and temporal fluctuations, as indicated by its averaged optical flow aligning more closely with the ground truth. This demonstrates the effectiveness of our learned spatiotemporal prior in enhancing both spatial and temporal consistency. 107

7.11 **Detailed comparison on black hole video reconstruction and dynamic MRI.** Left: We compare STEP (joint) and the BCS baseline [168] by visualizing the averaged delta frames (difference images) over an expanding window. The delta frames given by STEP (joint) better align with the ground truth, indicating better temporal consistency. Right: We also compare the spatial fidelity between STEP (joint) and its variant STEP (video-only). Trained on both images and videos, STEP (joint) provide reconstructions with less spatial hallucinations compared to STEP (video-only). 108

7.12 **Consistent improvement in image-video joint fine-tuning.** We evaluate intermediate checkpoints of (a) black hole video reconstruction and (b) dynamic MRI (8×). Spatial similarity (measured by PSNR), temporal consistency (measured by d-PSNR), and measurement data fit (measured by data misfit) all show steady improvement. 108

7.13 **Comparison between running STEP with DAPS versus PNP-DM as the inference backbone.** The DAPS version of STEP exhibits significantly better spatial consistency (shown by the first frames in the first row) and temporal consistency (shown by the x-t slice in the second row). This comparison illustrates the better compatibility of DAPS with latent video diffusion models than PNP-DM. 110

7.14 **Sampling results of the XOR task on the discretized MNIST dataset.** SGDD faithfully recovers the structural information of the ground truth signal. 122

7.15 **Diversified samples when the measurement y is sparse.** Samples are generated by SGDD when solving an MNIST inpainting task. . . 123

8.1 **Illustration of five benchmark problems in the INVERSEBENCH.** $\mathcal{A}$ represents the forward model that produces measurements from the underlying target. $\mathcal{A}^\dagger$ represents the inverse map. In the linear inverse scattering problem (left two), the measurements are the recorded data from the receivers, and the unknown source we aim to infer is the permittivity map of the object. The bottom panel displays the efficiency and accuracy plots for our benchmarked algorithms. Certain characteristics of the problem cause the efficiency and accuracy trade-offs of each algorithm to vary across tasks. In these plots, the larger radius of the points indicates greater interaction with the forward function $\mathcal{A}$, as measured by the number of forward model evaluations. 126

8.2 **Qualitative comparison showing representative examples of PnP-DP methods and domain-specific baselines across five inverse problems.** Note that for full waveform inversion, Adam* and LBFGS* are initialized with Gaussian-blurred ground truth, serving as references. 134

8.3 **Illustration of the failures of PnPDP methods (DAPS) as an example) on full waveform inversion.** With a small learning rate, DAPS is numerically stable but does not solve the inverse problem effectively. With a slightly larger learning rate, DAPS produces a noisy velocity map that breaks the stability condition of the PDE solver, resulting in a complete failure. 136

8.4 **Relative performance of plug-and-play diffusion prior methods compared with traditional baselines under different levels of measurement sparsity on different tasks.** Metrics are averaged over multiple PnPDP methods. The performance difference increases in general as the measurement becomes sparser. 137

8.5 **PnPDP methods on out-of-distribution test samples.** (a) Blackhole imaging problem on digit inputs; and (b) inverse scattering on sources that contain 9 cells, while the prior model is trained on images with 1 to 6 cells. 137

A.1 **Flow diagram of the reconstructor, $\mathsf{R}_\theta(\cdot)$, in the proposed framework.** We use a residual U-Net reconstructor for all of our models. 144

A.2 **Flow diagram of the samplers, $\mathsf{S}_q(\cdot)$, in the proposed framework.** We use a Multilayer Perceptron for 1D line sampling and a U-Net for 2D point sampling. Both networks take previous observations as inputs and output a probability map, which is rescaled and binarized into the final sub-sampling mask at the next iteration. 145

A.3 **Histograms of pair-wise SSIM differences on all 1,851 test images using 1D line sampling with 4× acceleration factor.** We calculate the improvement of our model with different sequential steps over LOUPE. Our sequential model and non-sequential baseline significantly outperform LOUPE for most subjects. 147

A.4 **Histograms of pair-wise SSIM differences on all 1,851 test images using 1D line sampling with 4× acceleration factor.** We calculate the improvement of the Evaluator (left), PG-MRI (middle), and our best sequential model (4-step sequential) (right) over LOUPE. Our 4-step sequential model significantly outperforms LOUPE, while the other two baselines are substantially worse than LOUPE for most subjects. 148

A.5 **Histograms of pair-wise SSIM differences on all 1,851 test images using 2D line sampling with 4× (first row), 8× (second row), and 16× (third row) acceleration factors.** We calculate the improvement of our model with different sequential steps over LOUPE in each column. For all three acceleration factors, our sequential model outperforms the non-sequential baseline and LOUPE on an increasing percentage of test samples as the number of sequential steps increases. Our sequential models also have increasingly larger advantages over LOUPE as the number of sampled measurements increases (i.e., the acceleration factor decreases). 149

A.6 **Visualizations of the reconstructions of the 394$^{\text{th}}$ (top), 1083$^{\text{th}}$ (middle), 1506$^{\text{th}}$ (bottom) test images with an acceleration factor of 4× for 1D line sampling.** Zoomed-in image patches highlight our significant improvement over previous methods. We find that our learned masks for the 1D line sampling case usually consist of adjacent low-frequency samples. However, only a few of the learned samples have their conjugate symmetric points sampled as well. Our learned policy appears to leverage the conjugate symmetry of the k-space and trade off taking more measurements with taking fewer measurements with higher SNR (by effectively sampling the same measurement twice). 150

A.7 **Visualizations of the reconstructions of the 1355$^{\text{th}}$ test sample with an acceleration factor of 4× (top) and 8× (bottom) for 2D point sampling.** A zoomed-in image patch is shown along with the cumulative k-space measurements selected by each policy. Orange arrows point out the regions where our sequential approach provides more accurate and detailed local structures. 151

B.1 **Conceptual illustration of the subsampling setup with a knee example.** The back dots on the k_y-k_z plane represent k-space trajectories along k_x, which are illustrated by the black arrows. We consider subsampling in the two phase-encoding dimensions (k_y and k_z) of a 3D Cartesian sequence, where the subsampling pattern $\boldsymbol{m}$ is learned from data for some specific downstream task. 156

B.2 **Box plots of the Meniscus Tear ROI reconstruction results under 8× (a) and 16× (b) accelerations.** Within the rectangle between each pair of methods, the top number is the percentage of samples that get improved, and the bottom number is the p-value given by the paired samples t-test. A higher percentage and a lower p-value indicate a more significant improvement. The 95% confidence intervals for all methods are given below their names. 158

B.3 **Visualization of the input of the predictor network for the brain tumor classification task under 16× acceleration on two samples of the BRATS dataset.** Similar to the segmentation results, as a co-design method, TACKLE$_{\text{class.}}$ circumvents the typical "reconstruction" in terms of point-wise similarity with the ground truth image. Instead, the retriever learns a feature map that highlights the region around the tumor for the downstream prediction. 159

C.1 **Example images from the dataset for training the black hole diffusion model prior.** 175

C.2 **Comparison of our method and DPS [64] on estimating the posterior distribution of a Gaussian deblurring problem under a Gaussian prior.** While the mean estimations of the two methods are of roughly the same quality, our approach provides a much more accurate estimation of the posterior per-pixel standard deviation than DPS. 181

C.3 **Visual comparison between our method and baselines on solving the Gaussian deblurring and super-resolution problems with i.i.d. Gaussian noise ($\sigma_y = 0.05$).** We visualize one sample generated by each algorithm. 182

C.4 **Additional visual examples for the Gaussian deblurring problem.** 182

C.5 **Additional visual examples for the motion deblurring problem.** . 183

C.6 **Additional visual examples for the 4× super-resolution problem.** . 184

C.7 **Visual comparison between our method and baselines on solving the coded diffraction pattern (CDP) reconstruction problems with i.i.d. Gaussian noise ($\sigma_y = 0.05$).** We visualize one sample generated by each algorithm. 184

C.8 **Additional visual examples for the Fourier phase retrieval problem.** 185

C.9 **Additional visual examples for the Fourier phase retrieval problem.** 186

C.10 **Additional visual examples given by PnP-DM and DPS using the simulated black hole data.** . . . 187
C.11 **Additional visual examples given by PnP-DM using the real M87 black hole data.** . . . 187
C.12 **Sensitivity analysis on the annealing schedule η_k with different decay rates α (left) and minimum coupling strength $\eta_{\min}$ (right) for a linear (super-resolution) and a nonlinear (coded diffraction patterns) inverse problem.** Recall from Appendix C.3.3 that $\eta_k := \max(\alpha^k \eta_0, \eta_{\min})$, where we set $\eta_0 = 10$ for this experiment. 188
C.13 **Visual examples of intermediate $\boldsymbol{x}^{(k)}$ and $\boldsymbol{z}^{(k)}$ iterates (left) and convergence plots of PSNR, SSIM, and LPIPS for $\boldsymbol{x}^{(k)}$ iterates (right) on the super-resolution problem.** The vertical dashed lines show the iterations at which the $\boldsymbol{x}^{(k)}$ and $\boldsymbol{z}^{(k)}$ iterates are visualized. 188
D.1 **The dirty images from the ideal visibilities.** We use the standard implementation in `EHT` library to get dirty images for each selected frame. . . . 196
D.2 **Subsampling masks of $6\times$ (left) and $8\times$ (right) accelerations used in dynamic MRI experiments.** The white areas in the center indicate the auto-calibration (ACS) signals. The horizontal and vertical directions are the frequency (k_x) and phase (k_x) encoding directions, respectively. The same mask is applied to the sampling of each individual frame of all videos. . . . 197
D.3 **Visualization of VAE reconstructions**. The VAE reconstructions are computed by first encoding the ground truth videos and then decoding them. . . . 201
D.4 **Visualization of video diffusion model unconditional samples**. The videos are sampled by solving the PF-ODE with 100 Euler steps. 202
D.5 **Visualization of STeP posterior samples**. The videos are sampled using the Algorithm 7. . . . 203
E.1 **Computational characteristics of each forward model.** Fwd: runtime of a single forward model evaluation tested on a single A100 GPU. DM: runtime of a single diffusion model evaluation. Fwd Grad: runtime of a single forward model gradient evaluation. DM Grad: runtime of a single diffusion model gradient evaluation. Note that the inverse problem of the Navier-Stokes equation only permits black-box access to the forward model, so its Fwd Grad has no value. 231

E.2 **Multi-modal example on black hole imaging.** The image shows two image modes discovered by DAPS and PnP-DM. 234

LIST OF TABLES

Number *Page*

3.1 **SSIM comparison of 2D point sampling for 4×, 8×, and 16× accelerations.** Our 4-step sequential model outperforms the previous approaches when tested on the fastMRI knee test set. For each model, we compute the test average and standard deviation obtained across three trained models with independent initialization. 18

3.2 **SSIM comparison of 1D line sampling under 4× acceleration.** Our 4-step sequential model outperforms the previous approaches when tested on the fastMRI knee test set. A paired samples t-test shows a statistically significant difference between our 4-step sequential model and LOUPE [12], with a p-value smaller than 10^{-300}. For each model, we compute the test average and standard deviation obtained across three trained models with independent initialization. 20

3.3 **The percentage of test samples on which our method with different numbers of sequential sampling steps (T) outperforms the LOUPE [12] baseline for 2D point sampling.** The percentage average and standard deviation are obtained using results from three trained models with independent initialization. 23

3.4 **Ablation results showing the advantage of co-design with a 4× acceleration ratio and 2D point sampling.** When co-design is specified as "Yes" the reconstruction network has been jointly optimized with the sampler. Otherwise, the sampler was optimized with a fixed reconstructor that was pre-trained with a random sampling policy. 24

4.1 **Comparison of average test local peak signal-to-noise ratio (Local PSNR) in decibel (dB) within Meniscus Tear ROIs under different acceleration ratios (R).** 34

4.2 **Comparison of average test Dice score on the SKM-TEA dataset [83] for segmenting four knee tissues under different acceleration ratios (R).** 36

4.3 **Comparison of average test Dice score on the brain segmentation task under different acceleration ratios (R).** 38

4.4 **Comparison of average test accuracy on the pathology classification task under different acceleration ratios (R).** 40

4.5 **Comparison of average reconstruction accuracy on the experimentally collected dataset under 4× acceleration (top: full-FOV reconstruction; bottom: ROI-oriented reconstruction).** 42
4.6 **Ablation studies on two aspects of co-design for all the considered tasks under 16× acceleration.** 44
4.7 **Ablation studies on model architecture and pre-training for non-reconstruction tasks under 16× acceleration.** 45
4.8 **Comparison of average test PSNR (dB) between reconstruction models trained with task-specific masks and TACKLE$_{\text{recon.}}$ on the fastMRI knee dataset.** 45
6.1 **Quantitative comparison on three noisy linear inverse problems for 100 FFHQ color test images. Bold:** best; Underline: second best. 73
6.2 **Quantitative evaluation on two noisy nonlinear inverse problems for 100 FFHQ grayscale test images. Bold:** best; Underline: second best. 77
6.3 **Quantitative results for the DSA deconvolution problem in radio astronomy.** We compare PnP-DM with the well-known baseline CLEAN on detecting the galaxies and estimating their shapes and fluxes. PnP-DM outperform CLEANs in terms of all the metrics. . . 80
7.1 **Quantitative evaluation for the super-resolution (16×), gaussian deblurring (σ = 6), and box inpainting tasks with matching prompts. Bold:** best; Underline: second best. For each task, we report the mean performance and standard deviation in PSNR, SSIM, and LPIPS across 14 test images. The results show that methods based on our unified framework outperform the baseline approaches, TReg and PSLD, especially on the super-resolution and deblurring problems. The unified methods maintain strong performance without relying on additional components, highlighting their efficiency and generality. 91

7.2 **Quantitative results on black hole video reconstruction and dynamic MRI.** We compare our method against baselines by reporting the mean and standard deviation (shown in parentheses) of selected evaluation metrics computed over 10 test videos (FVD is reported without standard deviation since it evaluates the set of 10 videos collectively). The results clearly demonstrate that by leveraging the spatiotemporal prior, STeP consistently achieves improvements in both spatial quality and temporal consistency relative to baseline methods. 106

7.3 **Comparison on the Navier-Stokes inverse problem.** Metrics are abbreviated as follows: Rel ℓ_2 (relative ℓ_2 error), CRPS (continuous ranked probability score), and SSR (spread-skill ratio). − indicates either that probabilistic metrics are inapplicable (deterministic models) or that it is too expensive to generate enough samples from the algorithm for reliable calculation. **Bold** and Underline indicate best and second best among methods that accommodate unpaired data, respectively. 115

7.4 **Quantitative results for XOR and AND problems on discretized MNIST.** We report the mean and standard deviation (shown in parentheses) of *PSNR* and *class accuracy* across 1,000 generated samples. SGDD demonstrates superior performance on both tasks. 122

8.1 **Requirements on the forward model of the algorithms evaluated in our experiments.** 126

8.2 **Characteristics of different inverse problems in InverseBench.** From left to right: whether the forward model is linear, whether one can efficiently compute the SVD from the forward model, the domain in which the inverse problem is defined, whether the forward model can be solved in closed form, whether one can access gradients from the forward model, and the noise type. 128

B.1 **Comparison of Tackle$_{\text{ROI}}$ on in- and out-of-distribution samples under different acceleration ratios (R).** 157

C.1 **List of hyperparameters for the likelihood step of PnP-DM.** . . . 172

C.2 **List of hyperparameters for the annealing schedule of η in PnP-DM.** 176

C.3 **Comparison of computational efficiency between PnP-DM and other baseline methods.** 178

D.1 **Hyperparameters used by each method on each task.** 191

D.2 **Summary of the hyperparameters of Algorithm 7 (STEP) for black hole video reconstruction and dynamic MRI.** The HMC-related parameters are tuned on 3 leave-out validation videos. The run time and memory are tested using one NVIDIA A100 GPU. . . . 193
D.3 **Summary of the training of spatiotemporal diffusion prior.** We provide and group the hyperparameters according to each component in the model. The model is trained with 1 NVIDIA A100-SCM4-80GB GPU. 199
D.4 **Additional results on dynamic MRI with 6× acceleration.** Following the same setup as in Table 7.2, we report the quantitative results on 10 test videos. 200
D.5 **The data misfit values for samples shown in Figure 7.6.** We report the data misfit metrics for the three obtained modes, which demonstrate that all modes fit the measurement data equally well. . . 200
D.6 **Hyperparameters of `Blade` for the Navier-Stokes experiments in Table 7.3.** 214
D.7 **Hyperparameters of SGDD used in each experiment.** 227
E.1 **Results on linear inverse scattering.** PSNR and SSIM of different algorithms on linear inverse scattering. Noise level $\sigma_y = 10^{-4}$. . . . 228
E.2 **Results on compressed sensing MRI.** Mean and standard deviation are reported over 94 test cases. 228
E.3 **Generalization results on compressed sensing MRI with ×4 acceleration and raw measurements.** Mean and standard deviation are reported over 94 test cases. 229
E.4 **Results on black hole imaging.** PSNR and Chi-squared of different algorithms on black hole imaging. Gain and phase noise and thermal noise are added based on the EHT library. 229
E.5 **Results on FWI.** Mean and standard deviation are reported over 10 test cases. †: initialized from data blurred by Gaussian filters with $\sigma = 20$. $*$: one test case is excluded from the results due to numerical instability. 229
E.6 **Results on Navier-Stokes equation.** Relative ℓ_2 error of different algorithms on 2D Navier-Stokes inverse problem, reported over 10 test cases. $*$: one or two test cases are excluded from the results due to numerical instability. 230

E.7 **Table of metrics we use to capture the computation complexity of each algorithm.** 230
E.8 **Diagnostic performance of compressed sensing MRI reconstructions.** 230
E.9 **Model card for pre-trained diffusion models.** 236
E.10 **Hyperparameter search space and final choices of the diffusion-model-based algorithms on all five inverse problems.** Columns marked with task names present the chosen values for the reported main results in Appendix E.1. These values are selected by a hybrid hyperparameter search strategy described in Appendix E.3.1.2. . . . 238

Chapter 1

INTRODUCTION

Imaging technologies enable us to visualize objects that cannot be observed directly—such as distant stars or internal anatomical structures hidden within the human body. For example, the German scientist Wilhelm Conrad Röntgen discovered X-rays in 1895 and found that bones and soft tissues absorb X-rays differently. X-ray images allow doctors to see inside the body and diagnose bone fractures, infections, and other medical conditions. In astronomy, radio telescopes have allowed scientists to survey the sky, leading to the discoveries of new stars, distant galaxies, and signals from deep space that help us understand the origins and structure of the universe.

Traditional imaging techniques have greatly benefited humanity, but they are approaching their limits as the demands of science and engineering continue to grow. For instance, while X-rays can distinguish between bones and soft tissues, they struggle to differentiate among various soft tissues, which often share similar physical properties. To see more subtle and nuanced structures, more sophisticated imaging techniques are needed. Similarly, imaging the suspected black hole at the center of the Milky Way would be impossible with a traditional radio telescope—unless its dish were as large as the Earth [32].

Computational Imaging refers to a class of techniques that integrate sensor design (hardware) with advanced computational algorithms (software) to form images [30]. Unlike traditional imaging techniques that rely on minimal post-processing, computational imaging techniques leverage complex reconstruction algorithms that are tightly coupled with the sensor system. These techniques have enabled unprecedented capabilities that are far beyond the reach of traditional techniques [71, 204, 304, 355]. To image a black hole, scientists connected radio telescopes distributed across the globe to create an Earth-sized computational telescope. By combining measurements from different telescopes in a physically meaningful way, the computational telescope achieves a high enough resolution to see black holes that are thousands or even millions of light-years away from us. Even so, sophisticated algorithms are required to recover the black holes from the combined measurements. In medicine, magnetic resonance imaging (MRI) was developed to visualize different

soft tissues in the human body [171, 186]. MRI uses electromagnetic waves and their interaction with tissue to acquire frequency-domain measurements, carefully designed so that the formed measurements correspond to the Fourier components of the target region. An image reconstruction algorithm is necessary to invert the Fourier measurements back to the image space. In both cases, computation plays a central role and is deeply integrated with the sensor system.

1.1 Computational Imaging as Inverse Problems

Many problems in computational imaging can be formulated as *inverse problems* [268], such as astronomical imaging [49], optical microscopy [62], and medical imaging [204]. Inverse problems are also common in many other domains of science and engineering, including geophysics [294] and fluid dynamics [143]. Mathematically, inverse problems often take the following form

$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}_0) + \boldsymbol{n}. \tag{1.1}$$

In this equation, $\boldsymbol{x}_0 \in \mathbb{C}^n$ denotes the target *ground truth* that we want to image. Since $\boldsymbol{x}_0$ cannot be directly observed, we collect indirect *measurements* $\boldsymbol{y} \in \mathbb{C}^m$ using a sensor system described by the *forward model* $\mathcal{A} : \mathbb{C}^n \to \mathbb{C}^m$. The term $\boldsymbol{n} \in \mathbb{C}^m$ accounts for measurement noise, which includes model mismatch, hardware imperfection, and other sources of error. For simplicity, we model the noise in an additive form, although more complex noise models may apply in practice [200, 280].

Solving inverse problems amounts to finding a mapping $\mathcal{A}^\dagger : \mathbb{C}^m \to \mathbb{C}^n$ such that

$$\widehat{\boldsymbol{x}} := \mathcal{A}^\dagger(\boldsymbol{y}) \approx \boldsymbol{x}_0 \tag{1.2}$$

where we often refer to $\widehat{\boldsymbol{x}}$ as a reconstruction of $\boldsymbol{x}_0$. Directly inverting $\mathcal{A}$ is usually infeasible because the problem is under-constrained and $\mathcal{A}$ can be highly nonlinear. Therefore, the design of the inverse mapping $\mathcal{A}^\dagger$ becomes challenging and requires the knowledge of both the forward model $\mathcal{A}$ and the underlying image $\boldsymbol{x}_0$.

1.2 Computational Imaging Pipeline and Some Key Challenges

Conceptually, computational imaging systems consist of two main stages. Figure 1.1 provides a schematic illustration using MRI as an example.

Stage 1: Measurement Acquisition Acquiring measurements is the first stage of a computational imaging system. The total number of measurements is usually

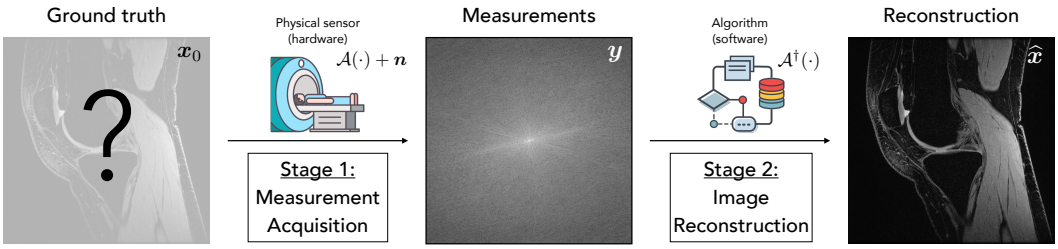


Figure 1.1: **Pipeline of computational imaging.** In Stage 1 (Measurement Acquisition), a physical sensor acquires indirect, noisy measurements of some unknown ground truth $\boldsymbol{x}_0$. Mathematically, this can be modeled by $\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}_0) + \boldsymbol{n}$ where $\boldsymbol{n}$ represents measurement noise. In Stage 2 (Image Reconstruction), an algorithm $\mathcal{A}^\dagger(\cdot)$ is designed to obtain a reconstruction $\widehat{\boldsymbol{x}}$ from the measurements.

limited because each measurement comes with a cost in, e.g., time and hardware. Therefore, it is important to wisely choose informative measurements of $\boldsymbol{x}_0$ for accurate recovery. But how do we define or quantify the amount of information in a set of measurements? How to choose the most informative measurements? Mathematically, different forward models have different numerical properties, some of which will make the recovery of $\boldsymbol{x}_0$ easier. In the case of linear forward models, i.e., $\mathcal{A} := \boldsymbol{A} \in \mathbb{R}^{m \times n}$, a matrix with a smaller condition number (defined as the ratio between the largest and the smallest singular value) is known to be easier to invert. But what about more general, nonlinear forward models? What if we have prior knowledge about the distribution of $\boldsymbol{x}_0$? If we already have some measurements about $\boldsymbol{x}_0$, could we use initial measurements to guide the selection of future ones? What if we particularly care about some local regions and/or specific features of $\boldsymbol{x}_0$? Answering these questions requires both domain-specific insights and a general optimization framework. In this thesis, we aim to address them by leveraging techniques in machine learning.

Stage 2: Image Reconstruction The second stage involves reconstructing an estimate $\widehat{\boldsymbol{x}}$ from the acquired measurements $\boldsymbol{y}$. Due to the non-trivial relationship between $\boldsymbol{y}$ and $\boldsymbol{x}_0$, having the most informative measurements does not guarantee a successful recovery. Reconstruction algorithms still play an important role because they are directly responsible for image formation and significantly impact the imaging quality. One central challenge is the *ill-posed* nature of Equation (1.1), meaning that multiple solutions could explain the same $\boldsymbol{y}$ and they depend sensitively on $\boldsymbol{y}$ [268]. There has been a rich literature on various kinds of image reconstruction algorithms. Some focus on specific domains [107, 134, 205, 267] and others are designed for generic problems [2, 19, 248, 291]. However, many open problems

remain. How to solve challenging inverse problems where a significant amount of information is lost in the forward model? Given that most existing methods focus on recovering static images, how to solve inverse problems on videos? How to discover multiple solutions to an inverse problem? Is there a reliable and efficient way of quantifying the likelihood of different solutions and their degrees of uncertainty? These questions are not only interesting from a theoretical perspective but also grounded in the needs of real-world applications. Addressing these challenges is essential for unlocking the next generation of computational imaging techniques.

1.3 Thesis Overview

The word "sample" in the title carries two different meanings. In the context of measurement acquisition, sampling means choosing a set of measurements from a space of possible measurements. In the context of image reconstruction, sampling means generating samples from the distribution of all possible solutions that agree with the observed measurements. This thesis presents two lines of work that leverage various machine learning techniques to address both notions of sampling in computational imaging.

In **Part I**, we investigate sampling pattern optimization in compressed sensing MRI (CS-MRI). Chapter 2 presents an overview of this part and some preliminaries of CS-MRI. We mainly consider two aspects that are particularly relevant for MRI. First, in Chapter 3, we present a method that takes advantage of the sequential nature of MRI acquisition and learns a sequential sampling strategy. Dividing the entire acquisition process into a few steps, the proposed method sequentially selects the next batch of measurements based on the previously observed ones. We show that this approach significantly outperforms the baseline that determines the choice of measurements ahead of time. Compared to other reinforcement learning-based methods, our approach is trained in a fully end-to-end manner, leading to better efficiency and reconstruction quality. Second, in Chapter 4, we propose a new framework for optimizing sampling patterns with respect to downstream diagnostic tasks beyond standard reconstruction. We recognize the fact that having a reconstruction is not the end of the workflow and that common image similarity metrics do not fully reflect the reconstruction quality from a downstream task perspective. We demonstrate the effectiveness of the proposed framework on three types of tasks—reconstruction that focuses on a region-of-interest (ROI), segmentation, and classification. Our framework shows consistent improvements over baselines that are limited to image reconstruction in terms of task-relevant metrics.

In **Part II**, we present a framework for sampling from posterior distributions using diffusion models (DMs), a state-of-the-art family of generative models. Chapter 5 covers relevant background on Bayesian inverse problems and DMs. In Chapter 6, we introduce PnP-DM, a principled posterior sampling framework inspired by the Split Gibbs sampler (SGS) [296], a rigorous type of Markov chain Monte Carlo (MCMC) sampler. Our main contribution is to identify a key connection between SGS and DMs via a unified formulation called the EDM framework [151]. This connection allows us to easily incorporate pre-trained DMs as image priors for solving inverse problems. We show both a convergence theory and a validation of our approach on a synthesis Gaussian prior example. Experimental results on a few linear and nonlinear inverse problems demonstrate that our method provides both better individual reconstructions and more accurate estimations of posterior distributions. We then work towards a unified framework for diffusion-based posterior estimation in Chapter 7. Specifically, we present four instantiations of the unified framework that handle more general inverse problems. Topics include (1) incorporating text as prior, (2) harnessing high dimension for video inverse problems, (3) accommodating black box forward models, and (4) generalizing to discrete spaces. Adopting similar alternating-update structures as PnP-DM, these instantiations exhibit similar convergence guarantees and strong empirical performance. Finally, in Chapter 8, we propose a comprehensive benchmark for systematically evaluating diffusion-based methods for solving scientific inverse problems. The insights gained from these experiments point to promising directions for further improving posterior estimation in computational imaging.

Part I

Sampling: Measurement Acquisition

Chapter 2

OVERVIEW AND PRELIMINARIES

In this part, we investigate the first topic around *sampling* in computational imaging—measurement acquisition—in the context of magnetic resonance imaging.

2.1 Magnetic Resonance Imaging (MRI)

Magnetic resonance imaging (MRI) is a widely used imaging technology for clinical diagnosis and biomedical research [171]. MRI provides powerful tools to non-invasively visualize anatomy and physiology without ionizing radiation. However, a central challenge of MRI is its slow acquisition process. This is due to the fact that the raw measurements of MRI must be sampled one at a time rather than simultaneously, leading to a total scan time on the order of minutes. This long acquisition time leads to high cost, patient discomfort, and significant reconstruction artifacts when patients move. Therefore, there is a strong demand for accelerating the MRI acquisition process.

2.2 Compressed Sensing MRI (CS-MRI)

Compressed sensing magnetic resonance imaging (CS-MRI) is a popular accelerated MRI technology based on *compressed sensing (CS)* [41], which aims to reconstruct the underlying image from a set of subsampled k-space measurements [204].

2.2.1 Basics

The common setup of CS-MRI involves reconstructing a target image $\boldsymbol{x}_0 \in \mathbb{C}^n$ from its subsampled, noisy k-space measurements

$$\boldsymbol{y} := \boldsymbol{M}\boldsymbol{F}\boldsymbol{x}_0 + \boldsymbol{n} \in \mathbb{C}^m \quad (m \ll n), \tag{2.1}$$

where $\boldsymbol{F}$ is the Fourier transform, $\boldsymbol{M} \in \{0,1\}^{m\times n}$ is the subsampling matrix with $\boldsymbol{m} \in \{0,1\}^n$ denoting its subsampling pattern, and $\boldsymbol{n} \in \mathbb{C}^m$ is the complex measurement noise. In the parallel imaging setup, the measurements are collected from multiple receiver coils. For the j-th coil, the measurements $\boldsymbol{y}_j$ can be expressed as

$$\boldsymbol{y}_j := \boldsymbol{M}\boldsymbol{F}\boldsymbol{S}_j\boldsymbol{x}_0 + \boldsymbol{n}_j \in \mathbb{C}^m, \tag{2.2}$$

where $\boldsymbol{S}_j$ is a diagonal matrix that represents the pixel-wise sensitivity map and $\boldsymbol{n}_j$ is the measurement noise of the j-th coil. For both settings, we refer to $b := \|\boldsymbol{m}\|_1$

as the sampling budget and $R := \frac{n}{b}$ as the acceleration ratio of the acquisition. Once properly implemented[1], the accelerated sequence will shorten the scan time by a factor of R, leading to significantly higher throughput.

2.2.2 Image Reconstruction

Traditional image reconstruction techniques in CS-MRI include solving a regularized optimization problem [205, 243]. Recently, deep learning (DL) methods have achieved state-of-the-art performance on CS-MRI reconstruction. One line of work combines data-driven priors with model-based iterative reconstruction (MBIR) [4, 149, 248, 291]. Another line of work learns a model-free reconstruction network via end-to-end training [173, 174, 235, 305, 329]. A third line of work, known as deep unrolling (DU), combines the characteristics of MBIR and end-to-end training [1, 2, 117, 123, 136, 191, 255, 267, 328, 331, 343]. The idea is to "unroll" an iterative optimization procedure into a cascade of mappings and train these mappings end-to-end so that they can gradually map a low-resolution input image to a high-quality output reconstruction. Inheriting the advantage of both MBIR and end-to-end learning, these methods exhibit state-of-the-art performance on CS-MRI reconstruction.

2.2.3 Sampling Pattern Design

Subsampling patterns in traditional CS-MRI are often generated randomly or handcrafted to have a point spread function (PSF) with a high degree of incoherence defined under the compressed sensing theory [41]. Popular subsampling patterns include the 2D variable density [204], bidirectional Cartesian [301], Poisson-disc [288], and continuous-trajectory variable density [54], among others [74, 239]. These subsampling patterns are designed for generic image reconstruction and not optimized for any specific body part or diagnostic purpose. Therefore, these patterns may lead to suboptimal performance for downstream tasks where specific anatomical or pathological information is relevant.

Recently, a new group of DL-based methods, known as *co-design*, has been proposed to jointly optimize the subsampling pattern $\boldsymbol{m}$ and a downstream reconstruction module, leading to better reconstruction performance than the traditional CS-MRI methods [3, 8, 12, 216, 229, 237, 299, 300, 302, 307, 327, 330, 344, 351, 362].

[1]The actual implementation of subsampling in MRI involves physical and hardware constraints, which can be complicated. Here we omit the details. We will provide details on the implementation in Chapter 4.

Due to the sequential nature of MRI acquisition, reinforcement learning (RL) based methods have also been considered to learn an adaptive policy for determining $\boldsymbol{m}$ [13, 195, 231].

2.3 Part Outline

For the rest of this part, we will discuss two topics around the optimization of sampling patterns for CS-MRI:

- In Chapter 3, we propose an end-to-end approach to learning a sequential sampling strategy. Our method adaptively selects which measurements to take on the fly based on the previous selections. We show that the proposed method not only outperforms previous non-adaptive end-to-end methods but also improves upon RL-based methods. The improvements are reflected in terms of both statistical significance and visual quality.
- In Chapter 4, we present a framework for optimizing the sampling patterns directly for downstream diagnostic tasks. We show that if there is a clear downstream objective for an MRI scan, such as inspecting a particular region of interest or type of tissue, optimizing directly for that objective often leads to significantly better performance than optimizing for reconstruction as a surrogate. Besides experiments on publicly available datasets, we also program our learned sequence on an MRI machine. Notably, our actual implementation achieves a 4× scan time reduction as expected while providing high-fidelity reconstructions.

Chapter 3

LEARNING SEQUENTIAL SAMPLING AND RECONSTRUCTION STRATEGIES

In this chapter, we investigate one way to improve measurement acquisition: leveraging the sequential nature of the acquisition process. Intuitively, if some measurements are observed before we have a chance to choose others, leveraging the information in the observed measurements allows us to make a wiser choice about which measurements to observe next. Based on this intuition, we propose a fully differentiable framework that *jointly* learns a *sequential* sampling policy simultaneously with a reconstruction strategy. This co-designed framework can adapt during acquisition to capture the most informative measurements for a particular target (Figure 3.1). Experimental results on the fastMRI knee dataset demonstrate that the proposed approach successfully utilizes intermediate information during the sampling process to boost reconstruction performance. In particular, our proposed method outperforms the learning-based LOUPE baseline [12] on over 96% of test samples. We also investigate the individual and collective benefits of the sequential sampling and co-design strategies.

This chapter is based on our work [335], published in the *Proceedings of the 1st Machine Learning for Health symposium 2021 (ML4H 2021)*, PMLR volume 158, ISSN: 2640-3498. This work also received the Best Paper Award for ML4H 2021. The appendix for this chapter is Appendix A. The code for the work presented in this chapter is available at `https://github.com/tianweiy/SeqMRI`.

3.1 Introduction

The success of CS-MRI depends on two critical factors: (a) a carefully designed k-space subsampling pattern to collect informative measurements, and (b) a reconstruction method that accurately recovers high-quality images from subsampled data. Current MRI protocols collect measurements over time using *static* subsampling patterns that were designed *a priori*. To further accelerate a scan, we are interested in *sequential* sampling patterns that adapt to a target based on intermediate information collected during acquisition.

A high-fidelity MRI reconstruction stems from the cooperation between the k-space

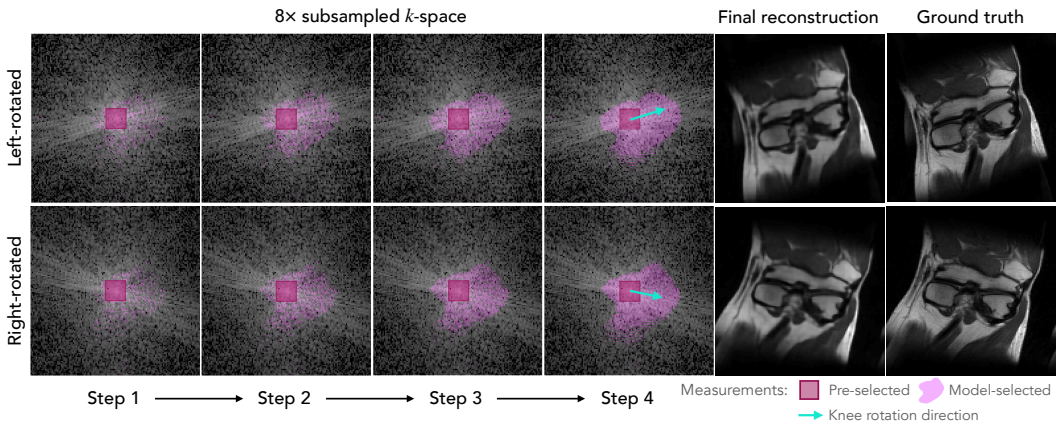


Figure 3.1: **We propose a sequential sampling and reconstruction co-design framework for accelerated MRI that adapts to a target during acquisition.** Here, we visualize the sampling policy and final reconstruction of rotated knees in a single-coil imaging setting with 8× acceleration (8× subsampling). The first four columns show the cumulative k-space measurements selected by the proposed learned sampler (pink) in acquisition steps 1 through 4 (during a 4-step acquisition). The fifth column shows the final image recovered by the proposed learned reconstructor, and the last column is the ground truth. This example illustrates how our model has learned to adapt to different k-space distributions: the final sampling patterns in the fourth column contain visible directional structure that aligns with the k-space power spectrum. Rotated anatomical images, such as these rotated knee images, were *not* included in the training set (or quantitatively evaluated test set).

sampling strategy and the reconstruction method. Traditionally, MRI subsampling patterns and reconstruction methods have been largely *independently designed*. We are instead interested in *co-design*, where jointly designing the two components can synergistically boost reconstruction quality. Our approach builds on neural network-based co-design frameworks that have shown strong empirical performance and take advantage of efficient differentiable training [12, 157, 158, 270].

In this chapter, we propose an end-to-end differentiable framework that successfully combines co-design and sequential sampling. Specifically, we design an explicit sequential structure of T steps, with each step consisting of a jointly learned k-space sampler and reconstructor. Comparing our model with prior work in accelerated MRI, we investigate the individual and collective benefits of sequential sampling and co-design. We evaluate the proposed model on the NYU fastMRI datasets and find that: (1) even a single sequential step consistently improves performance compared to using a pre-designed sampling pattern; (2) more sequential steps can improve reconstruction quality, but with diminishing returns; and (3) a fully differentiable approach enables more efficient and effective co-design than non-differentiable

methods. Notably, despite various published works on sequential sampling using reinforcement learning [13, 231], we are among the first to demonstrate consistent and statistically significant improvement over a state-of-the-art learned non-sequential baseline [12] through the use of a fully-differentiable sequential computation graph.

The rest of the chapter is organized as follows. In Section 3.2, we review past literature in accelerated MRI from the perspectives of co-design and sequential sampling. We introduce our proposed framework and its training procedure in Section 3.3. We then present our experimental settings, comparisons between our model and other baselines, and ablation studies in Section 3.4. Finally, we conclude with a discussion on future directions of our framework in Section 3.5.

3.2 Related Work

Prior work in accelerated MRI can be organized into four quadrants, split across two dimensions: methods that (1) independently (and/or manually) design the sampler and reconstructor versus data-driven co-design, and (2) specify the sampling pattern before a scan (pre-designed) versus adapt samples to the target during acquisition. In Section 3.2.1, we cover traditional methods that independently (and/or manually) design the sampler and reconstructor. In Section 3.2.2, we discuss previous methods that perform pre-designed acquisition in a co-design framework. In Section 3.2.3, we introduce recent work on sequential sampling for accelerated MRI. We conclude in Section 3.2.4 with an overview of methods that attempt to combine co-design and sequential sampling, but without end-to-end learning. Our end-to-end framework efficiently combines co-design and sequential sampling, successfully inheriting the advantages of both approaches.

3.2.1 Traditional Methods

Accelerated MRI sampling patterns implemented on commercial scanners are motivated by ideas in compressed sensing (CS) [41]. Since anatomical images are sparse in a linearly transformed space, it is possible to reconstruct a high-fidelity image with incoherent k-space data sampled below the Nyquist-Shannon rate [205]. In the context of 2D CS-MRI, prior work has investigated uniform density random sampling, variable density sampling [204], Poisson-disc sampling [288], continuous-trajectory variable density sampling [54], and equi-spaced sampling [121]. These sampling patterns are easy to implement, but not adaptive to specific datasets or target images.

Once sparse k-space measurements have been acquired, an image is typically reconstructed via an optimization problem that involves two objectives: the first encour-

ages a reconstruction that matches the observed data, while the second addresses the ill-posed nature of the under-determined system through image regularization. Common regularization terms include total variation (TV) [29] and the ℓ_1-norm after a sparsifying transformation (obtained using wavelets [204, 207] or dictionary decompositions [140, 243, 340]).

Recently, convolutional neural networks (CNNs) have demonstrated impressive performance in MRI reconstruction. Strategies include unrolled networks [123, 191, 255, 331], UNet-based networks [142, 174], GAN-based networks [235, 329], among others [190, 305, 360]. These learning methods have achieved state-of-the-art performance on public MRI challenge datasets [338]. In our proposed co-design model, we employ a convolutional UNet for image reconstruction.

3.2.2 Co-Design

The goal of co-design is to jointly identify the optimal sampling and reconstruction strategies. This is an NP-hard combinatorial optimization problem due to the discrete nature of the sampling pattern. Theoretically, one could identify an optimized reconstructor for every possible sampling strategy, and then pick the overall strategy that performs best. However, this brute-force optimization approach is not practical, as it requires enumerating an exponential number of possible sampling combinations. Early work formulated the co-design as a nested (or bi-level) optimization problem and alternated between optimizing a sampler and a reconstructor [242].

More recently, deep learning has enabled a data-driven solution to the co-design problem, where the sampler and reconstructor can be jointly learned through end-to-end training. For example, [12, 312, 344] proposed co-design frameworks for 2D Cartesian k-space sampling and [299, 311] applied co-design to 2D radial k-space sampling.[1] These methods have shown superior performance over previous baselines that combine an individually-optimized sampler and reconstructor pair [12, 270, 299, 312, 344]. However, these methods do not take advantage of the sequential nature of data collection during an MRI scan, and only solve for a generic sampling pattern for an entire dataset.

3.2.3 Sequential Sampling

Since MRI scanners acquire measurements in a sequential manner, recent work has modeled the sampling process in the context of sequential decision making.

[1] Differentiable co-design of discrete sensing and reconstruction methods has also been successfully applied to other imaging domains as well [270].

Sequential decisions enable the sampling pattern to adapt to different input images by choosing the next k-space sample based on prior measurements. Reinforcement learning (RL) methods have primarily been employed for this purpose. For example, [13, 231] formulate the sampling problem as a Partially Observable Markov Decision Process (POMDP) and use Policy Gradient [18] and DDQN [124] methods, respectively. These RL methods heavily rely on a pre-trained reconstructor, which leads to a training mismatch (and thus potentially suboptimal performance), since the reconstructor was trained with a sampling strategy that does not match the strategy eventually employed by the RL-learned sampler. Furthermore, these RL methods are difficult and costly to train, as they are non-differentiable. As a consequence, in the context of accelerated MRI, these methods either fail to be adaptive to different input images or have only limited improvement over simple baselines [13, 231].

3.2.4 Co-Design & Sequential Sampling

Several approaches have attempted to combine co-design with sequential sampling strategies, but they have achieved only limited success to date. The work of [146] draws inspiration from AlphaGo [258] and trains a sampler to emulate the policy distribution obtained through a Monte Carlo Tree Search (MCTS); the reconstructor is trained during alternating optimization steps. However, according to the results in [13], the MCTS method in [146] has limited improvement over simple baselines, and is outperformed by the sequential sampling method in [13] without co-design. This poor performance may be due to the overall MCTS framework not being end-to-end differentiable. Alternatively, [351] proposes a framework that trains a ResNet to reconstruct the anatomical image simultaneously with an evaluator network that is trained to select the most uncertain measurement in k-space. Although the authors demonstrate how this framework can be used to sequentially choose the next sample, it is not explicitly trained end-to-end and is outperformed by [231], which does not use co-design. This training-testing mismatch limits the potential improvement of sequential sampling. In contrast, we design a fully differentiable end-to-end framework that leverages the sequential nature of k-space MRI acquisition during both training and testing.

3.3 Method

Figure 3.2 summarizes the co-design framework for our sequential sampling and reconstruction model. We partition the k-space sampling budget into T steps and

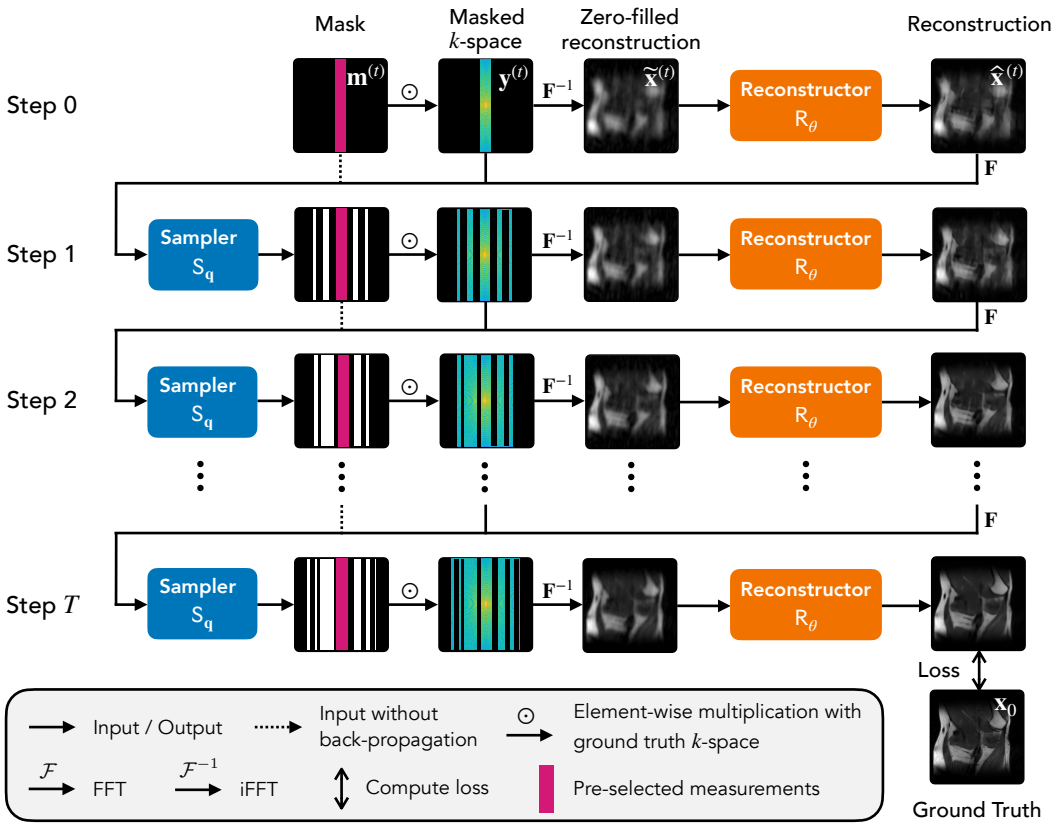


Figure 3.2: **Overview of the proposed sequential sampling framework.** Low-frequency samples are pre-selected and measured in k-space. The subsampled k-space is transformed into a zero-filled image, which is fed into a reconstructor $\mathsf{R}_{\boldsymbol{\theta}}(\cdot)$ to produce an intermediate image reconstruction (Equation (3.1)). The intermediate reconstruction and measurements are passed into a sampler network $\mathsf{S}_{\boldsymbol{q}}(\cdot)$, which outputs a discrete probability distribution representing suggested samples for the next iteration. An action is sampled from this distribution (Equation (3.2)), and the corresponding k-space measurements are acquired. The sampling and reconstruction process is repeated for T steps. The sampler and reconstructor are neural networks learned via end-to-end training with a loss on the final reconstructed image. Weights are shared across all T acquisition steps.

denote a model with T sequential steps as "T-Step Seq" hereafter. At each step t, the pipeline applies a sampler $\mathsf{S}_{\boldsymbol{q}}(\cdot)$ and a reconstructor $\mathsf{R}_{\boldsymbol{\theta}}(\cdot)$. The reconstructor aims to remove aliasing artifacts in the zero-filled reconstruction $\widetilde{\boldsymbol{x}}^{(t)}$ of the acquired measurements $\boldsymbol{y}^{(t)}$:

$$\widehat{\boldsymbol{x}}^{(t)} = \mathsf{R}_{\boldsymbol{\theta}}\left(\widetilde{\boldsymbol{x}}^{(t)}\right) = \mathsf{R}_{\boldsymbol{\theta}}\left(\boldsymbol{F}^{-1}\boldsymbol{y}^{(t)}\right). \tag{3.1}$$

The sampler, in turn, seeks to intelligently select which k-space samples to observe next, based on previously observed measurements and the k-space of the

reconstruction so far:

$$\boldsymbol{m}^{(t)} \sim \mathsf{S}_{\boldsymbol{q}}\left(\boldsymbol{m}^{(t-1)}, \boldsymbol{y}^{(t-1)}, \boldsymbol{F}\widehat{\boldsymbol{x}}^{(t-1)}\right) \quad \text{s.t. } \|\boldsymbol{m}^{(t)} - \boldsymbol{m}^{(t-1)}\|_1 = b^{(t)} \tag{3.2}$$

where $\boldsymbol{m}^{(t)}$ is a binary mask representing the sampling pattern collected up until step t and $b^{(t)}$ is the sampling budget at step t.

We model the sampler $\mathsf{S}_{\boldsymbol{q}}(\cdot)$ and reconstructor $\mathsf{R}_{\boldsymbol{\theta}}(\cdot)$ as neural networks, and jointly optimize their weights, $\boldsymbol{q}$ and $\boldsymbol{\theta}$, by minimizing the image reconstruction error between the final-step reconstruction $\widehat{\boldsymbol{x}}^{(T)}$ and the ground truth target image $\boldsymbol{x}_0$:

$$\boldsymbol{q}^*, \boldsymbol{\theta}^* = \arg\min_{\boldsymbol{q},\boldsymbol{\theta}} \mathcal{L}_{\text{recon.}}\left(\widehat{\boldsymbol{x}}^{(T)}, \boldsymbol{x}_0\right), \tag{3.3}$$

where $\mathcal{L}_{\text{recon.}}$ is an image reconstruction loss function based on, e.g., structural similarity index measure (SSIM) [310] or peak signal-to-noise ratio (PSNR). We choose to share sampler and reconstructor weights across all steps. The sampler and reconstructor are described in more detail in Section 3.3.1 and Section 3.3.2, respectively.

3.3.1 Sampler

Subsampling Patterns We follow prior work to consider two types of k-space sampling: 1D line sampling and unconstrained 2D point sampling [12, 13, 338, 351]. Figure 3.3 illustrates these two sampling scenarios, which enable different levels of sampling flexibility. In 1D line sampling, only vertical trajectories along the vertical (frequency encoding) direction can be sampled. This corresponds to the Cartesian subsampling setting in 2D MRI. On the other hand, 2D point sampling allows any measurement on the 2D grid in k-space to be acquired. Unconstrained 2D point sampling represents an upper bound on sampling flexibility and can be implemented on 3D Cartesian sequences. We note that our sequential sampling framework is applicable to other patterns, such as radial sampling [26].

As low-frequency k-space measurements contain the most information about large-scale anatomical structure, it is common practice in accelerated MRI to fix a small number of low-frequency k-space samples to always be collected [13, 231, 351]. We follow this strategy by allocating $1/8$ of the total sampling budget to the central low-frequency region in all experiments.

Probabilistic Modeling We follow [12] to model the subsampling strategy at step t as an element-wise Bernoulli policy. To learn the optimal probabilities at

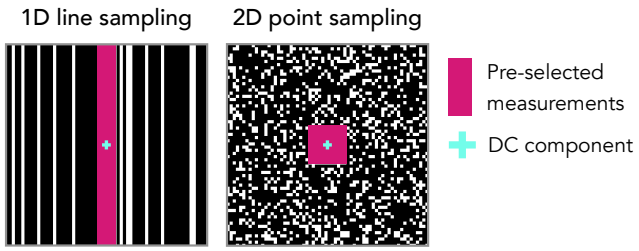


Figure 3.3: **Visualizations of two types of k-space sampling patterns: 1D line sampling and 2D point sampling.** White regions are sampled from a uniform distribution over the space of possible actions. The center low-frequency samples are pre-selected in all experiments before any further sampling. DC corresponds to the $(0, 0)$ frequency.

each step, the sampler takes in the sampling mask $\boldsymbol{m}^{(t-1)}$, k-space measurements $\boldsymbol{y}^{(t-1)}$, and the k-space of the reconstruction $\boldsymbol{F}\widehat{\boldsymbol{x}}^{(t-1)}$ up to date (see Equation (3.2)). The sampler then outputs a set of logits $\boldsymbol{q}_i^{(t)} \in \mathbb{R}^l$, where l denotes the number of possible sampling indices (i.e., number of columns for the 1D line pattern and number of pixels for the 2D point pattern). These logits are subsequently turned into probabilities $\widetilde{\boldsymbol{p}}_i^{(t)} := \text{Sigmoid}(\boldsymbol{q}_i^{(t)})$. We then rescale $\widetilde{\boldsymbol{p}}^{(t)}$ to obtain a probability map $\overline{\boldsymbol{p}}^{(t)}$ with $b^{(t)}$ measurements in expectation:

$$\overline{\boldsymbol{p}}^{(t)} = \begin{cases} \frac{\alpha}{\beta}\widetilde{\boldsymbol{p}}^{(t)} & \text{if } \beta \geq \alpha \\ \mathbf{1} - \frac{1-\alpha}{1-\beta}(\mathbf{1} - \widetilde{\boldsymbol{p}}^{(t)}) & \text{otherwise} \end{cases}$$

where $\alpha := \frac{b^{(t)}}{n}$, $\beta := \frac{\|\widetilde{\boldsymbol{p}}^{(t)}\|_1}{n}$, and $\mathbf{1}$ is the all-one vector. Additionally, to avoid acquiring the same measurements again, the probabilities of sampling previously acquired lines are set to zero:

$$\boldsymbol{p}^{(t)} = \overline{\boldsymbol{p}}^{(t)} \odot (1 - \boldsymbol{m}^{(t-1)}) \tag{3.4}$$

where $\odot$ is the element-wise multiplication. We then draw Bernoulli random samples for all sampling locations according to the probability map $\boldsymbol{p}^{(t)}$, where 1 indicates sampling a location and 0 indicates otherwise. Mathematically, we can write this sampling process as the k-space sampling mask $\boldsymbol{m}_t$ for acquisition step t via:

$$\boldsymbol{m}^{(t)} = \mathbb{1}[\boldsymbol{u} \leq \boldsymbol{p}^{(t)}] + \boldsymbol{m}^{(t-1)}, \tag{3.5}$$

where $\boldsymbol{u} \in [0, 1]^l$ is a vector of l independent realizations of the uniform distribution on the interval $[0, 1]$. We repeatedly sample $\boldsymbol{u}$ until $\|\boldsymbol{m}^{(t)} - \boldsymbol{m}^{(t-1)}\|_1 \approx b^{(t)}$ under a small tolerance. This sampling process encourages exploration of different patterns

Table 3.1: **SSIM comparison of 2D point sampling for 4×, 8×, and 16× accelerations.** Our 4-step sequential model outperforms the previous approaches when tested on the fastMRI knee test set. For each model, we compute the test average and standard deviation obtained across three trained models with independent initialization.

Method	4×	8×	16×
Random	90.40 (0.02)	87.43 (0.05)	84.25 (0.00)
Spectrum	92.39 (0.01)	90.38 (0.01)	88.37 (0.01)
LOUPE [12]	92.44 (0.01)	90.60 (0.03)	88.73 (0.04)
4-Step Seq. (ours)	**92.91** (0.01)	**91.07** (0.02)	**89.10** (0.03)

and ensures that the sampling patterns approximately satisfy the budget constraint. Note that the indicator function $\mathbb{1}[\cdot]$ is not differentiable, which hinders the training of the model through back-propagation. To overcome the non-differentiability, we use a straight-through estimator that applies the indicator function in the forward pass, while approximating its gradients by treating the binary indicator function as a sigmoid during back-propagation [20]. In this way, we can capture binary sampling in real MR scanning, while retaining gradients for end-to-end training.

Sampler Architecture For the 1D line sampler, we use a Multilayer Perceptron (MLP) network with five layers separated by ReLU activation functions. For the 2D point sampler, we instead use an eight-block UNet with ReLU activation functions whose architecture is more scalable on higher-dimensional action spaces. Further details of the network architectures for both samplers are included in Appendix A.1.

3.3.2 Reconstructor

Our proposed co-design sequential framework learns the parameters of a reconstructor jointly with the sampler. Although many networks have been proposed for MR image reconstruction [123, 255, 267, 331], the choice of reconstructor architecture is not the main focus of this chapter. We following [12, 13, 338] to adopt a standard 8-block U-Net architecture [250] . The input to the reconstructor at each step t is the complex-valued zero-filled image, $\widehat{\boldsymbol{x}}^{(t)}$, and the output is a single-channel real-valued image, $\widetilde{\boldsymbol{x}}^{(t)}$. The UNet reconstructor contains four downsampling blocks and four upsampling blocks, each consisting of two 3×3 convolutions separated by ReLU and instance normalization [286]. We note that our framework is agnostic to the specific reconstructor architecture.

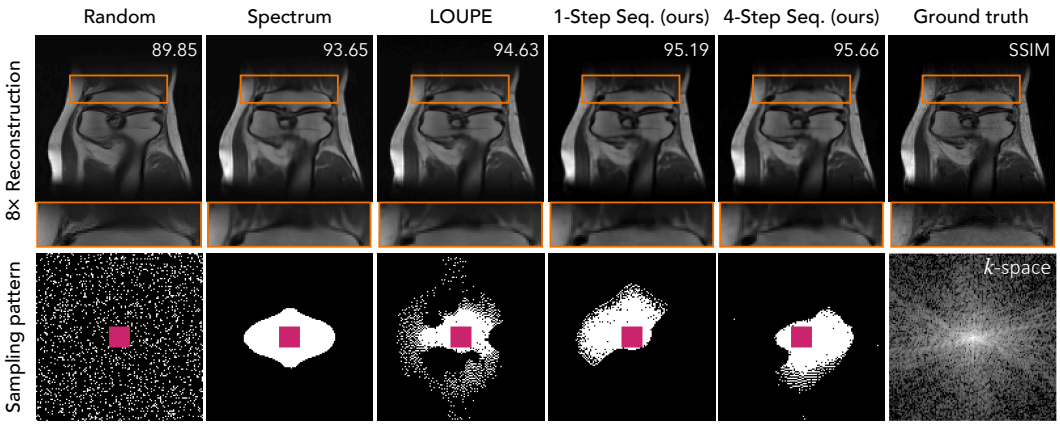


Figure 3.4: **Visualizations of example reconstructions with an 4× acceleration for 1D line sampling.** Two zoomed-in image patches are shown along with the cumulative k-space measurements selected by each policy. Our sequential approach often provides more accurate reconstructions with detailed local structures. More visualizations are included in the Appendix A.4.

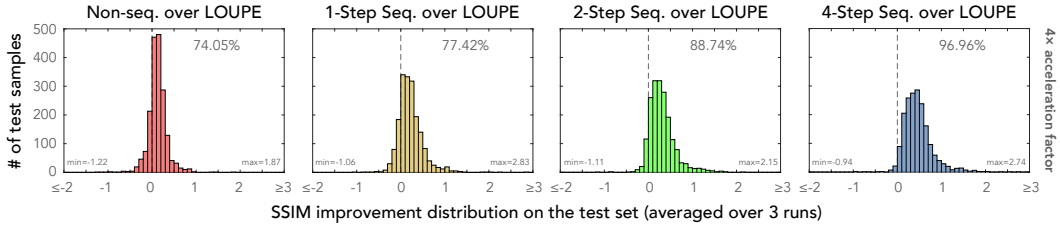


Figure 3.5: **Histograms of pair-wise SSIM differences between our sequential models and LOUPE [12] on all 1,851 test images.** Positive numbers indicate improvement over LOUPE. The results are acquired by averaging three runs of 4× accelerated 2D point subsampling. More sequential steps lead to a bigger advantage over LOUPE, with the 4-step sequential model outperforming LOUPE on 96.96% of samples. This performance pattern holds for the 1D line scenario and other acceleration factors as well, as shown in Appendix A.3. More quantitative results are given in Table 3.3.

3.4 Experiments

3.4.1 Setup and Implementation Details

We evaluate our sequential sampling and reconstruction method on the NYU fastMRI open dataset [338][2]. The dataset provides raw single-coil k-space measurements for knee images, with 973 training set volumes and 97 validation set volumes [338]. We follow the setup of [231] and split the original validation set into a new validation set with 48 volumes and a test set with 49 volumes, which results in 34,742 2D slices for training, 1,785 slices for validation, and 1,851 slices

[2] https://fastmri.org/

Table 3.2: **SSIM comparison of 1D line sampling under 4× acceleration.** Our 4-step sequential model outperforms the previous approaches when tested on the fastMRI knee test set. A paired samples t-test shows a statistically significant difference between our 4-step sequential model and LOUPE [12], with a p-value smaller than 10^{-300}. For each model, we compute the test average and standard deviation obtained across three trained models with independent initialization.

Random	Equispaced	Evaluator [351]	PG-MRI [13]	LOUPE [12]	4-Step Seq. (ours)
85.95 (0.05)	86.86 (0.06)	85.99 (0.04)	87.97 (0.09)	89.52 (0.02)	**91.08** (0.09)

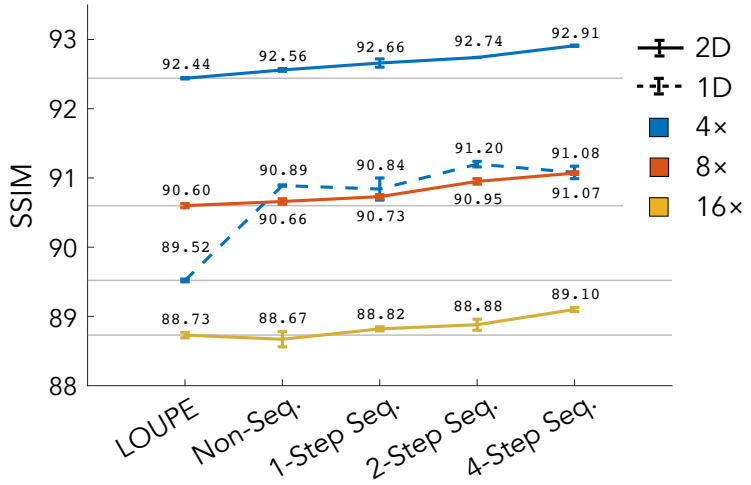


Figure 3.6: **Comparison between our sequential model and the LOUPE model on the fastMRI knee test set.** Our sequential model outperforms LOUPE for all acceleration ratios with an improvement comparable to 25% of the benefit of doubling the number of k-space measurements. The performance of our sequential model in the 1D line sampling case significantly outperforms LOUPE but plateaus after 2 sequential sampling steps, possibly due to the restricted action space of 1D line sampling.

for testing. For computational efficiency, we follow [13, 351] to crop the k-space to the center 128×128 region.

We use the structural similarity index measure (SSIM) for the primary evaluation metric, which has been found to correlate well with expert evaluations [164]. We define the loss function $\mathcal{L}_{\text{recon.}}(\boldsymbol{x}, \boldsymbol{x}_0) := -\text{SSIM}(\boldsymbol{x}, \boldsymbol{x}_0)$ following [13, 231, 267], which is computed using a window size of 7×7 and hyperparmeters $k_1 = 0.01$, $k_2 = 0.03$ following the fastMRI challenge's official implementation. We use the Adam optimizer [162] and train our model for 50 epochs with a learning rate of $1e-3$ for 2D point sampling experiments and $5e-5$ for 1D line sampling experiments. The learning rate is decreased by half every ten epochs. Training each model takes at most one day on a single NVIDIA RTX 2080Ti GPU.

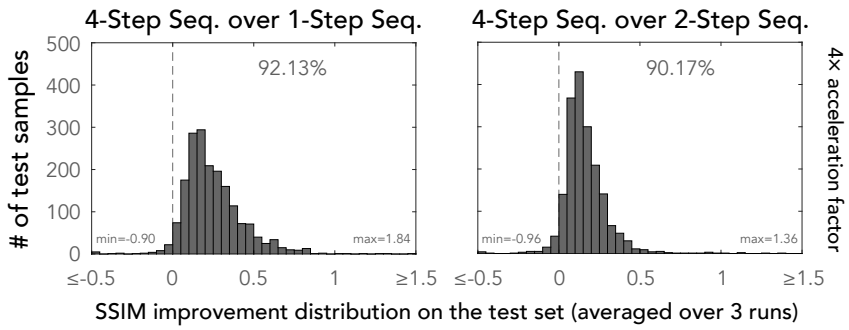


Figure 3.7: **Histograms of pair-wise SSIM comparison on all 1,851 test images with a different number of sequential steps (T), using 2D point sampling with a 4× acceleration factor.** The relative error between the 4-step and 1-step (left) or 2-step(right) demonstrates that additional sequential steps help to boost performance, but with diminishing returns as T increases.

3.4.2 Results

In Figure 3.1, we visualize our framework's sequential sampling masks and final reconstruction for rotated knees in the 8× acceleration setting. Starting from pre-selected measurements, our model sequentially samples 2D k-space measurements based on previous observations. Here, we demonstrate that our model can accurately estimate and leverage the k-space structure during the sequential sampling steps. In particular, the final sampling patterns contain visible directional structures that align with the true k-space power spectrum induced by knee rotation. This highlights the adaptivity of our sequential model, as no rotated anatomical images are included in the training set.

2D Point Sampling In Table 3.1, we compare our method to several baselines for 4×, 8×, and 16× accelerations, including: (1) Random [110]: randomly select points from a uniform distribution, (2) Spectrum [289]: select points with the largest k-space magnitude over the training set, (3) LOUPE [12]: select points before acquisition using a distribution learned via co-design. For each baseline, the reconstruction network has been trained with the specified sampling policy. Please refer to Appendix A.2 for the implementation details of these baseline methods. Our 4-step sequential model achieves the best reconstruction performance across different acceleration ratios. A paired samples t-test between our method and the previous state-of-the-art pre-designed sampling approach, LOUPE [12], indicates a statistically significant difference in performance, with a p-value less than 10^{-160} for all acceleration ratios. By inspecting Figure 3.6, one can see that our 4-step model outperforms LOUPE for all acceleration ratios, with an improvement comparable

to 25% of the benefit of doubling the number of k-space measurements from 8× to 4×.

1D Line Sampling We compare our model to previous methods for the 1D line sampling with a 4× acceleration factor in Table 3.2. The baselines we consider include: (1) Random: randomly select k-space lines from a uniform distribution, (2) Equispaced [121]: select equidistant lines, (3) Evaluator [351]: sequentially select lines following a learned evaluation function, (4) PG-MRI [13]: sequentially select lines using a conditional distribution trained by a policy gradient algorithm, (5) LOUPE [12]: select lines before acquisition using a distribution learned via co-design. The implementation details of these baselines are included in Appendix A.2. Our 4-step sequential framework significantly outperforms prior methods, with an SSIM improvement of roughly 1.6 over the previous learning-based method, LOUPE [12]. A paired samples t-test also indicates a highly statistically significant boost in performance compared to LOUPE with a t-score of 64.01 and a p-value smaller than 10^{-300}. Note that our differentiable end-to-end framework also significantly outperforms a sequential reinforcement learning optimization approach, PG-MRI [13].

Figure 3.4 shows sample images reconstructed using the approaches mentioned above. Using the same number of k-space samples, our 4-step sequential model most accurately recovers important anatomical structures and details. The orange and blue patches under each reconstruction highlight certain regions where our method significantly outperforms other baselines.

Adaptive vs. Pre-Designed Sampling Figure 3.5 shows histograms of pair-wise SSIM differences on each test sample, computed between our sequential method and LOUPE. Here we introduce a non-sequential baseline referred to as “non-seq." which uses the same network architecture as our sequential model but replaces the prior k-space measurements used as input with a random tensor. The “non-seq." baseline demonstrates a performance comparable to that of LOUPE in Figure 3.5, Figure 3.6, and Table 3.3. However, more sequential steps consistently lead to a higher percentage of improved samples. Thus, we can conclude that the improvement is not merely due to a better framework architecture but the adaptive sampling strategy of our approach.

Table 3.3: **The percentage of test samples on which our method with different numbers of sequential sampling steps (T) outperforms the LOUPE [12] baseline for 2D point sampling.** The percentage average and standard deviation are obtained using results from three trained models with independent initialization.

Method	4×	8×	16×
Non-Seq.	74.05 (2.56)	60.18 (3.03)	46.98 (8.58)
1-Step Seq.	77.42 (7.89)	57.05 (4.36)	51.09 (4.16)
2-Step Seq.	88.74 (0.45)	83.04 (3.78)	56.42 (4.62)
4-Step Seq.	**96.96** (0.73)	**92.62** (0.46)	**76.91** (2.29)

Number of Sequential Steps We further ablate the impact of the number of sequential steps. For the case of 2D point sampling in Figure 3.6, the accuracy consistently increases as the number of sequential sampling steps increases. To further understand the improvements seen with additional sequential steps, we perform a pair-wise SSIM comparison between our sequential models. Figure 3.7 shows the result of 2D point sampling with a 4× acceleration ratio. Additional sequential steps boost the reconstruction performance for almost all subjects, with diminishing returns as T increases. Table 3.3 shows quantitative results that compare the percentage of test samples that outperform the LOUPE baseline. On 2D point sampling, our 4-step sequential model outperforms LOUPE roughly 97%, 89%, and 77% of the time for the 4×, 8× and 16× acceleration factors, respectively.

Ablation Study on Co-Design We demonstrate the advantage of co-designing the sampler and reconstructor in Table 3.4. Specifically, we pre-train a reconstructor using a uniform sampling policy and demonstrate the performance improvement that occurs when jointly learning the reconstructor weights with the sampler. Co-designing the reconstructor with the sampler significantly improves performance, with an increase of 2.33–2.51 SSIM for 2D point sampling with a 4× acceleration factor.

3.5 Conclusion

Accelerating the MRI acquisition process has the potential to reduce patient discomfort, increase throughput, and expand the use of MRI worldwide. In this chapter, we proposed an end-to-end sequential sampling and reconstruction framework for accelerated MR imaging. We leveraged the sequential nature of MRI acquisition and design a model with an explicit sequential structure that jointly optimizes a neural network-based sampler simultaneously with a network-based reconstructor.

Table 3.4: **Ablation results showing the advantage of co-design with a 4× acceleration ratio and 2D point sampling.** When co-design is specified as "Yes" the reconstruction network has been jointly optimized with the sampler. Otherwise, the sampler was optimized with a fixed reconstructor that was pre-trained with a random sampling policy.

Co-design	1-Step Seq.	4-Step Seq.
✓	**92.66** (0.06)	**92.91** (0.01)
✗	90.33 (0.01)	90.40 (0.02)

In our experiments, this simple framework outperforms previous state-of-the-art MR sampling approaches for up to nearly 97% of the test samples on the fastMRI single-coil knee dataset. Overall, our results suggest that future methods for MRI sampling can benefit from the collaboration of sequential sampling and co-design via end-to-end learning.

Chapter 4

LEARNING END-TO-END STRATEGIES FOR THE DOWNSTREAM TASK OF INTEREST

In the previous chapter, we discussed one way of improving measurement acquisition via learning sequential sampling strategies. In this chapter, we explore another approach to improving measurement acquisition: optimizing the entire pipeline end-to-end for the downstream task of interest. Traditional CS-MRI methods design the measurement subsampling strategy independently of the downstream task prediction [12, 204, 305]. For example, these methods are optimized for reconstruction accuracy over the entire field-of-view, even when the goal of the MRI scan is, e.g., to inspect a certain kind of tissue or region of interest. This often results in suboptimal performance on the task that ultimately matters to the user. To address this limitation, we propose TACKLE, a unified co-design framework that jointly optimizes subsampling, reconstruction, and prediction strategies to maximize downstream task performance. The naïve approach of simply appending a task prediction module and training with a task-specific loss leads to suboptimal downstream performance. Instead, we develop a training procedure where a backbone architecture is first trained for a generic pre-training task (image reconstruction in our case), and then fine-tuned for different downstream tasks with a prediction head. Experimental results on multiple public MRI datasets show that TACKLE achieves an improved performance on various tasks over traditional CS-MRI methods. We also demonstrate that TACKLE is robust to some distribution shift by showing that it generalizes to a new dataset we experimentally collected using different acquisition setups from the training data. Without additional fine-tuning, TACKLE leads to both numerical and visual improvements compared to existing baselines. We have further implemented a learned 4×-accelerated sequence on a Siemens 3T MRI Skyra scanner. Compared to the fully-sampling scan that takes 335 seconds, our optimized sequence only takes 84 seconds, achieving a four-fold scan time reduction as desired, while maintaining high performance.

This chapter is based on our work [323], published in the *IEEE Transactions on Computational Imaging*, vol. 10, pp. 1040-1054, 2024, doi: 10.1109/TCI.2024.3410521. The appendix for this chapter is Appendix B. The code for the work presented in this chapter is available at `https://github.com/zihuiwu/TACKLE`.

4.1 Introduction

In the existing co-design literature for CS-MRI, task prediction is often viewed as a post-processing step decoupled from image reconstruction. Existing methods focus predominantly on image reconstruction and rely on standard image similarity metrics such as mean square error (MSE) or peak signal-to-noise ratio (PSNR) as a proxy for performance on a downstream task. Such a reconstruction-oriented formulation lacks a direct connection with the downstream tasks that reflect actual clinical needs [219]. We are thus motivated to ask:

> *Can one improve the accuracy of downstream task prediction by optimizing the entire CS-MRI pipeline in an end-to-end fashion?*

With end-to-end co-design methods, it seems like we are only one step away from incorporating downstream tasks as part of the optimization. Namely, one can simply append a task prediction module and add a task-specific loss. However, as shown in Figure 4.1 and Table 4.2, this approach leads to a suboptimal performance on the task prediction and is sometimes even worse than the traditional approach of separate reconstruction and task prediction. These results indicate that it remains a challenge to learn task-specific strategies robustly for CS-MRI.

In this chapter, we address this challenge by proposing a unified framework, ***ta**sk-specific **c**odesign of **k**-space subsamp**l**ing and pr**e**diction* (TACKLE), for designing task-specific CS-MRI systems. Different from existing works that focus on specific tasks, TACKLE is a general framework that accommodates different downstream tasks. To do so, we design a two-step training strategy that mimics the training of modern language and vision models. TACKLE is first trained for a generic task of image reconstruction, and then fine-tuned for specific downstream tasks. We find that this approach can effectively learn generalizable task-specific strategies that lead to significant and consistent improvements, with an example shown in Figure 4.1 (c). Besides the standard task of reconstructing the full field-of-view (which we call full-FOV reconstruction hereafter), we demonstrate TACKLE on three other tasks covering both pixel-level and image-level imaging problems: region-of-interest (ROI) oriented reconstruction, tissue segmentation, and pathology classification. Our experimental results show that end-to-end optimization for task prediction sometimes circumvents the typical reconstruction in terms of pixel-wise accuracy, but leads to improved accuracy on the task of interest by effectively extracting key visual information for task prediction.

The main contributions of this chapter are as follows:

- We provide a general framework (TACKLE) that learns specific strategies for a variety of CS-MRI tasks. TACKLE optimizes the entire CS-MRI pipeline, from measurement acquisition to label prediction, in an end-to-end fashion *directly* for a user-defined task.
- We validate TACKLE on multiple MRI datasets, covering different body parts, scanning sequences, and hardware setups. Experimental results show that TACKLE outperforms the reconstruction-oriented baseline methods on *all* considered settings. We evaluate the proposed end-to-end architecture and training procedure through ablation studies. Our results offer guidance for designing effective task-specific CS-MRI systems in the future.
- We show the generalization of TACKLE to out-of-distribution data by deploying it to a dataset we experimentally acquired using a different acquisition sequence from that of the training data. We further implement a learned 4×-accelerated sequence on a Siemens 3T MRI Skyra scanner. The sequence shortens the scan time from 335 seconds to 84 seconds, a four-fold time reduction as desired, while maintaining high performance. These experiments highlight the real-world practicality of our method.

4.2 Related Work

4.2.1 Reconstruction-Oriented Co-Design

The success of DL methods in CS-MRI reconstruction motivates the idea of jointly optimizing acquisition together with reconstruction via end-to-end training. Recently, there has been a rapidly growing literature on optimizing a parameterized sampling strategy jointly with a CNN reconstructor [3, 8, 12, 216, 229, 237, 299, 300, 302, 307, 327, 330, 344, 351, 353, 362]. These methods have different architectural designs and applicable scenarios, but all rely on the differentiable nature of neural networks to optimize the reconstruction accuracy over the choice of k-space measurements. The learned subsampling pattern and reconstruction network are thus specific to the dataset. The end-to-end training enables synergistic cooperation between the learned subsampling pattern and reconstructor, achieving state-of-the-art reconstruction performance. From a task perspective, however, having a reconstruction is not the end of the workflow. These methods rely on either human evaluation, a traditional task prediction algorithm, or a CNN for task predictions, which are out of the scope of these papers.

4.2.2 Task-Oriented Co-Design

Recent work has investigated the co-design idea in the context of limited tasks beyond full-FOV reconstruction, such as physical parameter estimation [325, 353, 363] and segmentation [98, 271, 308, 309, 311]. Using task-specific loss functions in their training procedures, these proposed methods demonstrate stronger task performance than methods trained by a reconstruction-only loss. Most of these proposed approaches leave either subsampling or prediction as a pre-determined fixed module, and focus on co-designing the other modules [98, 271, 325, 353, 363]. On the other hand, the authors of [309, 311] proposed to jointly optimize all three steps and investigated a brain segmentation task using a U-Net reconstructor and predictor. Although these methods show the potential of extending co-design beyond reconstruction, they are each fine-tuned for one particular task, do not easily accommodate different types of data (e.g., multi-coil), and have not been demonstrated on real out-of-distribution datasets. The most relevant work to ours in the literature is a concurrent work by Wang *et al.* [308], in which the authors presented a thorough investigation of optimizing the entire CS-MRI pipeline for various segmentation problems. In this work, we cast a wider net for the task-specific CS-MRI co-design problem. In particular, we demonstrate our unified framework for designing generalized CS-MRI pipelines, Tackle, on three different tasks beyond full FOV reconstruction. Tackle performs robustly on this broad range of tasks and experiments, and is implemented and tested on a Siemens scanner.

4.3 Method

Figure 4.2 illustrates the architecture of Tackle. As a co-design CS-MRI method, Tackle jointly optimizes the sampler, retriever, and predictor for a task-dependent loss. In the following subsections, we describe each module in order, and more implementation details can be found in Appendix B.1.2.

4.3.1 Sampler

We consider 2D Cartesian subsampling patterns, i.e., $\boldsymbol{m} \in \{0, 1\}^n$. Similar to the previous chapter, we model the subsampling strategy as the element-wise Bernoulli distribution with a probability vector $\boldsymbol{p} \in [0, 1]^n$, i.e., $\boldsymbol{m}_i \sim \text{Bernoulli}(\boldsymbol{p}_i)$, following [12, 327, 335]. To learn the optimal sampling probabilities, we directly optimize a set of logit parameters $\boldsymbol{q}_i$ that first give us a set of probabilities $\widetilde{\boldsymbol{p}}_i := \text{Sigmoid}(\boldsymbol{q}_i)$. We then rescale $\widetilde{\boldsymbol{p}}$ to obtain a probabilistic sampling mask $\boldsymbol{p}$ that would result in b

measurements in expectation via Bernoulli sampling:

$$\boldsymbol{p} = \begin{cases} \frac{\alpha}{\beta}\widetilde{\boldsymbol{p}} & \text{if } \beta \geq \alpha \\ \mathbf{1} - \frac{1-\alpha}{1-\beta}(\mathbf{1} - \widetilde{\boldsymbol{p}}) & \text{otherwise} \end{cases}$$

where $\alpha := \frac{b}{n}$, $\beta := \frac{\|\widetilde{\boldsymbol{p}}\|_1}{n}$, and $\mathbf{1}$ is the all-one vector. During training, the sampler draws a k-space sampling mask $\boldsymbol{m}$ by sampling $\boldsymbol{m}_i \sim \mathsf{Bernoulli}(\boldsymbol{p}_i)$. We repeatedly sample $\boldsymbol{m}$ until $\|\boldsymbol{m}\|_1 \approx b$ under a small tolerance. This sampling process encourages exploration of different patterns and ensures that the sampling patterns approximately satisfy the budget constraint. Since the sampling process is not differentiable, we use the same straight-through estimator technique as in the previous chapter to overcome the non-differentiability [20]. During testing, we set the top b indices of $\boldsymbol{p}$ with the highest probabilities to 1 (to sample) and others to 0 (not to sample). This binarization guarantees that the sampling mask strictly satisfies the sampling budget constraint and all slices of a volume share the same sampling mask. We also allocate $1/8$ of the sampling budget for the low-frequency region around the DC component, which we refer to as the pre-select region. The pre-selected measurements provide auto-calibration signals (ACS) for multi-coil reconstruction and stabilize the training of some baselines. Therefore, we include the pre-select region for all experiments for consistency. More discussion on this can be found in Appendix B.1.3. We denote the sampler as $\mathsf{S}_{\boldsymbol{q}}$ where $\boldsymbol{q}$ is the vector of learnable parameters.

4.3.2 Retriever

After acquiring measurements, we employ a retriever to extract visual information from noisy and subsampled k-space measurements. We note that we name the module "retriever" instead of "reconstructor" because it is jointly optimized with the downstream predictor for non-reconstruction tasks. Hence, the retriever should not be interpreted as a reconstructor as its output may not be a typical "reconstruction" in terms of pixel-wise accuracy. We denote the retriever as $\mathsf{R}_{\boldsymbol{\theta}}$ where $\boldsymbol{\theta}$ is its weights. We select the E2E-VarNet [267] since it is a model-based DU architecture that combines the MRI forward model and deep learning architectures, and achieves excellent performance on CS-MRI reconstruction [338]. E2E-VarNet also accommodates multi-coil k-space data with its ability to estimate coil sensitivity maps. Specifically, our E2E-VarNet retriever operates in k-space and consists of 12 refinement steps, each of which includes a U-Net [250] with independent weights from each other. For each U-Net, we use the standard architecture with the follow-

ing parameters: 2 input and output channels, 18 channels after the first convolution filter, 4 average down-pooling layers, and 4 up-pooling layers. The final output layer of the retriever is an inverse Fourier transform followed by a root-sum-squares reduction for each pixel over all coils. The output of the retriever is a batch of magnitude images. For reconstruction tasks, a loss function will be directly applied to the output. For non-reconstruction tasks, the output will be fed into an additional predictor module described in the next section.

4.3.3 Task-Specific Design: Predictor and Loss Function

We demonstrate TACKLE on three tasks that together represent a gradual progression from generic full-FOV reconstruction to clinically relevant tasks.

4.3.3.1 ROI-Oriented Reconstruction

For many MRI scans, only a small region of the FOV is relevant to the reader, so we define a task where we aim to maximize reconstruction quality around that region. In contrast to the full-FOV reconstruction task, the reconstruction accuracy in this task is only measured over the region of interest (ROI) of each image instead of the entire FOV. We hereafter refer to this task as *ROI-oriented reconstruction*. This task is a first step from generic full-FOV reconstruction to more specific downstream tasks in CS-MRI.

There is no predictor for this reconstruction task, and the output of the retriever will directly be used for evaluation. The evaluation metric we use is the local peak signal-to-noise ratio (PSNR), which is the PSNR within the ROI of an underlying image $\boldsymbol{x}$. Let $\mathcal{R}_{\boldsymbol{x}}$ be the set of indices i that are within the ROI of an image $\boldsymbol{x}$. Note that $\mathcal{R}_{\boldsymbol{x}}$ varies from one image $\boldsymbol{x}$ to another. We define the local PSNR within the ROI as

$$\mathsf{LocalPSNR}(\widehat{\boldsymbol{x}}, \boldsymbol{x}; \mathcal{R}_{\boldsymbol{x}}) := 10 \log_{10} \frac{\mathsf{max}(\boldsymbol{x})^2}{\mathsf{LocalMSE}(\widehat{\boldsymbol{x}}, \boldsymbol{x}; \mathcal{R}_{\boldsymbol{x}})} \tag{4.1}$$

where $\mathsf{LocalMSE}(\widehat{\boldsymbol{x}}, \boldsymbol{x}; \mathcal{R}_{\boldsymbol{x}}) := \frac{1}{|\mathcal{R}_{\boldsymbol{x}}|} \sum_{i \in \mathcal{R}_{\boldsymbol{x}}} (\widehat{\boldsymbol{x}}_i - \boldsymbol{x}_i)^2$ and $\mathsf{max}(\boldsymbol{x})$ is the largest pixel value of $\boldsymbol{x}$. We optimize our model for the local reconstruction quality using $\mathcal{L}_{\mathrm{ROI}}(\widehat{\boldsymbol{x}}, \boldsymbol{x}_0) := -\mathsf{LocalPSNR}(\widehat{\boldsymbol{x}}, \boldsymbol{x}_0; \mathcal{R}_{\boldsymbol{x}_0})$ as the training loss where $\boldsymbol{x}_0$ is the ground truth image.

4.3.3.2 Tissue Segmentation

For this task, we aim to predict segmentation maps of different body tissues. Accurately segmenting a tissue from the rest of the organ provides important anatomical and pathological information [128, 238, 254]. Conventional segmentation workflow involves human evaluation and traditional algorithms, which often require standard reconstructions of certain contrasts as input [105]. On the contrary, TACKLE does not require reconstruction as a necessary intermediate step, and is optimized for segmentation performance in an end-to-end fashion.

We include an additional predictor $\mathsf{P}_{\boldsymbol{\phi}}$ with weights $\boldsymbol{\phi}$ after the retriever. We choose the U-Net architecture due to its ability to solve medical image analysis tasks [14, 116, 250]. The specific parameters are: 1 input channel, c output channels (where c is the number of segmentation classes), 64 channels after the first convolution filter, 4 average down-pooling layers, and 4 up-pooling layers.

We use the Dice score [88, 222, 364] as the evaluation metric. The Dice score measures the degree of overlap between two segmentation maps and takes a value between 0 (no overlap) and 1 (perfect overlap). During training, we employ the Dice loss $\mathcal{L}_{\text{seg.}}(\widehat{\boldsymbol{s}}, \boldsymbol{s}_0) := 1 - \mathsf{DiceScore}(\widehat{\boldsymbol{s}}, \boldsymbol{s}_0)$ where $\boldsymbol{s}_0$ is the ground truth segmentation map. For both training and evaluation, we apply a Softmax function across all the classes for each pixel and then calculate the Dice loss/score. During the evaluation, we apply an additional binarization step where we set the class with the highest value after Softmax as 1 and others as 0. In this way, we assign each pixel of the predicted segmentation map $\widehat{\boldsymbol{s}}$ to exactly one class.

4.3.3.3 Pathology Classification

The third task we consider is to determine whether a potential pathology exists in an MRI image, such as a suspected tumor. Using algorithms to automatically analyze MRI scans could lead to improved diagnosis accuracy in clinical practice [219]. We formulate this task as a binary image classification problem, where the negative class means the underlying image $\boldsymbol{x}$ does not contain any pathology lesion, and the positive class means it does contain a lesion. Through this proof-of-concept classification task, we go beyond pixel-level problems and show the benefit of task-specific co-design for solving an image-level problem.

Similar to the segmentation task, we include an additional predictor in the pipeline, which we also denote as $\mathsf{P}_{\boldsymbol{\phi}}$ to simplify notations. Specifically, we choose the ResNet

[126], which is an established architecture for computer vision tasks, especially image classification. We use the standard ResNet18 architecture except for using 1 input channel and 2 output dimensions.

We use the binary cross entropy (BCE) as the loss function for this classification task, $\mathcal{L}_{\text{class.}}(\widehat{\boldsymbol{c}}, \boldsymbol{c}_0) := \mathsf{BCE}(\widehat{\boldsymbol{c}}, \boldsymbol{c}_0)$ where $\boldsymbol{c}_0$ is the ground truth classification label. For evaluation metrics, we consider both the classification accuracy ($\mathsf{ClsAcc} := \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}$) and the F_1 score ($\mathsf{F_1\ score} := \frac{2\text{TP}}{2\text{TP+FP+FN}}$) where TP, TN, FP, and FN are the number of True Positive, True Negative, False Positive, and False Negative, respectively. The classification accuracy is more interpretable, while the F_1 score is more robust to class imbalance. So we include both metrics for a more comprehensive evaluation.

4.3.4 Training Procedure

We summarize the training objective for each task as follows:

- ROI-oriented reconstruction:

$$\min_{\boldsymbol{q},\boldsymbol{\theta}} \mathcal{L}_{\text{ROI}} \left(\mathsf{R}_{\boldsymbol{\theta}} \left(\mathsf{S}_{\boldsymbol{q}} \odot \boldsymbol{k} \right), \boldsymbol{x}_0 \right)$$

- Segmentation:

$$\min_{\boldsymbol{q},\boldsymbol{\theta},\boldsymbol{\phi}} \mathcal{L}_{\text{seg.}} \left(\mathsf{P}_{\boldsymbol{\phi}} \left(\mathsf{R}_{\boldsymbol{\theta}} \left(\mathsf{S}_{\boldsymbol{q}} \odot \boldsymbol{k} \right) \right), \boldsymbol{s}_0 \right)$$

- Classification:

$$\min_{\boldsymbol{q},\boldsymbol{\theta},\boldsymbol{\phi}} \mathcal{L}_{\text{class.}} \left(\mathsf{P}_{\boldsymbol{\phi}} \left(\mathsf{R}_{\boldsymbol{\theta}} \left(\mathsf{S}_{\boldsymbol{q}} \odot \boldsymbol{k} \right) \right), \boldsymbol{c}_0 \right)$$

where $\boldsymbol{k} \in \mathbb{C}^n$ contains all k-space measurements of $\boldsymbol{x}$ and $\odot$ denotes element-wise multiplication.

When performing end-to-end training over multiple stages, we empirically observed that a model trained from scratch tends to run into either optimization (hard to train) or generalization (unable to generalize) issues. Some prior works address these problems using a hybrid of reconstruction and task-dependent loss [271, 309, 311, 325, 363]. This approach requires tuning a weight parameter that balances the two losses. We adopt an alternative approach that avoids tuning this additional parameter. Specifically, we first train the sampler and retriever jointly with a full-FOV PSNR loss until convergence:

$$\min_{\boldsymbol{q},\boldsymbol{\theta}} \mathcal{L}_{\text{FOV}} \left(\mathsf{R}_{\boldsymbol{\theta}} \left(\mathsf{S}_{\boldsymbol{q}} \odot \boldsymbol{k} \right), \boldsymbol{x}_0 \right)$$

Task	Training procedure and loss	Notation example
(ROI) recon.	S&R: PSNR loss	$\text{LP+UN}_{\text{FOV}}$
	S&R: local PSNR loss (w/ pre-training)	$\text{TACKLE}_{\text{ROI}}$
Tissue seg.	S&R: PSNR loss $\rightarrow$ P: Dice loss	$\text{PD+UN}_{\text{recon.}}$
	S&R&P: Dice loss (w/ pre-training)	$\text{TACKLE}_{\text{seg.}}$
Patho. class.	S&R: PSNR loss $\rightarrow$ P: BCE loss	$\text{LOUPE}_{\text{recon.}}$
	S&R&P: BCE loss (w/ pre-training)	$\text{TACKLE}_{\text{class.}}$

S: sampler, R: retriever, P: predictor
&: joint training in an end-to-end fashion
$\rightarrow$: separate training stages for reconstruction and prediction

where $\mathcal{L}_{\text{FOV}}(\widehat{\boldsymbol{x}}, \boldsymbol{x}_0) := -\text{PSNR}(\widehat{\boldsymbol{x}}, \boldsymbol{x}_0)$. We refer to this as the pre-training step in later sections. With the weights learned for the sampler and retriever, we then add the predictor (initialized with random weights) into the framework and fine-tune all three components. We find that the pre-training step allows the model to better learn task-specific strategies, as demonstrated by an ablation study in Section 4.6.2. This training procedure mimics the training of foundation models in state-of-the-art language and vision models, which are first pre-trained on a general task and then fine-tuned for more specific tasks. Similar procedures can be found in other task-specific co-design papers, such as [98, 271].

4.4 Experiments on Large-Scale Datasets

We first demonstrate the effectiveness of our framework on the three considered tasks using large-scale datasets. We categorize all the investigated datasets and settings in the bottom left panel of Figure 4.2. For each task, we demonstrate that the proposed task-specific co-design framework achieves better performance than baselines that separately design reconstruction and prediction. We abbreviate different variants of the proposed method and baselines in the following way, based on their task and training procedure:

To clarify, the subscript "recon." for the segmentation and classification methods means that the sampler and retriever are trained for full-FOV reconstruction, and a predictor is subsequently trained for the downstream task with the sampler and retriever fixed. This is equivalent to training a predictor with the reconstructed images by these methods as input for the downstream task.

Table 4.1: **Comparison of average test local peak signal-to-noise ratio (Local PSNR) in decibel (dB) within Meniscus Tear ROIs under different acceleration ratios (R).**

Data	R	LP+UN$_{\text{FOV}}$	PD+UN$_{\text{FOV}}$	LOUPE$_{\text{FOV}}$	TACKLE$_{\text{ROI}}$
Single-coil	8	26.95	28.23	30.32	**34.04**
	16	25.16	26.05	27.32	**31.54**
Multi-coil	8	27.55	32.68	34.88	**40.65**
	16	26.02	30.00	31.79	**37.89**

4.4.1 ROI-oriented Reconstruction

Dataset and Setup For the ROI-oriented reconstruction task, we use the images and raw single- and multi-coil k-space data from the fastMRI+ knee dataset [338, 354], which contains bounding box annotations for knee pathologies. Specifically, we investigate the most common knee pathology in the dataset called "Meniscus Tear" (MT). Each image $\boldsymbol{x}_0$ in the dataset contains at least one rectangular bounding box annotation $\mathcal{R}_{\boldsymbol{x}_0}$, which is drawn to include all the pathology but exclude the normal surrounding anatomy [354]. Therefore, the local image quality within each bounding box (i.e., ROI) is more indicative of the quality for pathology assessment than a metric over the entire FOV. We emphasize that the location of the bounding box $\mathcal{R}_{\boldsymbol{x}_0}$ *varies sample by sample* and is *never* an input to any method during inference. $\mathcal{R}_{\boldsymbol{x}_0}$ is only used for calculating the training loss and evaluating the local PSNR during test time according to Equation (4.1). Hence, the local PSNR performance reflects the quality of reconstructions by different methods for assessing the considered pathological lesions in the ROIs.

Baselines We compare TACKLE$_{\text{ROI}}$ with three full-FOV reconstruction-oriented baselines.

- *LOUPE$_{FOV}$:* Proposed in [12], LOUPE$_{\text{FOV}}$ jointly optimizes a sampler and a residual U-Net reconstructor.
- *Low-pass + U-Net$_{FOV}$ (LP+UN$_{FOV}$):* substitute the sampler in LOUPE$_{\text{FOV}}$ with a fixed low-pass filter sampling pattern.
- *Poisson-disc + U-Net$_{FOV}$ (PD+UN$_{FOV}$):* substitute the sampler in LOUPE$_{\text{FOV}}$ with a Poisson-disc sampling pattern drawn from a variable density distribution

and generated with the `sigpy.mri.poisson` function in the `SigPy` package[1].

Results We compare the average local PSNR of our method and other baselines over the test set in Table 4.1. For all settings, TACKLE outperforms other baselines designed for full-FOV reconstruction by at least 3 dB, indicating a significant improvement of image quality within the ROI.

In Figure 4.3, we provide example reconstructions by our method and three baseline methods. For each reconstruction, a zoom-in on its ROI is provided at the bottom with the corresponding local PSNR value labeled above in orange, and its full-FOV PSNR is labeled on the top right corner. As shown in the ground truth of the MT example, a meniscus tear is indicated by a streak (dark in the top row and bright in the bottom row) that is present on the meniscus (bright in the top row and dark in the bottom row), as indicated by the red pointers. To accurately detect the existence and assess the severity of a meniscus tear, a reconstruction should clearly show the boundaries of the meniscus and the details of the tear. However, the ROIs of both LP+UN$_{\text{FOV}}$ and LOUPE$_{\text{FOV}}$ reconstructions contain significant reconstruction artifacts that disguise the tear (see the red arrows). On the other hand, TACKLE$_{\text{ROI}}$ preserves the details of the tear and contains fewer artifacts than the baselines, providing a more accurate ROI reconstruction with a higher diagnostic value.

In Appendix B.3 we also include a validation of TACKLE$_{\text{ROI}}$ on images that either are healthy or contain pathologies other than the meniscus tear. Although TACKLE$_{\text{ROI}}$ is not designed to generalize across different pathologies, we empirically find that TACKLE$_{\text{ROI}}$ still yields high-fidelity reconstructions for out-of-distribution images so that the pathologies on these images remain detectable. This generalization of TACKLE$_{\text{ROI}}$ is consistent across the three acceleration ratios (4×, 8×, and 16×) for this fastMRI+ dataset.

Discussion Enhancing local ROIs for MRI may seem counterintuitive because the acquisition happens in k-space; each frequency measurement in theory corresponds to the entire FOV. Here, we understand the feasibility via a PSF analysis. Consider the zero-filled reconstruction $\widetilde{\boldsymbol{x}}$ from some (noiseless) single-coil k-space data:

$$\widetilde{\boldsymbol{x}} := \boldsymbol{F}^{-1}\left(\boldsymbol{m} \odot (\boldsymbol{F}\boldsymbol{x})\right) = \left(\boldsymbol{F}^{-1}\boldsymbol{m}\right) * \boldsymbol{x}$$

[1] `https://github.com/mikgroup/sigpy` (BSD-3-Clause license)

Table 4.2: **Comparison of average test Dice score on the SKM-TEA dataset [83] for segmenting four knee tissues under different acceleration ratios (R).**

R	PD+UN$_{\text{recon.}}$	LOUPE$_{\text{recon.}}$	SemuNet	TACKLE$_{\text{recon.}}$	TACKLE$_{\text{seg.}}$
16	0.7843	0.7888	0.8108	0.8232	**0.8532**
64	0.7486	0.6715	0.7741	0.8145	**0.8357**

where $*$ denotes convolution and the second equality holds due to the Fourier convolution theorem. Here, $\boldsymbol{F}^{-1}\boldsymbol{m}$ is the PSF of the subsampling mask $\boldsymbol{m}$ and determines the resolution of the CS-MRI system. We visualize the PSF of a sampling mask trained for full-FOV reconstruction and another trained for MT ROIs reconstruction with the same sampling budget in Figure 4.4. We plot the PSF profiles in the vertical direction around the main lobes. The PSF learned for MT ROIs reconstruction has around 40% improvement in vertical resolution in terms of *full width at half maximum (FWHM)* of the PSF profiles. Since MT ROIs contain the thin horizontal anatomy of the meniscus, it makes sense that the learned subsampling pattern has a narrower PSF profile (and thus higher resolution) in the vertical direction. This comparison demonstrates that the improvement on ROIs is partly due to the capability of our model to optimize the subsampling PSF for local ROI anatomy via co-design. This is particularly beneficial when there is a mismatch between the optimal subsampling PSF for full-FOV reconstruction and that for ROI reconstruction due to directional anatomical structure, which is the case for MT ROI reconstruction.

4.4.2 Knee Tissue Segmentation

Dataset and Setup We then show the performance of TACKLE on solving a task that involves segmenting four types of knee tissues: the patellar cartilage, the femoral cartilage, the tibial cartilage, and the meniscus. We use the *Stanford Knee MRI with Multi-Task Evaluation (SKM-TEA)* dataset [83], which contains pixel-level segmentation maps of the four tissues. Specifically, we use the raw 3D multi-coil k-space measurements of knee images and take a 1D inverse Fourier transform along the left-to-right direction to obtain the 2D k-space of sagittal slices. We train each method to minimize the Dice loss until convergence and select the model with the highest Dice score on the validation set.

Baselines We compare TACKLE$_{\text{seg.}}$ with four baselines.

- *LOUPE$_{recon.}$:* LOUPE$_{\text{recon.}}$ is a baseline based on LOUPE$_{\text{FOV}}$. We first train

a LOUPE$_{\text{FOV}}$ model for the full-FOV reconstruction task and then use the reconstructed images to separately train a segmentation network.

- *Poisson-disc + U-Net$_{recon.}$ (PD+UN$_{recon.}$):* same as LOUPE$_{\text{recon.}}$ except that the sampler is fixed to be a Poisson-disc sampling mask.
- *TACKLE$_{recon.}$:* same as LOUPE$_{\text{recon.}}$ except for using the proposed architecture of TACKLE.
- *SemuNet:* Proposed in [309], SemuNet uses a hybrid of ℓ_1 reconstruction loss and cross-entropy segmentation loss.

Results We provide a quantitative comparison in Table 4.2 and a box-plot comparison in Figure 4.5. Within the rectangle between each pair of methods in Figure 4.5, the top number is the percentage of samples that get improved, and the bottom number is the p-value given by the paired samples t-test. With an improved architecture, TACKLE$_{\text{recon.}}$ already outperforms the other baselines. Nevertheless, the segmentation-oriented method TACKLE$_{\text{seg.}}$ achieves even better performance on both 16× and 64× accelerations. TACKLE$_{\text{seg.}}$ also significantly outperforms SemuNet on both acceleration ratios and has a much smaller performance drop from 16× to 64× than SemuNet, indicating that the proposed approach is more robust to high acceleration ratios. We further provide some visual examples in Figure 4.6. The first row visualizes the input of the predictor by different methods, where each image is labelled by its PSNR value on the top right corner. The last row shows the predicted segmentation maps by different methods, where each prediction is labelled by its Dice score on the top right corner. The blue arrows point out the locations where TACKLE$_{\text{seg.}}$ provides more accurate reconstructions than other reconstruction-oriented baselines. We also provide a zoom-in on the region that contains the segmented tissues in the second row.

Discussion We note that TACKLE$_{\text{seg.}}$ learns an intermediate feature map as the input to the predictor, which circumvents a typically "good" reconstruction. It is interesting how the retriever produces an image where different knee tissues to be segmented have distinctive textures, which are easy to distinguish both from the background and from each other. Even though this feature map is not a typical "reconstruction" in terms of pixel-wise accuracy, it still accurately localizes the anatomy of the tissues to be segmented. We highlight that TACKLE$_{\text{recon.}}$ provides a high-fidelity reconstruction of the entire FOV with a PSNR of 33.00 dB, which

Table 4.3: **Comparison of average test Dice score on the brain segmentation task under different acceleration ratios (R).**

R	PD+UN$_{\text{recon.}}$	LOUPE$_{\text{recon.}}$	SemuNet	TACKLE$_{\text{recon.}}$	TACKLE$_{\text{seg.}}$
16	0.8952	0.9244	0.9196	0.9350	**0.9395**
64	0.8377	0.8733	0.3824	0.9181	**0.9218**

demonstrates that our model is well capable of doing the full-FOV reconstruction task accurately. However, TACKLE$_{\text{seg.}}$ still outperforms TACKLE$_{\text{recon.}}$ in terms of segmentation performance in Figure 4.6 and on average over the dataset in Table 4.6 (see Section 4.6.1 for more details). This observation demonstrates that finding the most accurate full-FOV reconstruction does not necessarily lead to the optimal result on the considered segmentation task.

4.4.3 Brain Tissue Segmentation

Dataset and Setup We demonstrate TACKLE on another task that involves segmenting four brain tissues: the cortex, the white matter, the subcortical gray matter, and the cerebrospinal fluid (CSF). Following [135], we use the 109th coronal slice of each full k-space sampled volume in the OASIS dataset [213] and the segmentation maps generated with SAMSEG in FreeSurfer [105]. SAMSEG, which stands for Sequence Adaptive Multimodal SEGmentation, is an established method for brain tissue segmentation and is considered a standard method for this task [234]. We use the segmentation maps generated by SAMSEG as the supervised labels for training. We use the same measurement simulation procedure as in the tumor classification experiments. We simulate the single-coil k-space data for each image by taking the Fourier transform of the image and adding complex additive white Gaussian noise (AWGN), according to the forward model in Equation (1). The standard deviation of the noise for each image is 0.05% of the magnitude of the DC component. We train each method to minimize the Dice loss until convergence and select the model with the highest Dice score on the validation set.

Baselines We compare TACKLE$_{\text{seg.}}$ with the same baselines as the ones in Section 4.4.2: LOUPE$_{\text{recon.}}$, PD+UN$_{\text{recon.}}$, TACKLE$_{\text{recon.}}$, and SemuNet.

Results We first provide a numerical comparison in Table 4.3 and a box-plot comparison in Figure 4.7. Within the rectangle between each pair of methods in Figure 4.7, the top number is the percentage of improved samples, and the bottom

number is the p-value given by the paired samples t-test. With an improved architecture, TACKLE$_{\text{recon.}}$ significantly outperforms the other reconstruction-oriented baselines. Nevertheless, TACKLE$_{\text{seg.}}$ still outperforms TACKLE$_{\text{recon.}}$ under both accelerations with significant p-values, highlighting the benefit of task-specific training. Compared to SemuNet [309], TACKLE$_{\text{seg.}}$ learns better segmentation strategies for both acceleration ratios and is more robust to high acceleration. We further provide some visual examples in Figure 4.8, visualizing the input and output of the predictor across different methods. The zoom-in regions highlight a location where the segmentation prediction of TACKLE$_{\text{seg.}}$ outperforms other baselines. Specifically, TACKLE$_{\text{seg.}}$ more accurately predicts the outline of the white matter (in yellow) than other methods. Such an improvement leads to more precise estimation of the thickness of the cortex (in orange), an important task for studying human cognition and neurodegeneration [10].

4.4.4 Pathology (Tumor) Classification

Dataset and Setup In this section, we demonstrate the effectiveness of the proposed method at detecting the existence of gliomas, a common type of brain tumor in adults. We use the images acquired by the FLAIR sequence in the Multimodal Brain Tumor Image Segmentation Benchmark (BRATS) dataset [219]. To obtain an image-level label of the existence of a tumor, we aggregate the pixel-level peritumoral edema (ED) segmentation annotations in the BRATS dataset by checking whether there exists any positive pixel in the segmentation map: negative (healthy) means there is no ED pixel, while positive (unhealthy) means there is at least one ED pixel. We simulate the single-coil k-space data for each image by taking the Fourier transform of the image and adding complex additive white Gaussian noise (AWGN), according to the forward model in Equation (2.1). The standard deviation of the noise for each image is 0.05% of the magnitude of the DC component. We train all models using the BCE loss and evaluate them using the classification accuracy and F_1 score as described in Section 4.3.3.3.

Baselines We compare the proposed method TACKLE$_{\text{class.}}$ with the first three baselines as in Section 4.4.2 and Section 4.4.3 except that the predictor of each baseline is subsequently trained for pathology classification rather than tissue segmentation (with input images optimized for full-FOV reconstruction). We do not include SemuNet here because it was originally proposed for the segmentation task only.

Table 4.4: **Comparison of average test accuracy on the pathology classification task under different acceleration ratios (R).**

Metric	R	PD+UN$_{\text{recon.}}$	LOUPE$_{\text{recon.}}$	TACKLE$_{\text{recon.}}$	TACKLE$_{\text{class.}}$
Cls. acc.	16	0.9016	0.9024	0.9062	**0.9159**
	64	0.8809	0.8930	0.9054	**0.9136**
F_1 score	16	0.8853	0.8846	0.8929	**0.9039**
	64	0.8628	0.8768	0.8910	**0.8992**

Results In Table 4.4, we compare TACKLE$_{\text{class.}}$ with reconstruction-oriented baselines, and find that TACKLE$_{\text{class.}}$ achieves higher classification accuracy under both performance metrics. Specifically, TACKLE$_{\text{class.}}$ outperforms the existing reconstruction-oriented baseline LOUPE$_{\text{recon.}}$ by around 2% in the extreme 64× accelerated acquisition scenario. Both variants of TACKLE maintain competitive performance under the highly accelerated setting (R=64), while PD+UN$_{\text{recon.}}$ and LOUPE$_{\text{recon.}}$ suffer from significant performance degradation. Note that TACKLE$_{\text{class.}}$ outperforms TACKLE$_{\text{recon.}}$ by more than 0.8% in both cases, despite having the same architecture. We also visualize and compare the classification performance of TACKLE$_{\text{class.}}$ and LOUPE$_{\text{recon.}}$ under 16× acceleration in Figure 4.9, using confusion matrices. The results show that TACKLE$_{\text{class.}}$ has substantially fewer false negatives (bottom left) and a higher overall accuracy compared to LOUPE$_{\text{recon.}}$.

4.5 Validation on an Experimentally Collected Out-of-Distribution Dataset

In practice, creating a large, well-annotated training set for a specific task can be very time-consuming or even infeasible. To demonstrate the immediate benefit of our method in a real-world setting, we conduct a validation of TACKLE on the ROI-oriented reconstruction task using experimentally collected data that is out of the distribution of the training data. Specifically, we train a TACKLE model on a large-scale dataset (fastMRI in this case) and directly test it on raw k-space data collected by *different hardware using a different type of sequence* from that of the training. Even without extra fine-tuning or test-time optimization, the learned ROI-specific model provides improved reconstructions on meniscus ROIs. In the following subsections, we present the details of this experiment.

Data Acquisition and Processing Two subjects were scanned at the Massachusetts General Hospital in accordance with institutional review board guidelines. Their right knees were scanned by a 3D-encoded Cartesian gradient-echo sequence with a

3 Tesla MRI scanner (Model: Skyra; Siemens Healthcare, Erlangen, Germany) and a single-channel extremity coil. To implement the 2D subsampling pattern in the coronal plane, we used a transversal orientation with the frequency encoding direction (k_x) pointing into the knee cap (anterior-posterior), so that the two phase encoding directions were left-right (k_y) and superior-inferior (k_z), respectively. The acquisition parameters were as follows: TE/TR=4.8/9.1ms, FOV=192×192×192mm^3, resolution=1×1×1mm^3, flip angle=10°. The total acquisition time of obtaining the fully sampled data for each subject was 5 minutes and 35 seconds. The raw k-space data has the shape of 192×192×192 ($k_x \times k_y \times k_z$). We apply the 1D inverse Fourier transform along k_x for downstream processing. Specifically, we take the middle 40 slices of each volume and annotated bounding boxes around the meniscus region using an image labelling tool[2]. Efforts were made such that the locations and sizes of the bounding boxes roughly match those in the fastMRI MT dataset. We emphasize that these bounding boxes are *only* for the purpose of measuring the accuracy of different models on reconstructing the meniscus region. The locations of the annotated ROIs are *not* the input to any of the tested models.

Generalization Gaps There are multiple generalization gaps between the training (fastMRI single-coil data) and the test data:

- *Different hardware:* The acquired data are collected directly with a single-channel extremity coil, while the training data are simulated from k-space data collected by multi-channel receive coils [165].
- *Different sequence and resolution:* The acquired data are given by a gradient-echo sequence with 1 mm isotropic resolution, while the training data are given by a spin-echo sequence with 0.5mm in-plane resolution [165].
- *Different distribution of the ROI anatomy:* The acquired data are collected from two subjects whose menisci are healthy and have no tears, while the ROIs in the training data contain meniscus tears.

Despite these generalization gaps, TACKLE$_{\text{ROI}}$ works robustly and leads to both numerical and visual improvement.

Baselines In this section, we compare TACKLE$_{\text{ROI}}$ with the following baselines under 4× acceleration.

[2] `https://github.com/heartexlabs/labelImg` (MIT license)

Table 4.5: **Comparison of average reconstruction accuracy on the experimentally collected dataset under 4× acceleration (top: full-FOV reconstruction; bottom: ROI-oriented reconstruction).**

Full-FOV recon.	PD+TV$_{\text{FOV}}$	LOUPE$_{\text{FOV}}$	TACKLE$_{\text{FOV}}$	TACKLE$_{\text{ROI}}$
PSNR (dB)	27.94	28.00	**28.70**	28.18

ROI recon.	PD+TV$_{\text{FOV}}$	LOUPE$_{\text{FOV}}$	TACKLE$_{\text{FOV}}$	TACKLE$_{\text{ROI}}$
Local PSNR (dB)	24.45	24.67	25.16	**25.72**

▭ indicates the variant of TACKLE with matching training and evaluation metrics

- *Poisson-disc + Total Variation$_{FOV}$ (PD+TV$_{FOV}$):* The subsampling pattern is the same as the Poisson-disc sampling pattern generated by `sigpy.mri.poisson` for PD+UN$_{\text{FOV}}$ in Section 4.4.1. The reconstruction is obtained by solving a total variation (TV) regularized optimization problem with the Sparse MRI toolbox[3].
- *LOUPE$_{FOV}$:* the same LOUPE$_{\text{FOV}}$ baseline as in Section 4.4.1.
- *TACKLE$_{FOV}$:* a TACKLE model trained for full-FOV reconstruction.
- *LOUPE$_{ROI}$:* the same architecture as LOUPE$_{\text{FOV}}$ but trained for ROI reconstruction following the same training procedure as TACKLE$_{\text{ROI}}$.

Results We present a quantitative comparison in Table 4.5. For both the full-FOV and ROI-oriented reconstruction tasks, TACKLE outperforms the baselines under the corresponding metric. For each task, we highlight the variant of TACKLE trained for the evaluation metric in green. Our results show that the highlighted variant outperforms the other variant of TACKLE, indicating a tradeoff between full-FOV and ROI reconstruction accuracy.

We further conduct a slice-wise PSNR analysis in Figure 4.10. For both histograms, the horizontal axis is the improvement on the respective metric, and the vertical axis is the count. We also quantify the significance of the improvements using the paired samples t-test. For the full-FOV reconstruction, TACKLE$_{\text{FOV}}$ outperforms LOUPE$_{\text{FOV}}$ on *all* 80 slices, giving a highly significant p-value of 3.10e-57. We then compare TACKLE$_{\text{ROI}}$ with the better full-FOV reconstruction method, TACKLE$_{\text{FOV}}$,

[3] `https://people.eecs.berkeley.edu/~mlustig/Software.html` (unknown license)

on the ROI-oriented reconstruction task. Despite having the same architecture, TACKLE$_{\text{ROI}}$ still outperforms TACKLE$_{\text{FOV}}$ on 72.5% of slices, leading to a p-value of 5.12e-8, which is also statistically significant. This result indicates that the ROI-oriented model TACKLE$_{\text{ROI}}$ indeed provides more accurate ROI reconstructions on this out-of-distribution dataset. We further provide some visual examples in Figure 4.11. Below each reconstruction is a zoom-in on the region around the ROI and the error map of the region with respect to the ground truth. TACKLE not only achieves higher PSNR values in both cases but also visually recovers the ROIs with fewer artifacts.

Implementation Besides the above results based on retrospective subsampling for quantitative comparison, we have also tested the learned sequence on a Siemens 3T MRI Skyra scanner. Specifically, we implement a re-ordering loop that iterates through all the trajectories based on our learned subsampling mask $\boldsymbol{m}$. The implemented sequence prospectively subsamples in k-space and shortens the scan time from 335 seconds to 84 seconds. In Figure 4.12, we compare the reconstruction given by the prospectively subsampling sequence we implement with the reconstruction given by the retrospectively subsampled measurements from the fully sampling sequence. We note that the images labelled as "TACKLE$_{\text{ROI}}$ (retrospective)" and "TACKLE$_{\text{ROI}}$ (prospective)" are taken by two consecutive but separate scans, so there might be some subtle motion between them. Nevertheless, the two images have no significant visual difference, indicating that the improvement we show on retrospective simulations translates into actual improvement in practice. The prospective reconstruction successfully recovers important anatomical features around the meniscus region while only taking a quarter of the scan time compared to the full-sampled image.

4.6 Ablation Studies

4.6.1 Effectiveness of Co-Design

We evaluate the effectiveness of two aspects of co-design used in the proposed framework: learnable subsampling and task-specific training. In Table 4.6, we compare four variants of the proposed method that have neither, one, or both aspects of co-design. The meanings of having or not having each aspect are summarized as follows:

- Learnable subsampling (column 2)

Table 4.6: **Ablation studies on two aspects of co-design for all the considered tasks under 16× acceleration.**

Method	Ablated component		ROI-oriented reconstruction (Local PSNR in dB)		Tissue segmentation (Dice score)		Pathology classification (Cls. acc.)	(F_1 score)
	Learned subsampling	Task-specific training	Single-coil	Multi-coil	Knee	Brain	Gliomas tumor	
PD+VN$_\flat$	✗ (Poisson-disc)	✗	29.91	36.48	0.8018	0.9257	0.9024	0.8871
PD+VN$_\sharp$	✗ (Poisson-disc)	✓	30.15	36.51	0.8474	0.9256	0.9072	0.8966
TACKLE$_\flat$	✓	✗	30.14	37.53	0.8232	0.9350	0.9062	0.8929
TACKLE$_\sharp$	✓	✓	**31.54**	**37.89**	**0.8532**	**0.9395**	**0.9159**	**0.9039**

$\flat$ indicates full-FOV reconstruction oriented versions of PD+VN and TACKLE
$\sharp$ indicates task-specific versions of PD+VN and TACKLE

✗ *(Poisson-disc)*: use a Poisson-disc subsampling pattern that is randomly generated and then fixed

✓: learn the subsampling pattern from data

- Task-specific training (column 3)

  ✗: separately optimize retriever and predictor

  ✓: jointly optimize retriever and predictor

To eliminate the effect of different network architectures, all four variants have exactly the same architectures. Overall, we find both aspects of co-design are beneficial. For the task of reconstructing meniscus tear ROIs, learning the subsampling pattern is particularly helpful. Task-specific training, on the other hand, is more important for the knee segmentation task. Highlighted in cyan, the last row is the full-fledged version of TACKLE, which achieves the best performance for all considered scenarios with both aspects of co-design.

4.6.2 Effectiveness of the Proposed Architecture and Training Procedure

The proposed architecture of the mapping from measurements $\boldsymbol{y}$ to prediction $\widehat{\boldsymbol{z}}$, which we denote as $\mathsf{T}_{\boldsymbol{\psi}}$, consists of an E2E-VarNet retriever and a U-Net predictor. A natural question is how this architecture compares with a single model-free neural network with a comparable number of parameters that directly maps subsampled measurements to the final prediction. We consider the following comparisons in Table 4.7:

- Single larger predictor (row 1)
  - *Tissue seg.*: U-Net with 128 channels after the first convolution layer and the same number of pooling layers (42.2M parameters)

Table 4.7: **Ablation studies on model architecture and pre-training for non-reconstruction tasks under 16× acceleration.**

Ablated component		Tissue segmentation (Dice score)		Pathology classification (F_1 score)	(Cls. acc.)
Arch. of T_ψ	Pre-train	Knee	Brain	Gliomas tumor	
Predictor only‡	✗	0.7539	0.9005	0.8966	0.8788
VN+predictor‡	✗	0.8163	0.9371	0.9102	0.8969
VN+predictor§	✓	**0.8532**	**0.9395**	**0.9159**	**0.9039**

‡ U-Net(128) / ResNet(101) for tissue seg. / patho. class.
§ E2E-VarNet + U-Net(64) / ResNet(18) for tissue seg. / patho. class.

Table 4.8: **Comparison of average test PSNR (dB) between reconstruction models trained with task-specific masks and TACKLE$_{\text{recon.}}$ on the fastMRI knee dataset.**

Method	Brain seg.		Knee seg.		Tumor class.	
	16×	64×	16×	64×	16×	64×
Task-specific mask+VN	**38.47**	33.04	32.53	30.10	44.20	37.07
TACKLE$_{\text{recon.}}$	38.44	**33.13**	**32.63**	**30.24**	**44.48**	**37.26**

  - *Patho. class.*: ResNet101 (42.5M parameters)

- VN+predictor (rows 2&3)
  - *Tissue seg.*: E2E-VarNet + standard U-Net (29.9M + 10.6M = 40.5M parameters)
  - *Patho. class.*: E2E-VarNet + ResNet18 (29.9M + 11.2M = 41.1M parameters)

Comparing the first two rows, we find that the proposed "VN+predictor" architecture significantly outperforms the "single larger predictor" baseline on all settings. This is likely due to the model-based nature of the "VN+predictor" architecture, which more effectively extracts useful information from subsampled measurements for downstream tasks. Finally, we include the pre-training step discussed in Section 4.3.4. Highlighted in cyan, the full-fledged version of TACKLE in the last row significantly outperforms the ablated baselines on both non-reconstruction tasks, indicating the importance of both the proposed architecture and training procedure.

4.6.3 Using Task-specific Sequences for Reconstruction

Our optimized task-specific pipeline learns to adjust the image representation from a conventional form to one that is more readily interpretable by the predictor network. This often adds additional textures to the images, making them look different from traditional reconstructions. However, this does not imply there is a significant loss in information that could be used for image reconstruction. Despite being optimized for task-specific objectives, our learned task-specific subsampling patterns can be used retrospectively for generating high-fidelity reconstructions. To show this, we conduct an experiment where we take the learned subsampling patterns of TACKLE$_{\text{seg.}}$ and TACKLE$_{\text{class.}}$ and train an additional reconstruction network for each subsampling pattern. The subsampling pattern is fixed during the training. This experiment mimics the scenario if one wants a traditional reconstruction out of the collected k-space samples from our task-specific sequences. In Table 4.8, we provide a comparison with TACKLE$_{\text{recon.}}$, which jointly optimizes the subsampling pattern and reconstructor, on the fastMRI knee dataset. One can see that the reconstruction models trained with task-specific masks (row 1) come close to TACKLE$_{\text{recon.}}$ (row 2) in terms of reconstruction performance. These results indicate that our task-specific models do not incur a significant loss of image information but achieve a better trade-off for the downstream task accuracy. It is thus possible to recover better images retrospectively using the k-space measurements collected by the task-specific sequences.

4.7 Limitations

Building on the promising results we have achieved, we acknowledge opportunities for further improvement of our current study.

Data Usage Similar to other works on task-specific CS-MRI co-design, our approach requires matched k-space, image, and annotation labels, which are of limited quantity in the research community. Due to this limitation, two of our experiments (brain segmentation and tumor classification tasks) are conducted with k-space data simulated from magnitude images.

Sequence Implementation Although we have implemented a prospectively subsampling sequence with a learned sampling pattern by TACKLE$_{\text{ROI}}$ on a Siemens MRI scanner, it was done using only one type of 3D gradient echo sequence. Other physical constraints affect the deployment of our method for general MRI sequences.

For example, in spin-echo sequences, the order of sampling should be considered to mitigate spin-relaxation effects.

Controlled Study The evaluation in the current study is based on conventional quantitative metrics and qualitative visual comparisons. The number of volunteers for testing our learned sequences on a Siemens MRI scanner is relatively small. To further assess prospective subsampling, future evaluations should involve controlled studies of image quality with radiologists.

4.8 Conclusion

In this chapter, we generalized the objective of CS-MRI co-design to a variety of tasks beyond full-FOV reconstruction. We introduced TACKLE as a unified approach for robustly learning task-specific strategies. Through comprehensive experiments, we showed that TACKLE outperforms existing DL techniques that separately learn subsampling pattern, reconstruction, and prediction. Additionally, TACKLE outperforms naive approaches to co-design that directly learn mappings from measurements to predictions. We found that the optimized strategies sometimes circumvent the typical reconstruction in terms of pixel-wise accuracy, but effectively extract key visual information useful for task prediction. Through ablation studies, we justified multiple design choices about architecture and training procedure, and showed their importance in effectively learning CS-MRI strategies for tasks that go beyond full-FOV reconstruction. We further implemented a learned subsampling sequence and tested it on a Siemens 3T MRI Skyra scanner, which led to a four-fold scan time reduction without sacrificing visual quality. Our study demonstrates the exciting promise of employing end-to-end co-design techniques, suggesting a future where clinical CS-MRI requirements are addressed with enhanced efficiency while maintaining accuracy.

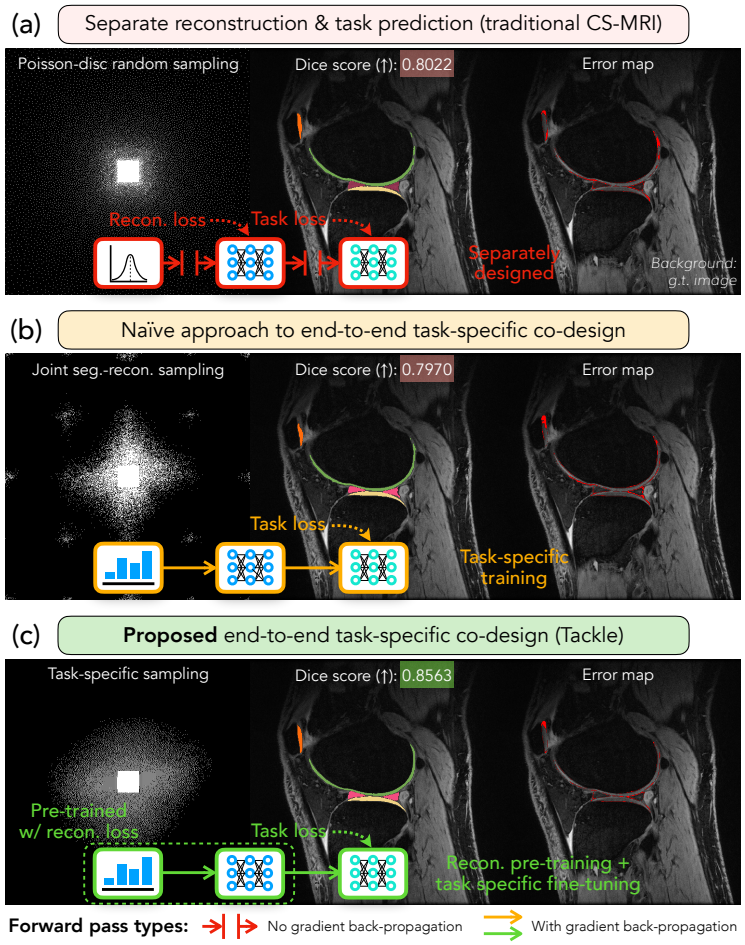


Figure 4.1: **Comparison between (a) traditional CS-MRI, (b) a naïve approach to task-specific CS-MRI, and (c) the proposed Tackle framework.** Compared with panel (a) which separately deals with reconstruction and task prediction, panel (b) is a simple extension of co-design methods for solving downstream tasks by adding a learnable mapping from measurements to task predictions. However, this naïve approach leads to a suboptimal performance and can even lead to a worse task prediction accuracy, as shown in the example above. On the other hand, we introduce Tackle for effectively learning task-specific CS-MRI strategies. Tackle is first pre-trained for generic reconstruction, and then all three modules are fine-tuned for a more specific downstream task. We find that this training schedule allows Tackle to robustly learn generalizable task-specific strategies. In the above knee segmentation example, all three approaches are trained with the same architectures for the reconstructor (second module) and predictor (third module). Nevertheless, Tackle significantly outperforms the two baseline approaches.

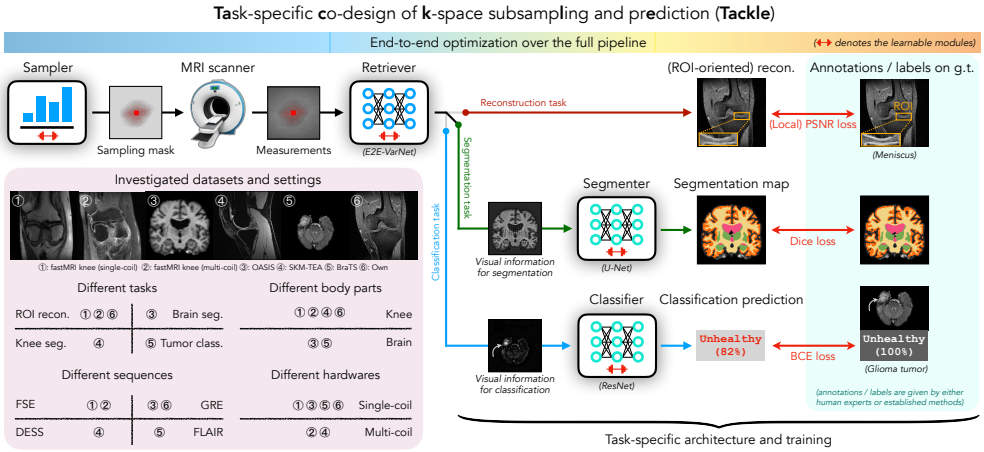


Figure 4.2: **Block diagram of the proposed framework TACKLE and a summary of the investigated datasets and settings.** TACKLE uses a task-specific loss to jointly optimize a sampler, a retriever, and an optional predictor, ranging from scanner-level sampling to human-level diagnosis. A summary of the investigated settings is presented in the bottom left panel. FSE, GRE, DESS, and FLAIR stand for fast spin echo, gradient echo, double-echo steady-state, and fluid-attenuated inversion recovery, respectively. We comprehensively investigate multiple CS-MRI tasks on a variety of common MRI settings with six datasets.

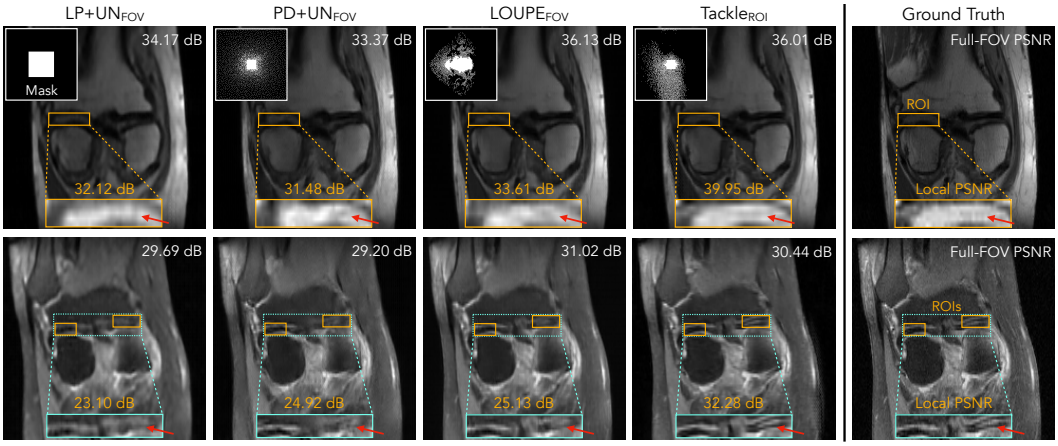


Figure 4.3: **Visual examples of two Meniscus Tear samples reconstructed by different methods in the 16× acceleration single-coil setting.** For each reconstruction, the full-FOV PSNR is labeled in white, and the local PSNR for the ROI is in orange. Note how TACKLE$_{ROI}$ recovers the structure and details of the ROI more accurately than the two baselines, as indicated by the red arrows. The better recovery of TACKLE$_{ROI}$ over the ROI leads to a more accurate diagnosis of the Meniscus Tear. We emphasize that the location of the ROI is not an input to any of these models and is only used for evaluating the accuracy of each method on the region that contains the pathology.

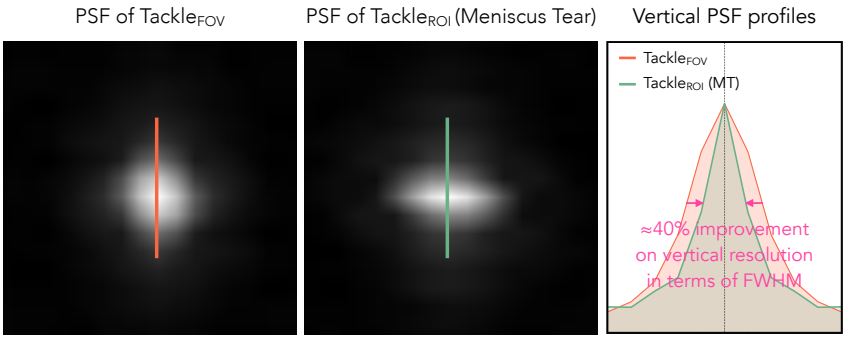


Figure 4.4: **Comparison of a subsampling PSF optimized for full-FOV reconstruction and another optimized for the reconstruction of menicus tear (MT) ROIs.** Optimizing for MT ROI reconstruction leads to around 40% improvement on the vertical resolution in terms of the *full width at half maximum (FWHM)*, as shown by the PSF profiles in the bottom panel. This improved vertical resolution leads to a better reconstruction of the meniscus that has horizontal anatomy.

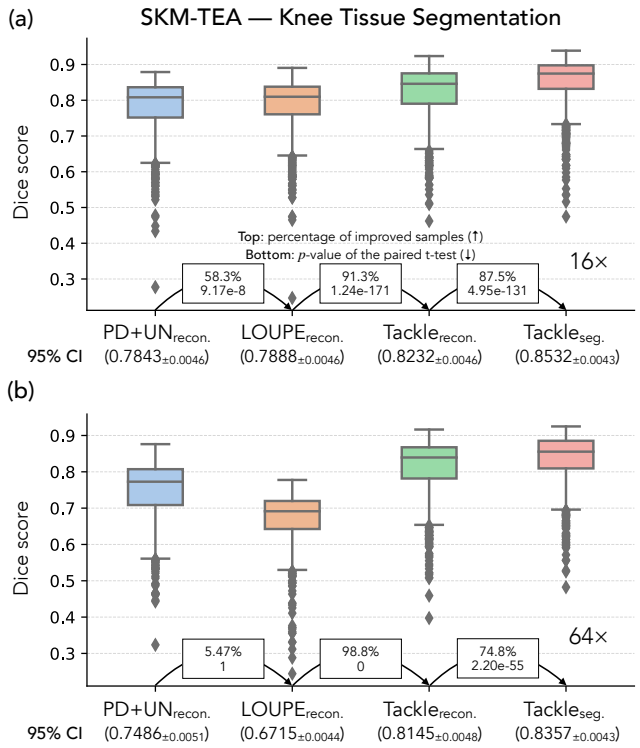


Figure 4.5: **Box plots of the knee tissue segmentation results under 16× (a) and 64× (b).** Within the rectangle between each pair of methods, the top number is the percentage of samples that get improved, and the bottom number is the p-value given by the paired samples t-test. A higher percentage and a lower p-value indicate a more significant improvement. We also provide the 95% confidence intervals for all methods below their names. For both acceleration ratios, TACKLE$_{seg.}$ outperforms other baselines in terms of all the statistical measures.

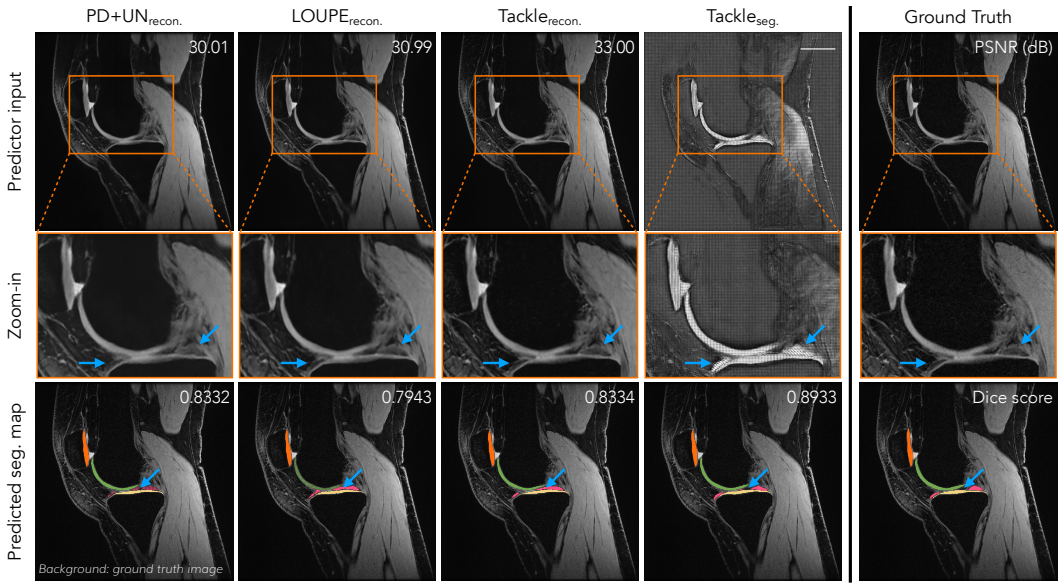


Figure 4.6: **Comparison of segmentation results under 16× acceleration on one sample of the SKM-TEA dataset.** We show the input of the predictor in the first row, a zoom-in on the region that contains the tissues to be segmented in the second row, and the output of the predictor in the third row. Note that TACKLE$_{\text{seg.}}$ circumvents the typical "reconstruction" in terms of pixel-wise similarity with the ground truth image. Instead, it learns a feature map that accurately localizes the anatomy, leading to better segmentation prediction than other baselines both for this sample and on average over the test set (Table 4.2).

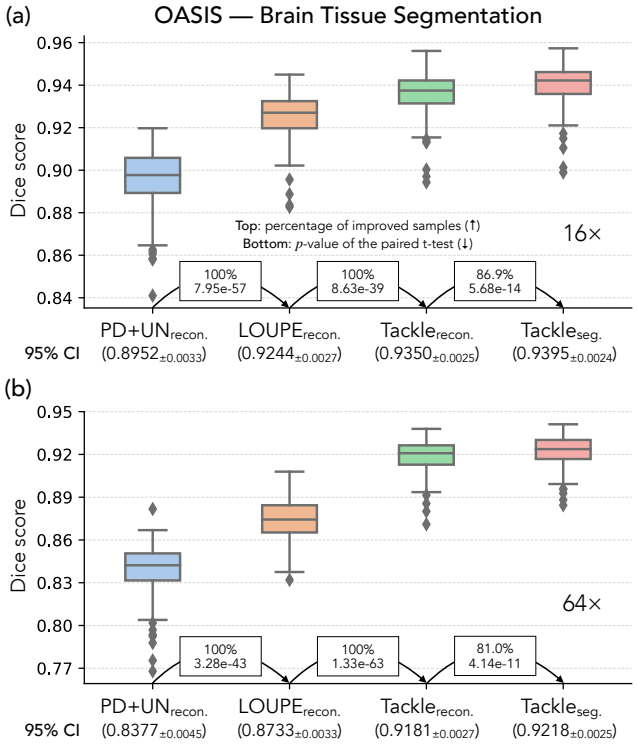


Figure 4.7: **Box plots of the brain tissue segmentation results under 16× (a) and 64× (b) accelerations.** Within the rectangle between each pair of methods, the top number is the percentage of samples that get improved, and the bottom number is the p-value given by the paired samples t-test. A higher percentage and a lower p-value indicate a more significant improvement. We also provide the 95% confidence intervals for all methods below their names. Similar to the knee segmentation results, the proposed method TACKLE outperforms other baselines in terms of all the statistical measures for both acceleration ratios.

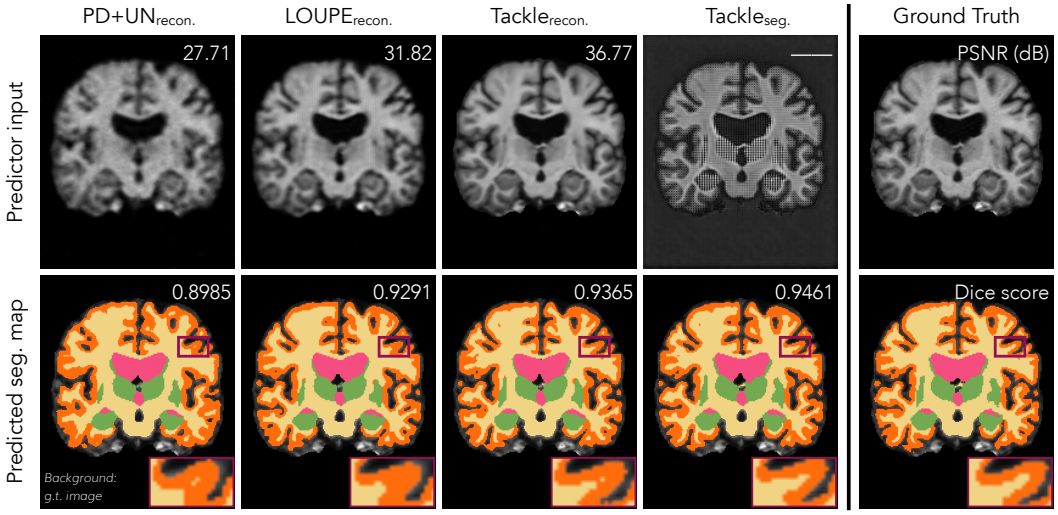


Figure 4.8: **Comparison of segmentation results under 16× acceleration on one sample from the OASIS dataset.** Similar to the knee segmentation results, TACKLE$_{\text{seg.}}$ circumvents the typical "reconstruction" in terms of pixel-wise similarity with the ground truth image. Instead, it learns an anatomically accurate feature map, which enables better segmentation prediction than other baselines both for this sample and on average over the test set (Table 4.3). The zoom-in panels highlight a region where TACKLE$_{\text{seg.}}$ more accurately predicts the outline of white matter (in yellow) than other methods. This improvement leads to a more precise estimation of the thickness of the cortex (in orange), an important task for studying human cognition and neurodegeneration [10].

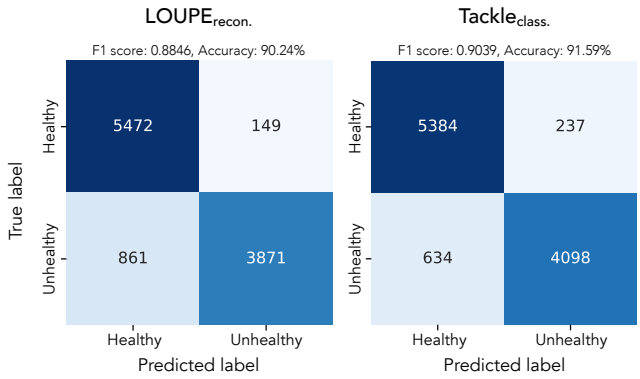


Figure 4.9: **Confusion matrices of the classification results by LOUPE$_{\text{recon.}}$ and TACKLE$_{\text{class.}}$.** Overall, TACKLE$_{\text{class.}}$ achieves greater accuracy in terms of both classification accuracy and F_1 score than LOUPE$_{\text{recon.}}$. TACKLE$_{\text{class.}}$ also has a significantly lower number of false negatives (bottom left) compared to LOUPE$_{\text{recon.}}$, which could lead to more patients receiving early treatment.

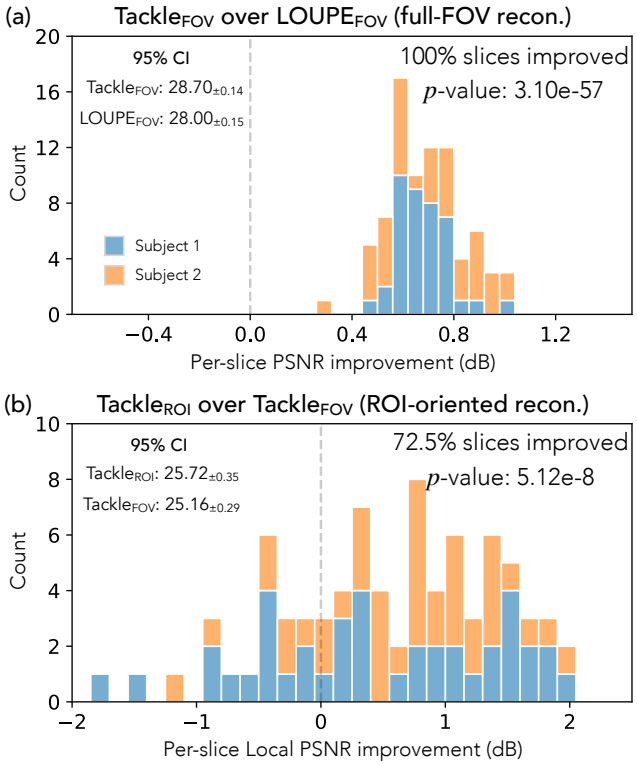


Figure 4.10: **Per-slice difference histograms.** (a): TACKLE$_{FOV}$ over LOUPE$_{FOV}$ on the full-FOV reconstruction task and (b): TACKLE$_{ROI}$ over LOUPE$_{FOV}$ on the ROI-oriented reconstruction task. The 95% confidence intervals are given in the top left corner of each plot. In both cases, the vast majority of slices improve, and the p-values given by the paired samples t-test are highly significant.

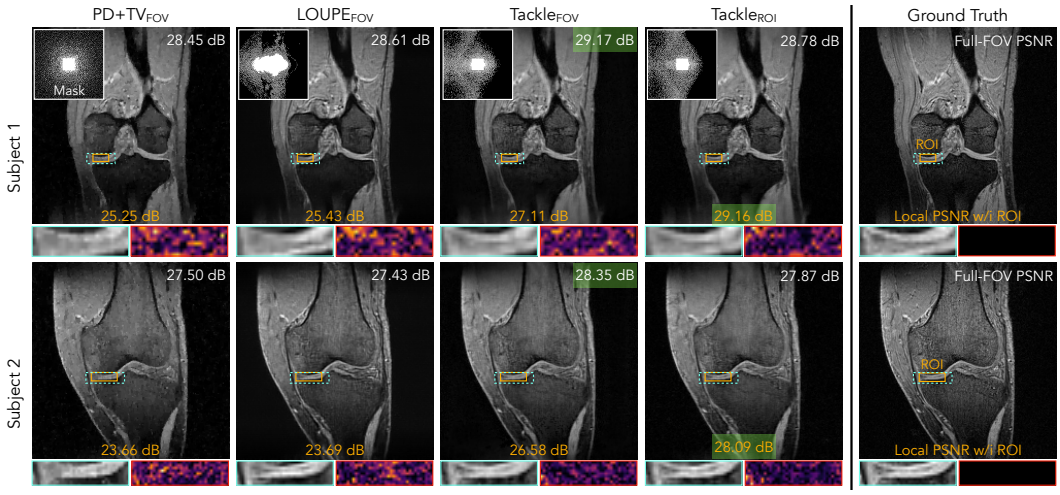


Figure 4.11: **Reconstruction comparison of two samples in the experimentally collected dataset (top: from subject 1; bottom: from subject 2) by different methods under 4× acceleration.** The sampling mask, a zoom-in on the ROI, and the error map are presented for each method. By sampling more frequencies along the vertical direction in k-space, TACKLE$_{ROI}$ has a higher vertical resolution in the image space and thus outperforms other baselines optimized for full-FOV reconstruction on the ROIs with directional anatomical structure.

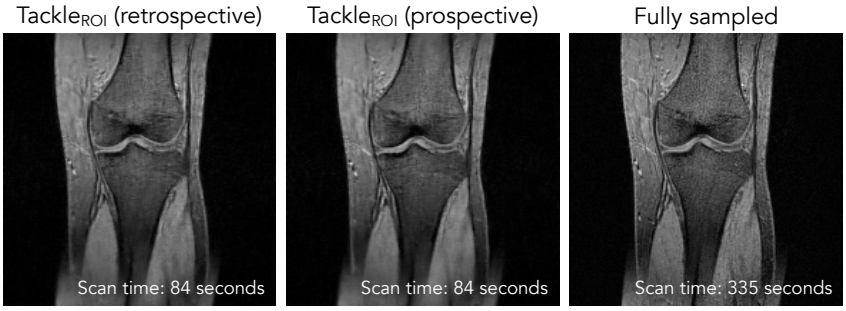


Figure 4.12: **Reconstruction comparison between the implemented prospective subsampling sequence and the retrospective subsampling sequence.** Our learned sequence can be implemented on an MRI scanner and generates images of quality indistinguishable from those recovered from retrospectively sampled data. Compared to the ground truth image, our prospectively subsampled reconstruction recovers important features around the meniscus region, which is the ROI it is trained to enhance.

Part II

Sampling: Posterior Estimation

Chapter 5

OVERVIEW AND PRELIMINARIES

In this part, we investigate the second topic around *sampling* in computational imaging—posterior estimation—in the context of Bayesian inverse problems.

5.1 Bayesian Inverse Problems

The Bayesian formulation provides a principled framework for solving inverse problems via a probabilistic viewpoint [268]. We refer to this formulation as Bayesian Inverse Problems (BIPs) and formally introduce it in this section.

5.1.1 Basics

Recall from Chapter 1 that we consider inverse problems of reconstructing $\widehat{\boldsymbol{x}} \approx \boldsymbol{x}_0$ from measurements

$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}_0) + \boldsymbol{n}, \tag{5.1}$$

where $\mathcal{A}$ is the forward model and $\boldsymbol{n}$ is the noise term. Assume that $\boldsymbol{n}$ is a random variable with density π. Then the probability density of observing measurements $\boldsymbol{y}$ given $\boldsymbol{x}_0$ is

$$p(\boldsymbol{y} \mid \boldsymbol{x}_0) := \pi(\boldsymbol{y} - \mathcal{A}(\boldsymbol{x}_0)) = \pi(\boldsymbol{n}), \tag{5.2}$$

commonly referred to as the *data likelihood* (or likelihood in short). Let $p(\boldsymbol{x}_0)$ denote the *prior distribution* (or prior in short) of $\boldsymbol{x}_0$, which encodes our knowledge about $\boldsymbol{x}_0$ before any measurements are acquired. According to Bayes' theorem, the *posterior distribution* (or posterior in short) of $\boldsymbol{x}_0$ given $\boldsymbol{y}$ is

$$p(\boldsymbol{x}_0 \mid \boldsymbol{y}) = \frac{p(\boldsymbol{x}_0, \boldsymbol{y})}{p(\boldsymbol{y})} = \frac{p(\boldsymbol{y} \mid \boldsymbol{x}_0)p(\boldsymbol{x}_0)}{p(\boldsymbol{y})} = \frac{p(\boldsymbol{y} \mid \boldsymbol{x}_0)p(\boldsymbol{x}_0)}{\int p(\boldsymbol{y} \mid \boldsymbol{x}_0)p(\boldsymbol{x}_0)\mathrm{d}\boldsymbol{x}_0}, \tag{5.3}$$

where the denominator $p(\boldsymbol{y})$ is the *model evidence* (or evidence in short)—a normalization constant ensuring that the posterior is a valid probability distribution. The posterior distribution represents our updated belief about $\boldsymbol{x}_0$ after incorporating the information provided by the measurements $\boldsymbol{y}$. Solving BIPs amounts to generating samples from the posterior distribution, i.e., sampling $\widehat{\boldsymbol{x}} \sim p(\boldsymbol{x}_0 \mid \boldsymbol{y})$.

Despite being a standard Bayesian inference problem theoretically, solving BIPs in computational imaging still faces several challenges:

- In imaging applications, the unknown target $\boldsymbol{x}_0$ often represents a high-resolution image or video. It is common for $\boldsymbol{x}_0$ to have thousands, millions, or even higher dimensions. Characterizing a high-dimensional distribution $p(\boldsymbol{x}_0)$ of images or videos is hard in the first place.

- The posterior distribution involves the normalizing constant (evidence) $p(\boldsymbol{y})$, which is an integration over all candidates $\boldsymbol{x}_0$'s that could potentially lead to the measurements $\boldsymbol{y}$. Evaluating this term is almost impossible in real-world problems due to the high dimensionality and lack of a closed form for $p(\boldsymbol{x}_0)$. It is a challenge to estimate $p(\boldsymbol{x}_0 \mid \boldsymbol{y})$ without access to $p(\boldsymbol{y})$.

- Equation (5.3) only elucidates the connection among prior, posterior, likelihood, and evidence in terms of numerical values. It is non-trivial how to design a sampler that actually provides samples from the posterior distribution, especially for high-dimensional problems.

- In many real-world applications, the measurements $\boldsymbol{y}$ only provides limited information for $\boldsymbol{x}_0$. Successful recovery of $\boldsymbol{x}_0$ relies on a sophisticated prior $p(\boldsymbol{x}_0)$ that captures the set of possible solutions. One important challenge in computational imaging is to design and leverage priors with sufficient expressiveness for posterior estimation.

5.1.2 Maximum a Posteriori Estimation

Traditional approaches to Bayesian inverse problems circumvent these challenges by finding the maximum a posteriori (MAP) estimator. Instead of sampling the full posterior distribution, the MAP approach aims to maximize the (logarithmic) posterior, i.e., finding

$$\widehat{\boldsymbol{x}}_{\mathrm{MAP}} := \arg\max_{\boldsymbol{x}} \log p(\boldsymbol{x} \mid \boldsymbol{y}) = \arg\max_{\boldsymbol{x}} \left[\log p(\boldsymbol{y} \mid \boldsymbol{x}) + \log p(\boldsymbol{x})\right]. \tag{5.4}$$

Note that the intractable evidence term is eliminated because it does not depend on $\boldsymbol{x}$. Assuming Gaussian noise with zero mean and covariance $\boldsymbol{\Sigma}$, we have that

$$\log p(\boldsymbol{y} \mid \boldsymbol{x}) = -\frac{1}{2}\|\mathcal{A}(\boldsymbol{x}) - \boldsymbol{y}\|^2_{\boldsymbol{\Sigma}^{-1}} + C \tag{5.5}$$

where $\|\cdot\|_{\boldsymbol{\Sigma}^{-1}} := \langle \cdot, \boldsymbol{\Sigma}^{-1} \cdot \rangle$ and C is a constant that does not depend on $\boldsymbol{x}$. The resulting optimization problem is

$$\widehat{\boldsymbol{x}}_{\mathrm{MAP}} := \arg\max_{\boldsymbol{x}} \left[-\frac{1}{2}\|\mathcal{A}(\boldsymbol{x}) - \boldsymbol{y}\|^2_{\boldsymbol{\Sigma}^{-1}} + \log p(\boldsymbol{x})\right]. \tag{5.6}$$

This optimization problem can be solved by existing algorithms for relatively simple priors [19, 143, 162]. For more sophisticated priors, the plug-and-play prior (PnP) [291] and regularization by denoising (RED) [248] frameworks provide algorithmic tools for solving Equation (5.6).

Despite its clear path toward a solution, the MAP approach has some fundamental limitations:

- **Instability:** Unlike the full posterior, the MAP estimate does not possess Lipschitz continuity with respect to $\boldsymbol{y}$ [268]. As a result, small perturbations in the measurements due to the noise could lead to large variations in the solution $\widehat{\boldsymbol{x}}_{\text{MAP}}$. This is undesirable from a theoretical perspective.
- **Lack of uncertainty quantification:** Most MAP estimation algorithms only provide a deterministic solution, which cannot represent the full solution space. This problem is particularly prominent for ill-posed problems where the reconstruction comes with significant uncertainty. Even if one can heuristically obtain multiple solutions by introducing randomness or choosing different initializations and hyperparameters, these solutions may be biased and lack a principled interpretation.

These limitations motivate the need for methods that can sample from the full posterior distribution, enabling both accurate reconstructions and rigorous uncertainty quantification.

5.2 Diffusion Models

5.2.1 Basics

Diffusion models (DMs) are a class of generative models that can capture complicated high-dimensional distributions [131, 151, 266]. They have achieved remarkable success across a variety of domains, including natural image synthesis [249], protein structure generation [108], molecular design [203], and robotic trajectory modeling [59].

We adopt the continuous-time formulation of DMs based on stochastic differential equations (SDEs), as introduced by Song et al. [266]. First consider a *forward diffusion process* that gradually transforms a data distribution $\boldsymbol{x}_0 \sim p(\boldsymbol{x}_0)$ into an approximately Gaussian distribution $\boldsymbol{x}_T \sim \mathcal{N}(\boldsymbol{0}, \sigma_T^2 \boldsymbol{I})$ where σ_T is the noise level at

time T. Mathematically, the forward process can be characterized by

$$\mathrm{d}\boldsymbol{x}_t = f(\boldsymbol{x}_t, t)\mathrm{d}t + g(t)\mathrm{d}\boldsymbol{w}_t, \tag{5.7}$$

where f is the drift coefficient, g is the diffusion coefficient, and $\boldsymbol{w}$ is the standard Wiener process with time t flowing from 0 to T. A key observation is that this forward SDE has a reverse time SDE with the same marginal distributions [9]. The backward process removes noise and gradually transforms a Gaussian sample back to a clean sample, defined by the reverse-time SDE

$$\mathrm{d}\boldsymbol{x}_t = \left(f(\boldsymbol{x}_t, t) - \frac{1}{2} g^2(t) \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t) \right) \mathrm{d}t + g(t)\mathrm{d}\bar{\boldsymbol{w}}_t, \tag{5.8}$$

where $p_t(\boldsymbol{x}_t)$ is the probability density of $\boldsymbol{x}_t$ at time t and $\bar{\boldsymbol{w}}_t$ is the reverse-time Wiener process. A neural network is trained to learn the *score function* $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t)$ for $t \in [0, T]$ via denoising score matching [292]. Once trained, we can generate new samples from the learned data distribution by first sampling $\boldsymbol{x}_T \sim \mathcal{N}(\mathbf{0}, \sigma_T^2 \boldsymbol{I})$ and solving Equation (5.8) with numeric solvers.

5.2.2 Latent Diffusion Models

Latent diffusion models (LDMs) [249] offer a way to reduce the computational cost otherwise necessary to model the original high-dimensional data distribution. LDMs generate an efficient, low-dimensional latent representation $\boldsymbol{z}_0 \in \mathbb{R}^d$ of data $\boldsymbol{x}_0 \in \mathbb{R}^n$ with a pre-trained perceptual compression encoder $\mathcal{E}$ and decoder $\mathcal{D}$, which satisfy $\boldsymbol{z}_0 = \mathcal{E}(\boldsymbol{x}_0)$ and $\mathcal{D}(\boldsymbol{z}_0) \approx \boldsymbol{x}_0$. The compression models $\mathcal{E}$ and $\mathcal{D}$ can be trained as VAE variants [38, 163, 246] with KL divergence regularization or VQGAN variants [95, 120, 228] with quantization regularization. The generation process solves the reverse-time SDE (5.8) in the latent space, followed by decoding with the decoder $\mathcal{D}$.

5.3 Solving Inverse Problems with Diffusion Models

The expressive power of diffusion models for modeling complex, high-dimensional distributions makes them a promising choice for solving Bayesian inverse problems [64, 265]. Prior work has largely focused on two main strategies[1]: conditional diffusion models (CDMs) and plug-and-play diffusion priors (PnPDP).

[1] Additional formulations include Variational Bayes [101, 103, 214] and Sequential Monte Carlo (SMC) [44, 89, 283, 318].

5.3.1 Conditional Diffusion Models

Inspired by classifier-free guidance [132], conditional diffusion models (CDMs) consider the conditional reverse diffusion process

$$\mathrm{d}\boldsymbol{x}_t = \left(f(\boldsymbol{x}_t, t) - \frac{1}{2} g^2(t) \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t \mid \boldsymbol{y}) \right) \mathrm{d}t + g(t) \mathrm{d}\bar{\boldsymbol{w}}_t. \tag{5.9}$$

where the conditional score function $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y})$ can be learned by applying denoising score matching to pairs $(\boldsymbol{x}_0, \boldsymbol{y})$.

CDMs naturally extend DMs to conditional sampling, but suffer two major limitations: (1) Training a CDM requires the joint distribution of $\boldsymbol{x}_0$ and $\boldsymbol{y}$, which depends on the specific forward model $\mathcal{A}$ and noise profile of the inverse problem. As a result, a CDM is learned for a specific inverse problem and does not generalize if the forward model $\mathcal{A}$ or the noise distribution changes. One needs to retrain the model for each problem, even when the same prior distribution is considered, making this approach inflexible. (2) If $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y})$ is learned directly with a neural network, the measurements $\boldsymbol{y}$ will be an input to the network. In general, especially for nonlinear inverse problems, $\boldsymbol{y}$ may belong to a totally different space from $\boldsymbol{x}_t$. This mismatch poses practical challenges when designing neural networks to learn the conditional score, as the network must infer a complex relationship between measurements and image space.

Due to these limitations, CDMs are mainly considered for relatively simpler problems where the spatial structure of $\boldsymbol{y}$ aligns with the target image $\boldsymbol{x}_0$, such as image inpainting or super-resolution [17, 79, 253].

5.3.2 Plug-and-Play Diffusion Priors

Plug-and-play diffusion priors (PnPDP) extend the philosophy of plug-and-play priors [291] to diffusion models. Rather than training a conditional model, PnPDP relies on pre-trained unconditional diffusion models as stand-alone priors for inverse problems [34, 64, 262, 265, 306, 361]. Instead of solving Equation (5.9) directly, one popular approach is to apply Bayes' theorem and rewrite the conditional score function as

$$\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t \mid \boldsymbol{y}) = \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y} \mid \boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t). \tag{5.10}$$

Note that the second term is exactly the score function of the prior term, which can be modeled directly by the pre-trained DM. This decomposition decouples the prior from the likelihood, allowing the same diffusion model to be reused across

different inverse problems without re-training. However, the first term is difficult to evaluate as it depends on $\boldsymbol{x}_t$, an intermediate noisy variable, instead of $\boldsymbol{x}_0$. In fact, the likelihood $p_t(\boldsymbol{y} \mid \boldsymbol{x}_t)$ can be expressed as

$$p_t(\boldsymbol{y} \mid \boldsymbol{x}_t) = \int p_t(\boldsymbol{y} \mid \boldsymbol{x}_t, \boldsymbol{x}_0) p(\boldsymbol{x}_0 \mid \boldsymbol{x}_t) \mathrm{d}\boldsymbol{x}_0 = \int p(\boldsymbol{y} \mid \boldsymbol{x}_0) p_t(\boldsymbol{x}_0 \mid \boldsymbol{x}_t) \mathrm{d}\boldsymbol{x}_0, \quad (5.11)$$

which involves integration over all possible clean images $\boldsymbol{x}_0$. This integral is analogous to the model evidence term in Equation (5.10), and is similarly intractable in practice. Various approximations have been proposed to overcome the intractability, but they often lead to significant sampling errors even in simple, low-dimensional settings [44, 341].

5.4 Part Outline

In the remainder of this part, we present a series of works that push the boundary of posterior estimation in computational imaging:

- In Chapter 6, we introduce a novel PnPDP method for posterior sampling with DMs called PnP-DM. We leverage a principled Markov chain Monte Carlo (MCMC) approach, called Split Gibbs Sampler (SGS), and identify its key connection with the EDM formulation of DMs. This perspective allows us to use pre-trained DMs as image priors without re-training or heuristic approximations. We provide both theoretical guarantees and empirical results, showing that our method achieves more accurate posterior estimates and higher-quality reconstructions across a range of linear and nonlinear inverse problems.

- In Chapter 7, we generalize PnP-DM to a unified PnPDP framework for diffusion-based posterior estimation that can accommodate a broader class of inverse problems. This framework gives rise to four instantiations, each targeting a specific challenge beyond the original PnP-DM formulation: leveraging semantic information through text conditioning, scaling to high-dimensional video settings, operating with non-differentiable (black-box) forward models, and performing posterior inference in discrete domains. Despite tackling different problem settings, all four variants share a common alternating-update structure inspired by PnP-DM, and they demonstrate both theoretical convergence properties and strong empirical results across a variety of tasks.

- In Chapter 8, we introduce a comprehensive benchmark that systematically evaluates diffusion-based methods for solving inverse problems. Specifically,

we consider 14 popular methods and 5 representative inverse problems in various scientific domains. Through extensive comparisons and ablation studies, we highlight key strengths and limitations of existing techniques and identify promising directions for advancing posterior sampling in computational imaging.

Chapter 6

PNP-DM: A PRINCIPLED FRAMEWORK FOR POSTERIOR ESTIMATION USING DIFFUSION MODELS

In this chapter, we propose a new framework that leverages diffusion models (DMs) for sampling the posterior distribution of an inverse problem. DMs have recently shown outstanding capability in modeling complex image distributions, making them expressive image priors for solving Bayesian inverse problems. However, most existing DM-based methods rely on approximations in the generative process to be generic to different inverse problems, leading to inaccurate sample distributions that deviate from the target posterior defined within the Bayesian framework. To harness the generative power of DMs while avoiding such approximations, we propose a Markov chain Monte Carlo algorithm that performs posterior sampling for general inverse problems by reducing it to sampling the posterior of a Gaussian denoising problem. Crucially, we leverage a general DM formulation as a unified interface that allows for rigorously solving the denoising problem with a range of state-of-the-art DMs. We demonstrate the effectiveness of the proposed method on seven inverse problems (four linear and three nonlinear), including a real-world black hole imaging problem. Experimental results indicate that our proposed method offers more accurate reconstructions and posterior estimation compared to existing DM-based imaging inverse methods.

This chapter is based on our work [320], published in the *Proceedings of the 38th Annual Conference on Neural Information Processing Systems 2024 (NeurIPS 2024)*. The appendix for this chapter is Appendix C. The code for the work presented in this chapter is available at `https://github.com/zihuiwu/PnP-DM-public`.

6.1 Introduction

Diffusion models generate samples from a distribution by reversing a diffusion process from the target distribution to a simple (usually Gaussian) distribution [131, 266]. In particular, it estimates a clean image $\boldsymbol{x}_0$ from a Gaussian noise image $\boldsymbol{x}_T$ by successively denoising noisy images, where $\boldsymbol{x}_t \sim p_t$ is the intermediate noisy image at time $t \in [0, T]$. Reversing diffusion requires one to estimate the time-varying gradient log density (score function) $\nabla \log p_t(\boldsymbol{x}_t)$ along the diffusion process, or $\nabla \log p_t(\boldsymbol{x}_t \mid \boldsymbol{y})$ in the case of sampling the posterior $p(\boldsymbol{x} \mid \boldsymbol{y})$.

To design generic DM-based inverse problem solvers, most existing methods attempt to approximate the time-varying gradient log density $\nabla \log p_t(\boldsymbol{x}_t \mid \boldsymbol{y})$ [34, 60, 64, 65, 155, 189, 251, 260, 262, 265, 306, 324, 361]. In particular, they first apply Bayes' rule to separate the forward operator from an unconditional prior over the intermediate noisy image $\boldsymbol{x}_t$:

$$\nabla \log p_t(\boldsymbol{x}_t \mid \boldsymbol{y}) = \nabla \log p_t(\boldsymbol{y} \mid \boldsymbol{x}_t) + \nabla \log p_t(\boldsymbol{x}_t). \tag{6.1}$$

By instead aiming to evaluate the right-hand side, one can leverage the existing pre-trained DMs for the unconditional term $\nabla \log p_t(\boldsymbol{x}_t)$. However, the main challenge in this case is that $\nabla \log p_t(\boldsymbol{y} \mid \boldsymbol{x}_t)$ is intractable to compute in general, as $p_t(\boldsymbol{y} \mid \boldsymbol{x}_t)$ involves an integral over all possible $\boldsymbol{x}_0$'s that could give rise to $\boldsymbol{x}_t$ [64]. Various methods have been proposed to circumvent the intractability and can mostly be categorized into two groups. One group of methods explicitly approximate $\nabla \log p_t(\boldsymbol{y} \mid \boldsymbol{x}_t)$ by making simplifying assumptions [34, 64, 262, 265]. However, even for arguably the finest approximation to date proposed in the recent work [34], it is exact only when the prior distribution $p(\boldsymbol{x})$ is Gaussian. For general prior distributions beyond Gaussian, these methods do not sample the true posterior $p(\boldsymbol{x} \mid \boldsymbol{y})$. The other group of methods do not make explicit approximations but instead substitute $\nabla \log p_t(\boldsymbol{y} \mid \boldsymbol{x}_t)$ with empirically designed updates where $\boldsymbol{y}$ is treated as a guidance signal [60, 65, 155, 189, 251, 260, 306, 324, 361]. Although these methods may have strong empirical performance, they have deviated from the Bayesian formulation and no longer aim to sample the target posterior. In summary, these existing DM-based inverse methods should be best viewed as *guidance methods*, where the generative process is guided towards the regions where the measurement $\boldsymbol{y}$ is more likely to be observed, not as posterior sampling methods [34]. We also note that some recent work considered combining DMs with Sequential Monte Carlo to ensure asymptotic consistency in posterior sampling [44, 89], but the investigation has been limited to linear imaging inverse problems.

Chapter Summary In this chapter, we pursue a different path towards posterior sampling with DM priors by proposing a new Markov chain Monte Carlo (MCMC) algorithm, which we call *Plug-and-Play Diffusion Models* (PnP-DM). It incorporates DMs in a principled way and circumvents the approximation required when taking the approach in Equation (6.1). The proposed algorithm is based on the Split Gibbs Sampler [296] that alternates between two sampling steps that separately involve the likelihood and prior. While the likelihood step can be tackled with traditional sampling techniques, the prior step involves a Bayesian denoising problem that

requires careful design. Importantly, we identify a connection between the Bayesian denoising problem and the unconditional image generation problem under a general formulation of DMs presented in [151] (which is referred to as the EDM formulation hereafter). This connection allows us to perform rigorous posterior sampling for denoising using DMs without approximating the generative process and enables the use of a wide range of pre-trained DMs through the unified EDM formulation. We present an analysis of the non-asymptotic behavior of PnP-DM by establishing a stationarity guarantee in terms of the average Fisher divergence. We further demonstrate the strong empirical performance of PnP-DM by investigating four linear and three nonlinear noisy inverse problems, including a black hole interferometric imaging problem involving real data that is both nonlinear and severely ill-posed. Overall, PnP-DM outperforms existing baseline methods, achieving higher accuracy in posterior estimation.

6.2 Preliminaries

Split Gibbs Sampler (SGS) is an MCMC approach developed for Bayesian inference [296]. It is also related to the *Proximal Sampler* [58, 97, 176, 336] and serves as the backbone for the *Generative Plug-and-Play (GPnP)* [31] and *Diffusion Plug-and-Play (DPnP)* [326] frameworks in computational imaging. The goal of SGS is to sample the posterior distribution

$$p(\boldsymbol{x} \mid \boldsymbol{y}) \propto p(\boldsymbol{y} \mid \boldsymbol{x})p(\boldsymbol{x}) = \exp(-f(\boldsymbol{x};\boldsymbol{y}) - g(\boldsymbol{x})) \tag{6.2}$$

where $f(\boldsymbol{x};\boldsymbol{y}) := -\log p(\boldsymbol{y} \mid \boldsymbol{x})$ and $g(\boldsymbol{x}) := -\log p(\boldsymbol{x})$ are the potential functions of the likelihood and prior distribution, respectively. The dual dependence of Equation (6.2) on both the likelihood and prior makes it nontrivial to directly sample from it in general. Instead, SGS leverages the composite structure of the posterior distribution by adopting a variable-splitting strategy and considers sampling an alternative distribution

$$\pi(\boldsymbol{x}, \boldsymbol{z}) \propto \exp\left(-f(\boldsymbol{z};\boldsymbol{y}) - g(\boldsymbol{x}) - \frac{1}{2\eta^2}\|\boldsymbol{x} - \boldsymbol{z}\|_2^2\right) \tag{6.3}$$

where $\boldsymbol{z} \in \mathbb{R}^n$ is an augmented variable and $\eta > 0$ is a hyperparameter that controls the strength of the coupling between $\boldsymbol{x}$ and $\boldsymbol{z}$. We denote the $\boldsymbol{x}$- and $\boldsymbol{z}$-marginal distributions of Equation (6.3) as $\pi^X(\boldsymbol{x}) := \int \pi(\boldsymbol{x}, \boldsymbol{z})\mathrm{d}\boldsymbol{z}$ and $\pi^Z(\boldsymbol{x}) := \int \pi(\boldsymbol{x}, \boldsymbol{z})\mathrm{d}\boldsymbol{x}$, respectively. As $\eta \to 0$, π^X converges to the target posterior $p(\boldsymbol{x} \mid \boldsymbol{y})$ in terms of total variation distance [296], so one can obtain approximate samples from the target posterior by sampling Equation (6.3) instead.

SGS samples Equation (6.3) via Gibbs sampling. Specifically, SGS starts from an initialization $\boldsymbol{x}^{(0)}$ and, for iteration $k = 0, \cdots, K-1$, alternates between

1. **Likelihood step:** sample $\boldsymbol{z}^{(k)} \sim \pi^{Z|X=\boldsymbol{x}^{(k)}}(\boldsymbol{z}) \propto \exp\left(-f(\boldsymbol{z};\boldsymbol{y}) - \frac{1}{2\eta^2}\|\boldsymbol{x}^{(k)} - \boldsymbol{z}\|_2^2\right)$

2. **Prior step:** sample $\boldsymbol{x}^{(k+1)} \sim \pi^{X|Z=\boldsymbol{z}^{(k)}}(\boldsymbol{x}) \propto \exp\left(-g(\boldsymbol{x}) - \frac{1}{2\eta^2}\|\boldsymbol{x} - \boldsymbol{z}^{(k)}\|_2^2\right)$.

Note that the two conditional distributions separately involve $f(\cdot;\boldsymbol{y})$ and $g(\cdot)$. The likelihood and prior are decoupled so that these two steps can be designed in a modular way. A similar variable-splitting strategy is also adopted in optimization methods such as the Half-Quadratic Splitting (HQS) method [114] and the Alternating Direction Method of Multipliers (ADMM) [33, 109]. In fact, SGS can be viewed as a sampling analogue of HQS. SGS is a principled approach to posterior sampling if the two sampling steps are rigorously implemented.

Existing Works Related to SGS Several works have designed algorithms for solving imaging inverse problems based on SGS [31, 69, 100, 230, 326]. The key distinction among these methods lies in their approaches to the prior step. For instance, the works [31, 100, 230] applied Langevin-based updates for sampling $\pi^{X|Z=z}$ such that the prior information is encoded by either traditional regularizers or off-the-shelf image denoisers. The work [69] tackled the prior step by heuristically customizing a diffusion model (i.e., DDPM [131]) for sampling $\pi^{X|Z=z}$. A concurrent work [326] improved the implementation by devising two diffusion processes that rigorously solve the prior step. Our method differs from [326] by connecting the prior step to the EDM formulation [151]. This connection allows us to seamlessly integrate state-of-the-art DMs as expressive image priors for Bayesian inference through a unified interface, eliminating the need for additional customization for each model and leading to better empirical performance. We also note the recent work [181] that adopted the optimization-based variable-splitting formulation of HQS and utilized general DMs as image priors. We instead consider the SGS formulation from a Bayesian posterior sampling standpoint. Additionally, while SGS-based methods theoretically accommodate general inverse problems, empirical evidence on real-world nonlinear inverse problems remains scarce in the literature. In this work, we demonstrate our method on three nonlinear inverse problems, including a black hole imaging problem. For a more comprehensive review of related works, see Appendix C.5.

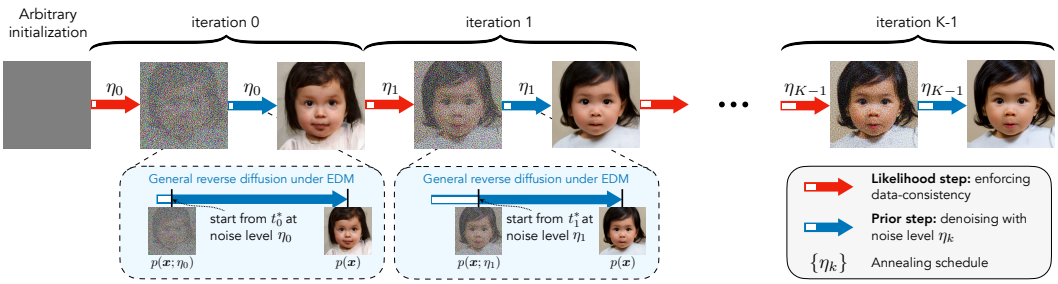


Figure 6.1: **A schematic diagram of our method.** Our method alternates between a likelihood step that enforces data consistency and a prior step that solves a denoising posterior sampling problem by leveraging the Split Gibbs Sampler [296]. An annealing schedule controls the strength of the two steps at each iteration to facilitate efficient and accurate sampling. A crucial part of our design is the prior step, where we identify a key connection to a general diffusion model framework called the EDM [151]. This connection allows us to easily incorporate a family of state-of-the-art diffusion models as priors to conduct posterior sampling in a principled way without additional training. Our method demonstrates strong performance on a variety of linear and nonlinear inverse problems.

6.3 Method

A schematic diagram for the proposed method is shown in Figure 6.1. Our method, dubbed PnP-DM, builds upon the SGS framework with rigorous implementations of the two sampling steps and an annealing schedule for the coupling parameter η. We start with our implementations of the first step for solving both linear and nonlinear inverse problems.

6.3.1 Likelihood Step: Enforcing Data Consistency

For the likelihood step at iteration k, we sample

$$z^{(k)} \sim \pi^{Z|X=\boldsymbol{x}^{(k)}}(z) \propto \exp\left(-f(z;\boldsymbol{y}) - \frac{1}{2\eta^2}\|\boldsymbol{x}^{(k)} - z\|_2^2\right). \tag{6.4}$$

Linear Forward Model and Gaussian Noise We first consider a simple yet common case where the forward model $\mathcal{A}$ is linear and the noise distribution is zero-mean Gaussian, i.e., $\mathcal{A} := \boldsymbol{A} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$. In this case, the potential function of the likelihood term is $f(\boldsymbol{x};\boldsymbol{y}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{\boldsymbol{\Sigma}}^2$ (up to an additive constant that does not depend on $\boldsymbol{x}$ and $\boldsymbol{y}$) where $\|\cdot\|_{\boldsymbol{\Sigma}}^2 := \langle\cdot, \boldsymbol{\Sigma}^{-1}\cdot\rangle$. It is then straightforward to show that

$$\pi^{Z|X=\boldsymbol{x}} = \mathcal{N}(\boldsymbol{m}(\boldsymbol{x}), \boldsymbol{\Lambda}^{-1})$$

where $\boldsymbol{\Lambda} := \boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A} + \frac{1}{\eta^2}\boldsymbol{I}$ and $\boldsymbol{m}(\boldsymbol{x}) := \boldsymbol{\Lambda}^{-1}(\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{y} + \frac{1}{\eta^2}\boldsymbol{x})$. The problem of sampling from Gaussian distributions has been systematically studied [295]. We refer readers to Appendix C.3.1 for a more detailed discussion.

General Case For general nonlinear inverse problems, the likelihood step is not sampling from a Gaussian distribution anymore. Nevertheless, since we have access to $\pi^{Z|X=\boldsymbol{x}}$ in closed form up to a multiplicative factor, we can use Monte Carlo methods based on Langevin dynamics to draw samples from it as long as the likelihood potential is differentiable. Specifically, we first set up the following Langevin SDE that admits $\pi^{Z|X=\boldsymbol{x}}$ as the stationary distribution

$$\mathrm{d}\boldsymbol{z}_t = \nabla \log \pi^{Z|X=\boldsymbol{x}}(\boldsymbol{z}_t)\mathrm{d}t + \sqrt{2}\mathrm{d}\boldsymbol{w}_t = \left[-\nabla f(\boldsymbol{z};\boldsymbol{y}) - \frac{1}{\eta^2}(\boldsymbol{z}-\boldsymbol{x})\right]\mathrm{d}t + \sqrt{2}\mathrm{d}\boldsymbol{w}_t.$$

We then initialize the SDE at $\boldsymbol{z}_0 = \boldsymbol{x}$ and run it with Euler discretization. The pseudocode is provided in Appendix C.3.1.

6.3.2 Prior Step: Denoising via the EDM Framework

For the prior step at iteration k, we sample

$$\boldsymbol{x}^{(k+1)} \sim \pi^{X|Z=\boldsymbol{z}^{(k)}}(\boldsymbol{x}) \propto \exp\left(-g(\boldsymbol{x}) - \frac{1}{2\eta^2}\|\boldsymbol{x}-\boldsymbol{z}^{(k)}\|_2^2\right). \tag{6.5}$$

A closer examination of Equation (6.5) reveals that this prior step is essentially to draw posterior samples for a Gaussian denoising problem, where the "measurement" is $\boldsymbol{z}^{(k)}$, the noise level is η, and the prior distribution is $p(\boldsymbol{x}) \propto \exp(-g(\boldsymbol{x}))$.

We tackle this denoising posterior sampling problem within SGS using DMs as image priors. In particular, we leverage the EDM framework [151], which was originally proposed to unify various formulations of DMs for unconditional image generation. To see the connection of the EDM framework to Equation (6.5), consider a family of mollified distributions $p(\boldsymbol{x};\sigma)$ given by adding i.i.d. Gaussian noise of standard deviation σ to the prior distribution $p(\boldsymbol{x})$, i.e., $\boldsymbol{x}+\sigma\boldsymbol{\epsilon} \sim p(\boldsymbol{x};\sigma)$. The core idea of the EDM framework is that a variety of state-of-the-art DMs can be unified into the following reverse SDE:

$$\mathrm{d}\boldsymbol{x}_t = \left[\frac{\dot{s}(t)}{s(t)}\boldsymbol{x}_t - 2s(t)^2\dot{\sigma}(t)\sigma(t)\nabla \log p\left(\frac{\boldsymbol{x}_t}{s(t)};\sigma(t)\right)\right]\mathrm{d}t + s(t)\sqrt{2\dot{\sigma}(t)\sigma(t)}\mathrm{d}\bar{\boldsymbol{w}}_t \tag{6.6}$$

where $\bar{\boldsymbol{w}}_t$ is an n-dimensional Wiener process running backward in time, $\sigma(t) > 0$ is a pre-defined noise level schedule with $\sigma(0) = 0$, $s(t)$ is a pre-defined scaling

schedule, and $\dot{\sigma}(t)$, $\dot{s}(t)$ are their time derivatives. As shown in [151], the defining property of Equation (6.6) is that $\boldsymbol{x}_t/s(t) \sim p(\boldsymbol{x}; \sigma(t))$ for any time t. Therefore, solving this SDE backward in time allows us to travel from any noise level $\sigma(t)$ to the clean image distribution at $t = 0$. This means that we can use Equation (6.6) to solve Equation (6.5) with arbitrary noise level η as long as η is within the range of $\sigma(t)$. Indeed, the distribution of $\boldsymbol{x}_0$ conditioned on $\boldsymbol{x}_t$ is

$$\begin{aligned} p(\boldsymbol{x}_0 \mid \boldsymbol{x}_t) &\propto p(\boldsymbol{x}_t \mid \boldsymbol{x}_0) p(\boldsymbol{x}_0) \\ &\propto \mathcal{N}(s(t)\boldsymbol{x}_0, s(t)^2\sigma(t)^2\boldsymbol{I}) \exp(-g(\boldsymbol{x}_0)) \\ &\propto \exp\left(-g(\boldsymbol{x}_0) - \frac{1}{2\sigma(t)^2}\|\boldsymbol{x}_0 - \boldsymbol{x}_t/s(t)\|_2^2\right). \end{aligned}$$

We highlight that the last line exactly matches Equation (6.5) when $\boldsymbol{x}_t = s(t)\boldsymbol{z}^{(k)}$ and $\sigma(t) = \eta$. Therefore, we can naturally design a practical algorithm that samples Equation (6.5) by following these three steps: (1) find t^* such that $\sigma(t^*) = \eta$, (2) initialize at $\boldsymbol{x}_{t^*} = s(t^*)\boldsymbol{z}^{(k)}$, and (3) solve Equation (6.6) backward from t^* to 0 by choosing the discretization time steps and integration scheme. Through this unified interface, any DMs, once converted to the EDM formulation, can be directly turned into a rigorous solver for Equation (6.5).

Leveraging the connection with EDM, our prior step implementation comes with a large design space that encompasses a variety of existing DMs, such as DDPM (or VP-SDE) [131], VE-SDE [266], and iDDPM [224]. In our experiments, we conduct posterior sampling with all these different models within our framework and all of them provide high-quality samples. The pseudocode of our implementation and more details on the EDM formulation for the prior step is given in Appendix C.3.2.

6.3.3 Overall Algorithm

The pseudocode of PnP-DM in complete form is presented in Algorithm 1. PnP-DM alternates between the two sampling steps with an annealing schedule $\{\eta_k\}$ for the coupling parameter. We find that the annealing schedule on η accelerates the mixing time of the Markov chain and prevents the algorithm from getting stuck in bad local minima for solving highly ill-posed inverse problems. This is a common practice in both Langevin-based [145, 156, 272] and SGS-based [31, 326] MCMC algorithms to improve the empirical performance in solving inverse problems.

Our work shares some similarities with PnP-SGS [69] but contains three main key differences. First, as demonstrated in our experiments, we investigate three nonlinear inverse problems, while nonlinear inverse problems are beyond the scope of [69].

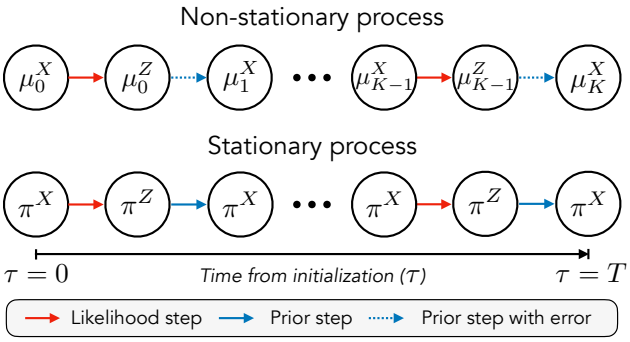


Figure 6.2: **A conceptual illustration of the non-stationary and stationary time-continuous processes as interpolations of K discretize iterations of PnP-DM.**

Our experiments show that PnP-SGS struggles with challenging nonlinear inverse problems such as Fourier phase retrieval. Second, we adopt the EDM formulation to ensure that the prior step of PnP-DM is a rigorous mapping from the image manifold with the desired noise level to the clean image manifold, aligning with the theory of SGS. In contrast, the prior step of PnP-SGS [69] is heuristic (which is also pointed out by [326]) and not rigorously designed to sample Equation (6.5). Third, unlike PnP-SGS [69] that uses a constant η, we consider an annealing schedule $\{\eta_k\}$ for the coupling parameter, which is important for highly ill-posed inverse problems.

Algorithm 1 **P**lug-and-**P**lay **D**iffusion **M**odels (PnP-DM)

Require: initialization $\boldsymbol{x}_0 \in \mathbb{R}^n$, total number of iterations $K > 0$, coupling strength schedule $\{\eta_k > 0\}_{k=0}^{K-1}$, likelihood potential $f(\,\cdot\,;\boldsymbol{y})$ with measurements $\boldsymbol{y} \in \mathbb{R}^m$, pre-trained model $D_{\boldsymbol{\theta}}(\,\cdot\,;\,\cdot\,)$ that approximates $\nabla \log p\,(\boldsymbol{x};\sigma)$ with $(D_{\boldsymbol{\theta}}(\boldsymbol{x};\sigma) - \boldsymbol{x})/\sigma^2$.

1: **for** $k = 0, \ldots, K-1$ **do**
2: $\quad \boldsymbol{z}^{(k)} \leftarrow \texttt{LikelihoodStep}(\boldsymbol{x}^{(k)}, f(\,\cdot\,;\boldsymbol{y}), \eta_k)$ ▷ Section 6.3.1
3: $\quad \boldsymbol{x}^{(k+1)} \leftarrow \texttt{PriorStep}(\boldsymbol{z}^{(k)}, D_{\boldsymbol{\theta}}(\,\cdot\,;\,\cdot\,), \eta_k)$ ▷ Section 6.3.2
4: **end for**
5: **return** $\boldsymbol{x}^{(k+1)}$

6.4 Convergence Analysis

We provide some theoretical insights on the non-asymptotic behavior of PnP-DM via a convergence analysis. We start with the following definitions. For two probability measures μ and π such that $\mu \ll \pi$, the *Kullback–Leibler (KL) divergence* and *Fisher divergence (or relative Fisher information)* of μ with respect to π are defined, respectively, as

$$\mathsf{KL}(\mu||\pi) := \int \mu \log \frac{\mu}{\pi} \quad \text{and} \quad \mathsf{FI}(\mu||\pi) := \int \mu \left\| \nabla \log \frac{\mu}{\pi} \right\|_2^2 .$$

Both divergences are equal to zero if and only if $\mu = \pi$. KL divergence is a common metric for quantifying the difference of one distribution with respect to another. Fisher divergence has been used for analyzing the stationarity of sampling algorithms [15, 273].

We analyze PnP-DM via a continuous-time perspective, leveraging the interpolation techniques introduced for Langevin Monte Carlo [15, 273, 290]. We assume that the likelihood step Equation (6.4) can be implemented exactly and the prior step Equation (6.5) involves running the reverse diffusion process Equation (6.6) with an approximated score function $\boldsymbol{s}_t \approx \nabla \log p_t := \nabla \log p(\,\cdot\,; \sigma(t))$. Let μ_0^X be the distribution of the initialization $\boldsymbol{x}^{(0)}$. Let μ_k^Z and μ_{k+1}^X be the distributions of $\boldsymbol{z}^{(k)}$ and $\boldsymbol{x}^{(k+1)}$ at the k^{th} iteration. Recall that the stationary distributions are π^X and π^Z. Our analysis is concerned with two *continuous-time* processes: (1) the non-stationary process from μ_0^X, a non-stationary initialization, to μ_K^X where Equation (6.6) is run with the approximated score function $\boldsymbol{s}_t$ and (2) the stationary process that alternates between stationary distributions π^X and π^Z. These two processes are the interpolation PnP-DM in non-stationary and stationary states and define continuous transitions over discrete iterations. A conceptual illustration of the two processes is provided in Figure 6.2 with the exact formulations in Appendix C.1. Now we present our main result:

Theorem 6.4.1. *Consider running K iterations of PnP-DM with $\eta_k \equiv \eta > 0$ and a score estimate $\boldsymbol{s}_t \approx \nabla \log p_t := \nabla \log p(\,\cdot\,; \sigma(t))$. Let $t^* > 0$ be such that $\sigma(t^*) = \eta$ and $\delta := \inf_{t \in [0, t^*]} v(t)$ where $v(t) := s(t)\sqrt{2\dot{\sigma}(t)\sigma(t)}$. Define μ_τ and π_τ as the distributions at time τ of the non-stationary and stationary process, respectively. Then, for over K iterations of PnP-DM, or equivalently over $\tau \in [0, T_K]$ with $T_K := K(t^* + 1)$, we have*

$$\underbrace{\frac{1}{T_K}\int_0^{T_K} \mathsf{FI}\left(\pi_\tau || \mu_\tau\right) \mathrm{d}\tau}_{\substack{\textit{average Fisher divergence} \\ \textit{over } K \textit{ iterations of PnP-DM}}} \leq \underbrace{\frac{4\mathsf{KL}(\pi^X || \mu_0^X)}{K(t^*+1)\min(\eta, \delta)^2}}_{\textit{convergence from initialization}} + \underbrace{\frac{4\epsilon_{score}}{(t^*+1)\delta^2}}_{\textit{score error}}, \tag{6.7}$$

where we assume that the score estimation error $\epsilon_{score} := \int_1^{t^*+1} v(\tau)^2 \mathbb{E}_{\pi_\tau} \|\boldsymbol{s}_\tau - \nabla \log p_\tau\|_2^2 \mathrm{d}\tau < \infty$.

The proof is provided in Appendix C.1. This theorem states that the average distance (measured by Fisher divergence) of the non-stationary process with respect to the stationary process over K iterations of PnP-DM goes to zero at a rate of $O(1/K)$

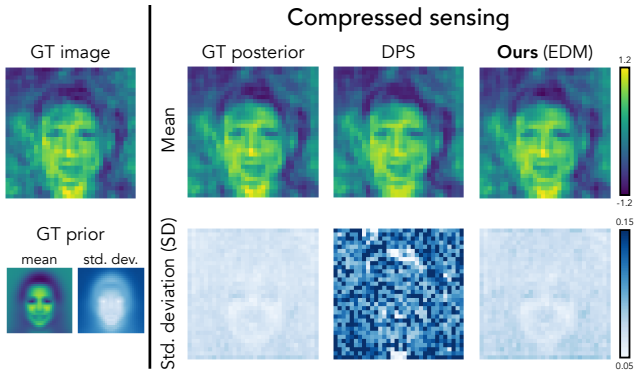


Figure 6.3: **Results on a synthetic problem with the ground truth posterior available.** PnP-DM can sample it more accurately that DPS [64].

Table 6.1: **Quantitative comparison on three noisy linear inverse problems for 100 FFHQ color test images. Bold:** best; Underline: second best.

Method	Gaussian deblur			Motion deblur			Super-resolution (4×)		
	PSNR (↑)	SSIM (↑)	LPIPS (↓)	PSNR (↑)	SSIM (↑)	LPIPS (↓)	PSNR (↑)	SSIM (↑)	LPIPS (↓)
PnP-ADMM [52]	26.88	0.7855	0.3472	26.55	0.7655	0.3600	26.61	0.7634	0.3766
DPIR [345]	28.74	0.8348	0.2677	29.97	0.8529	0.2404	28.75	0.8378	0.2577
DDRM [155]	27.05	0.7819	0.2570	–	–	–	29.47	**0.8437**	0.2322
DPS [64]	28.83	0.8212	0.2330	27.87	0.8035	0.2542	29.45	0.8379	0.2274
PnP-SGS [69]	27.46	0.8356	0.2445	28.98	0.8447	0.2190	28.30	0.8349	0.2160
DPnP [326]	29.24	0.8360	0.2098	30.21	0.8527	0.2010	29.32	0.8407	0.2127
PnP-DM (VP)	29.46	0.8215	0.2202	30.06	0.8336	0.2099	29.40	0.8238	0.2219
PnP-DM (VE)	29.65	0.8399	**0.2090**	**30.38**	0.8547	**0.1971**	29.57	0.8431	**0.2108**
PnP-DM (iDDPM)	29.60	0.8383	0.2203	30.26	0.8507	0.2103	29.53	0.8404	0.2213
PnP-DM (EDM)	**29.66**	**0.8411**	0.2170	30.35	**0.8547**	0.2062	**29.60**	0.8435	0.2191

under certain conditions up to the score approximation error. Note that our theory only requires L^2-accurate score estimate under the measure π_τ, which is a relatively weaker condition than the common L^∞-accurate score estimate assumption in prior analysis of sampling methods involving score estimates [28, 273]. This result resembles the first-order stationarity for Langevin Monte Carlo [15]. Unlike the non-asymptotic analysis in [326], we utilize the average Fisher divergence instead of the total variation distance, enabling us to obtain an explicit convergence rate. Here δ is the infimum of the diffusion coefficient along the reverse diffusion in Equation (6.6); see further discussions on the role of δ in Appendix C.1.4. Our theory shows that the accurate implementations of the two sampling steps lead to a sampler that provably converges to the stationary process that alternates between the two target stationary distributions.

6.5 Experiments

6.5.1 Validation with Ground Truth Posterior

We first demonstrate the accuracy of PnP-DM for posterior sampling on a simulated compressed sensing problem with a Gaussian prior where the posterior distribution can be expressed in a closed form. The mean and per-pixel standard deviation of the prior are visualized on the bottom left of Figure 6.3. The linear forward model $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ is a Gaussian matrix ($m = n/2$), i.e., $\boldsymbol{A}_{ij} \sim \mathcal{N}(0, 1)$. A test image is randomly generated from the prior (see top left of Figure 6.3), and the measurement is calculated according to Equation (1.1) with $\boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, 0.01^2\boldsymbol{I})$. We compare our method with the popular DM-based method DPS [64]. We draw 1,000 samples and visualize the empirical mean and per-pixel standard deviation for both algorithms. Compared with the true posterior (second column), we find that both methods accurately estimate the mean. However, the standard deviation image estimated by DPS significantly deviates from the ground truth. In contrast, our standard deviation image matches the ground truth in terms of both absolute magnitude and spatial distribution. These results highlight the accuracy of our method over DPS by taking a more principled Bayesian approach.

6.5.2 Benchmark Experiments

Dataset and Inverse Problems We test our proposed algorithm and several baseline methods on 100 images from the validation set of the FFHQ dataset [152] for five inverse problems: (1) *Gaussian deblur* with kernel size 61×61 and standard deviation 3.0, (2) *Motion deblur* with kernel size 61×61 and intensity of 0.5, (3) *Super-resolution* with 4× downsampling ratio, (4) the coded diffraction patterns (CDP) reconstruction problem (nonlinear) in [42, 221] (phase retrieval with a phase mask), and (5) the Fourier phase retrieval (nonlinear) with 4× oversampling. We add i.i.d. Gaussian noise to all the simulated measurements $\boldsymbol{y}$. In particular, i.e., $\boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, \sigma_{\boldsymbol{y}}^2\boldsymbol{I})$. For all problems except for Fourier phase retrieval, the noise standard deviation is set as $\sigma_{\boldsymbol{y}} = 0.05$. Due to the severe ill-posedness of Fourier phase retrieval, we consider a smaller noise standard deviation $\sigma_{\boldsymbol{y}} = 0.01$.

Baselines and Comparison Protocols We consider four variants of DMs as plug-in priors for our method, namely VP-SDE (VP) [131], VE-SDE (VE) [266], iDDPM [224], and EDM [151]. We compare our method with various baselines, including (1) optimization-based methods: PnP-ADMM [52], DPIR [345]; (2) conditional DMs: DDRM [155], DPS [65]; and (3) SGS-based method: PnP-SGS [69], DPnP

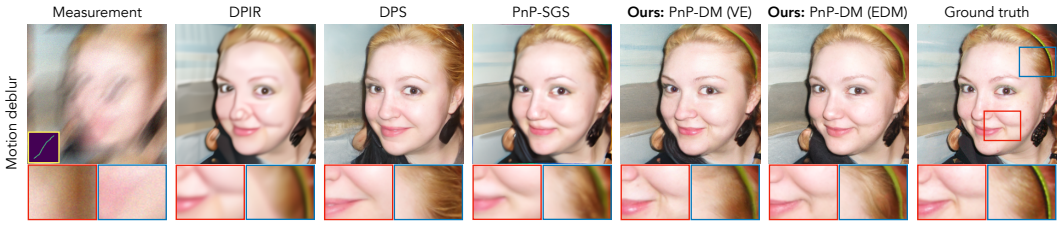


Figure 6.4: **Visual examples for the motion deblur problem ($\sigma_y = 0.05$).** We visualize one sample generated by each sampling algorithm.

[326]. For fair comparison, we use the same pre-trained score function checkpoint for all DM-based methods. Since the pre-trained score function was trained with the DDPM formulation (VP-SDE) [131], we convert it to the EDM formulation by applying the VP preconditioning [151]. We use the Peak Signal-to-Noise Ratio (PSNR), the Structural Similarity Index Measure (SSIM), and the Learned Perceptual Image Patch Similarity (LPIPS) distance for quantitative comparison. For each sampling method, we draw 20 random samples, calculate their mean, and report the metrics on the mean image. More experimental details are provided in Appendices C.2, C.3, and C.4.

Results: Linear Problems A quantitative comparison is provided in Table 6.1. PnP-DM generally outperforms the baseline methods, and the VE and EDM variants consistently outperform the other two variants on these linear problems. Figure 6.4 contains visual examples for the motion deblur problem (see Appendix C.6.2 for the other two linear problems). PnP-DM provides high-quality reconstructions that are both sharp and consistent with the ground truth image. We also provide an uncertainty quantification analysis based on pixel-wise statistics in Figure 6.5. In the left three columns, we visualize the absolute error ($|\widehat{\boldsymbol{x}} - \boldsymbol{x}_0|$), standard deviation (std), and absolute z-score ($|\widehat{\boldsymbol{x}} - \boldsymbol{x}_0|/\text{std}$). In the third column, red pixels highlight locations where the ground truth pixel values are outliers of the 3-sigma credible interval (CI) under the estimated posterior uncertainty. The fourth column contains scatter plots of $|\widehat{\boldsymbol{x}} - \boldsymbol{x}_0|$ versus std for each pixel of the reconstructions, where red boxes show the percentages of outliers (outside of 3-sigma CI) and gray boxes indicate the percentages within the 3-sigma CI. Similar to the synthetic prior experiment, DPS tends to have larger standard deviation estimations, as shown by the less concentrated distribution of gray points around the origin. Compared with baselines, especially PnP-SGS, our approach captures a higher percentage (97.46%) of ground truth pixels than the baselines (96.20% and 88.77%). If the true posterior were truly Gaussian,

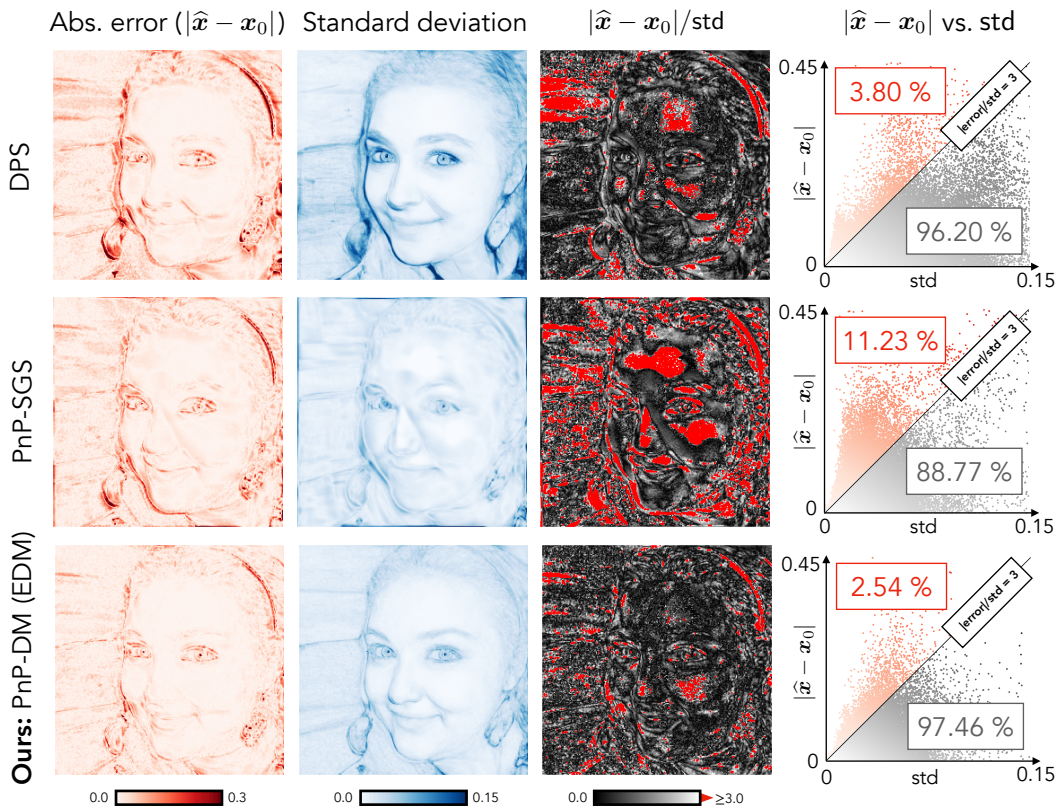


Figure 6.5: **Comparison of uncertainty quantification (UQ) for the motion deblur.** Left 3 columns: absolute error ($|\widehat{x} - x_0|$), standard deviation (std), and absolute z-score ($|\widehat{x} - x_0|$/std) with the outlier pixels in red. Right column: scatter plot of $|\widehat{x} - x_0|$ versus std. Note that PnP-DM leads to a better UQ performance than the baselines by having the lowest percentage of outliers while avoiding having overestimated per-pixel standard deviations.

99% of the ground-truth pixels should lie within the 3-sigma CI; however, as the posterior is not Gaussian with a DM-based prior, we do not necessarily expect to reach 99% coverage.

Results: Nonlinear Problems We provide a quantitative comparison in Table 6.2. For the CDP reconstruction problem, PnP-DM performs on par with DPS but outperforms other SGS-based methods. We then consider the Fourier phase retrieval (FPR) problem, which is known to be a challenging nonlinear inverse problem. One challenge lies in its invariance to 180° rotation, so the posterior distribution has two modes, one with upright images and another with 180°-rotated images, that equally fit the measurement. To increase the chance of getting properly-oriented reconstructions, we run each algorithm with four different random initializations and report the metrics for the best run, following the practice in [65]. We find that PnP-

Table 6.2: **Quantitative evaluation on two noisy nonlinear inverse problems for 100 FFHQ grayscale test images. Bold:** best; Underline: second best.

Method	Coded diffraction patterns			Fourier phase retrieval		
	PSNR (↑)	SSIM (↑)	LPIPS (↓)	PSNR (↑)	SSIM (↑)	LPIPS (↓)
HIO [104]	–	–	–	20.66	0.4308	0.6469
DPS [64]	**33.43**	0.9049	**0.1374**	23.60	0.6804	0.3126
PnP-SGS [69]	32.19	0.8889	0.2010	15.36	0.3659	0.5730
DPnP [326]	32.19	0.8853	0.2000	29.28	0.8397	0.2180
PnP-DM (VP)	32.91	0.8846	0.1906	30.36	0.8553	0.2115
PnP-DM (VE)	33.13	0.8971	0.1663	29.88	0.8464	0.2186
PnP-DM (iDDPM)	33.35	**0.9083**	0.1471	30.61	0.8718	**0.1975**
PnP-DM (EDM)	33.25	0.9050	0.1386	**31.14**	**0.8731**	0.2024

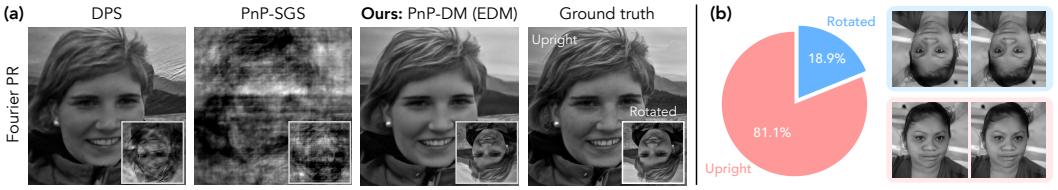


Figure 6.6: **Results of the Fourier phase retrieval problem.** (a) PnP-DM provides both upright and rotated reconstructions (two modes given by the invariance of the forward model to 180° rotation) with high fidelity, while the baseline methods cannot. (b) We visualize the percentages of upright and rotated reconstructions out of 90 runs for a test image with two samples for each orientation.

DM significantly outperforms the baselines on this highly ill-posed inverse problem. As shown in Figure 6.6 (a), our method can provide high-quality reconstructions for both orientations, while the baseline methods fail to capture at least one of the two modes. We further run our method for a test image with 100 different random initializations and collect reconstructions in both orientations that are above 28 dB in PSNR (90 out of 100 runs). The percentage of upright and rotated reconstructions is visualized by the pie chart in Figure 6.6 (b). With a prior on upright face images, our method generates mostly samples with the upright orientation. Nevertheless, it can also find the other mode that has an equal likelihood, demonstrating its ability to capture multi-modal posterior distributions.

6.5.3 Experiments on Black Hole Imaging

Problem Setup We validate PnP-DM on a real-world nonlinear imaging inverse problem: black hole imaging (BHI) (see Appendix C.2 for more details). A visual illustration of BHI is provided in Figure 6.7 (a). This BHI inverse problem is severely ill-posed. Even with an Earth-sized telescope, only a small fraction of the

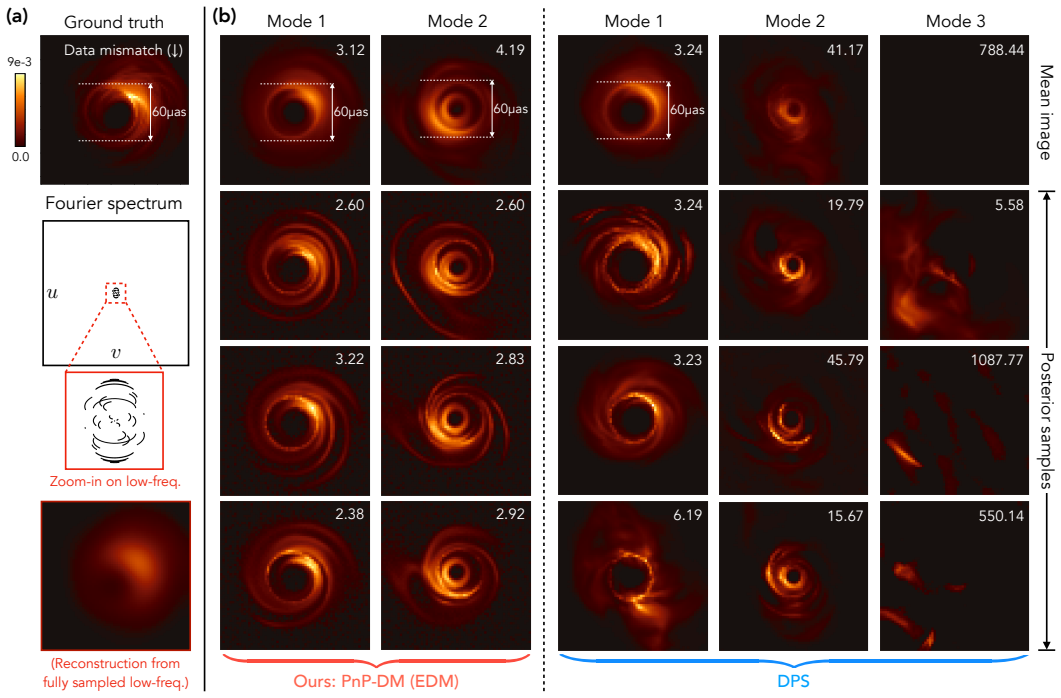


Figure 6.7: **Results on the black hole imaging problem with simulated data.** Due to severe noise corruption and measurement sparsity, this problem is non-convex and highly ill-posed, leading to a bi-modal posterior distribution as previously found in [269]. Here we compare our method, PnP-DM, with the DPS baseline [64]. A metric quantifying the mismatch with the observed measurements is labeled for each sample, which should be around 2 for an ideal measurement fit. Samples generated by PnP-DM exhibit two distinct modes with sharp details and a consistent ring structure, while samples given by DPS display inconsistent ring sizes and sometimes fail to capture the black hole structure entirely, with samples having poor measurement fit.

Fourier frequencies of the target black hole can be measured (region within the red box); in reality, this region is further subsampled with a highly sparse pattern (black lines). Additionally, the atmospheric noise causes nonlinearity of this BHI problem that sometimes results in a multi-modal posterior distribution of the reconstructed image [269]. Here we demonstrate the effectiveness of PnP-DM in capturing a multi-modal posterior distribution. For brevity, we restrict our choice of diffusion models in PnP-DM to EDM and use DPS as the baseline. The pre-trained diffusion model is trained on images from the GRMHD simulation [72].

Results on Simulated Data We first use the simulated data from [269] where the measurements are generated assuming that the ground-truth black hole image was at the location of the Sagittarius A* black hole. Figure 6.7 (b) visually compares the

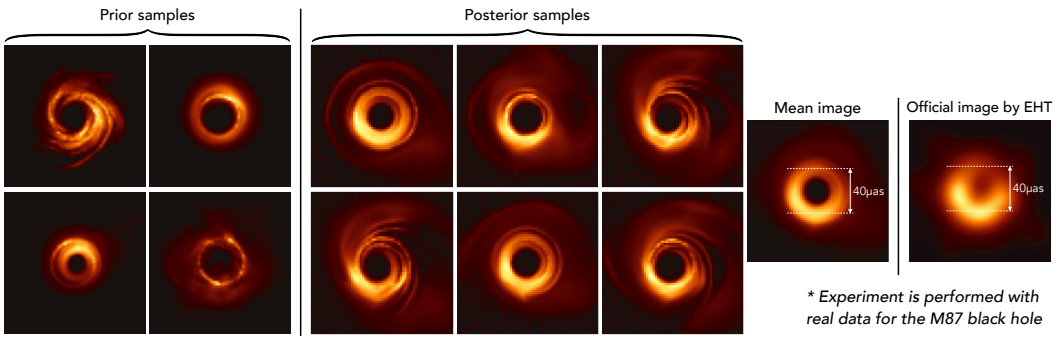


Figure 6.8: **Results on the black hole imaging problem with the real data for the M87 black hole from April 6$^{\text{th}}$, 2017 [71].** The posterior samples from PnP-DM contain fine-grained features that align with the prior distribution; see left for a few samples generated by the pre-trained diffusion model from the prior. Besides having high visual quality, our posterior samples accurately capture key features of the official reconstruction by EHT as well, such as the bright spot location and ring diameter.

results obtained by PnP-DM and DPS. We use the t-SNE method [210] to cluster the generated samples (100 for each method) and identify two modes in the samples generated by PnP-DM and three modes in those generated by DPS. We visualize the mean and three samples for each image mode. A metric for quantifying the degree of data mismatch is labeled on the top right corner of each image. As illustrated by both the mean and sample images, PnP-DM successfully captures the two modes previously identified for this dataset [269]. Note that PnP-DM generates high-fidelity samples from both modes with sharp details of the flux ring, and its samples from "Mode 1" align well with the ground truth image. In contrast, two out of the three modes sampled by DPS fail to exhibit a meaningful black hole structure and do not correspond with the observed measurements, as indicated by the significantly larger data mismatch values.

Results on Real Data We finally apply PnP-DM to the real M87 black hole data from April 6$^{\text{th}}$, 2017 [71], where the results are shown in Figure 6.8. By leveraging an expressive DM-based image prior, PnP-DM generates high-quality samples that are both visually plausible and consistent with the ring diameters observed in the official EHT reconstruction. These results highlight the robustness and effectiveness of our method in tackling a highly ill-posed real-world inverse problem.

Table 6.3: **Quantitative results for the DSA deconvolution problem in radio astronomy.** We compare PnP-DM with the well-known baseline CLEAN on detecting the galaxies and estimating their shapes and fluxes. PnP-DM outperform CLEANs in terms of all the metrics.

	Detection			Shape & flux estimation		
	Precision ↑	Recall ↑	F_1 score ↑	Semi-major axis MSE ↓	Semi-minor axis MSE ↓	Flux MSE ↓
CLEAN	0.8675	0.9791	0.9199	26.84	21.94	1.48×10^5
PnP-DM	**0.8866**	**0.9991**	**0.9395**	**4.81**	**3.20**	**897**

6.5.4 Experiments on Deconvolution with the Deep Synoptic Array (DSA)

Problem Setup We finally show the effectiveness of PnP-DM on solving a real-world deconvolution problem that arises in radio astronomy (see Appendix C.2 for more details). We adopt the same experimental setup as [73] based on the Deep Synoptic Array (DSA) [122]. We first generate a dataset of 800 true sky images of size 512×512 with a spatial resolution of 0.25 arcsecond, used for training the diffusion model in PnP-DM, and a separate dataset from the same distribution for testing. The true sky images contain 2D Gaussian ellipsoids with random orientations on the sky. The brightness, ellipticity, and angular size of each galaxy are chosen to match the empirically measured distributions for radio galaxies [284]. We simulate dirty sky measurements by convolving the true sky images with the full-band point spread function (PSF) of DSA with a bandwidth of 1,300 MHz. The main difference from the deblurring problems in Section 6.5.2 is that the additive Gaussian noise is applied before the blurring operation for this problem, which models the random fluctuations in the background of the true sky before reaching the telescopes. The standard deviation of the noise is chosen to have a signal-to-noise ratio of around 6, which is the expected level for DSA. Since some galaxies in the true sky could have intensities on par with the level of the background noise, there are uncertainties about their existence. Posterior sampling methods are desirable for quantifying the likelihoods of their existence and discovering new galaxies.

Results We compare PnP-DM with CLEAN [134], a widely used baseline in radio astronomy, which suppresses PSF-shaped artifacts in the dirty image by iteratively subtracting the PSF near the bright galaxies. We first provide a visual comparison in Figure 6.9. The top panel displays the mean of 20 posterior samples generated by PnP-DM. Since the PSF in DSA lacks strong side lobes, CLEAN is less effective in this setting where the dirty image does not contain any obvious PSF-induced artifacts. In contrast, PnP-DM faithfully reconstructs both the positions and shapes

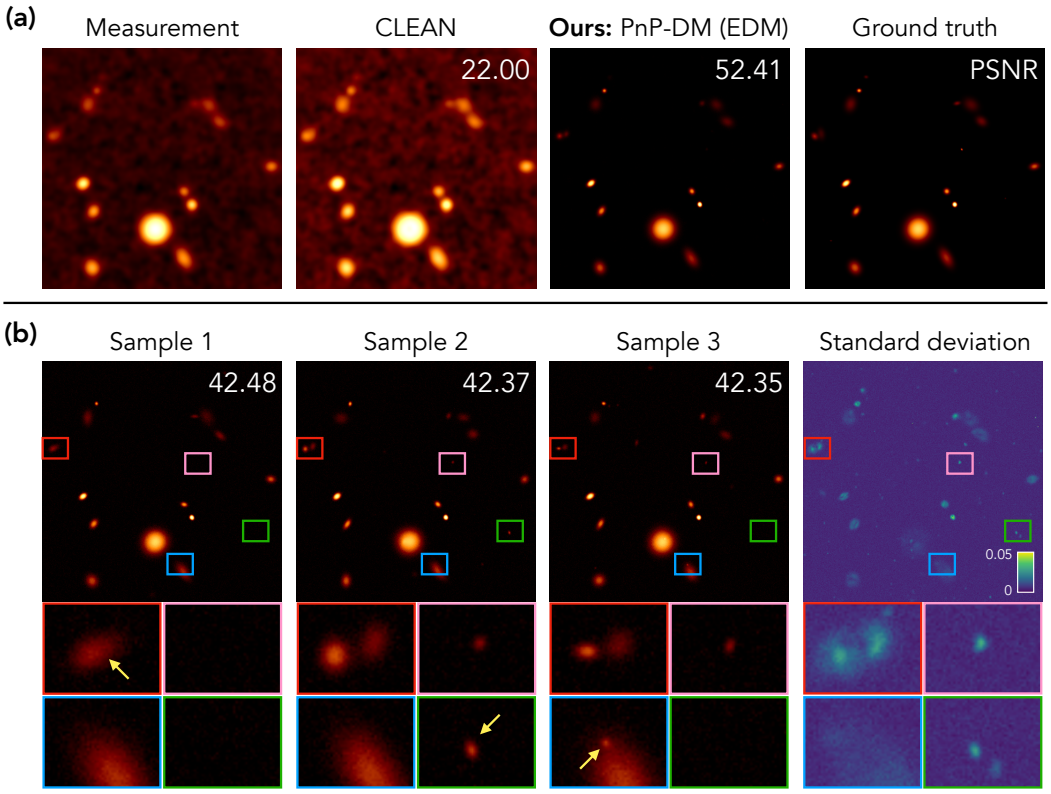


Figure 6.9: **Visual examples of the DSA deconvolution problem in radio astronomy.** All images undergo a nonlinear transformation to visualize the weaker galaxies. (a): The mean image of PnP-DM (based on 20 samples) significantly outperforms the reconstruction of the classic CLEAN algorithm [134]. (b): We visualize three posterior samples and the per-pixel standard deviation map computed from all 20 samples. Zoom-in regions highlight areas with notable sample variability.

of the galaxies, achieving a peak signal-to-noise ratio (PSNR) of 52.41dB. The bottom panel of Figure 6.9 illustrates three representative posterior samples and the per-pixel standard deviation map computed from all 20 samples. Zoom-in regions highlight areas with notable structural differences. As indicated by the yellow arrows, certain galaxies only appear in some samples but not others, illustrating the ability of PnP-DM to recover diverse structures in its reconstructions. This diversity enables us to understand and quantify the uncertainty of the reconstruction. We further assess the quality of the reconstructed astronomical images in a task-relevant manner using the `extract` function from the SEP package[1]. Specifically, we quantify their accuracies for detecting galaxies and estimating their shapes and fluxes and provide the quantitative results in Table 6.3. A detected galaxy is considered

[1] `https://github.com/sep-developers/sep` (Lesser GNU Public License)

Figure 6.10: **Comparison of galaxy property estimation accuracy between CLEAN (blue) and PnP-DM (orange).** Each scatter plot shows predicted versus true values for semi-major axis θ_A (left), semi-minor axis θ_B (middle), and flux (right), across all detected sources. The dashed line indicates perfect prediction. PnP-DM produces estimates that lie closer to the diagonal, indicating more accurate recovery of galaxy shapes and fluxes. Notably, PnP-DM avoids the strong overestimation biases seen in CLEAN.

a true positive if its center lies within 10 pixels of the nearest galaxy in the ground truth image; otherwise, it is counted as a false positive. Conversely, a false negative is recorded when no detection falls within the 10-pixel radius of a ground truth galaxy. Based on these criteria, we compute standard detection metrics: Precision, Recall, and F_1 score. Additionally, we report the mean squared errors (MSE) of the semi-major axis (θ_A), semi-minor axis (θ_B), and flux based on the metrics provided by the source detection algorithm. PnP-DM significantly outperforms CLEAN in terms of all metrics. The improvements are further shown by the scatter plots in Figure 6.10, where each point represents a galaxy detection. The estimates given by PnP-DM align with the ground truth much better than those given by CLEAN.

6.6 Conclusion

In this chapter, we introduced PnP-DM, a posterior sampling method for solving imaging inverse problems. The backbone of our method is a split Gibbs sampler that iteratively alternates between two steps that separately involve the likelihood and prior. Crucially, we established a link between the prior step and a general DM framework known as the EDM formulation. By leveraging this connection, we seamlessly integrated a diverse range of state-of-the-art DMs as priors through a unified interface. Experimental results demonstrate that our method outperforms existing DM-based methods across both linear and nonlinear inverse problems, including a nonlinear and severely ill-posed black hole interferometric imaging problem.

Limitations PnP-DM can be further improved in the following two aspects. First, PnP-DM currently requires evaluating the likelihood and prior steps for the entire image at a time. This potentially poses computational challenges in solving large-scale inverse problems (e.g., 3D imaging) or those with expensive likelihood evaluation (e.g., PDE inverse problems). Second, the current theoretical analysis does not consider the approximation error introduced in the likelihood step for general nonlinear inverse problems when running Langevin MCMC for finite iterations. Explicit incorporation of this error would offer further insights into the empirical performance of PnP-DM.

Chapter 7

BEYOND PNP-DM: TOWARDS A UNIFIED FRAMEWORK FOR GENERAL POSTERIOR ESTIMATION

In Chapter 6, we introduced PnP-DM as a general method for solving inverse problems using diffusion models. While it has proven effective across a range of linear and nonlinear imaging problems, several important challenges remain unaddressed. In this chapter, we take the core idea of PnP-DM further and work toward a unified PnPDP framework capable of handling more general classes of inverse problems.

We begin in Section 7.1 by outlining four challenges that are beyond the reach of the original PnP-DM in Chapter 6. We then formulate a unified framework in Section 7.2 for addressing these challenges. The remainder of the chapter presents four instantiations (Section 7.3, Section 7.4, Section 7.5, Section 7.6), each designed to address one of these challenges. This chapter is based on our works [22, 63, 297, 352], which collectively expand the scope and applicability of diffusion-based posterior estimation.

7.1 Limitations of PnP-DM

Recall that our goal is to sample from the posterior distribution $p(\boldsymbol{x}_0 \mid \boldsymbol{y})$ given measurements $\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}_0) + \boldsymbol{n}$. PnP-DM turns a pre-trained diffusion model (DM) that characterizes the prior $p(\boldsymbol{x}_0)$ into a principled sampler for the posterior. However, a few important issues remain unaddressed by the original formulation.

First, there may be additional sources of information about $\boldsymbol{x}_0$ that we would like to incorporate, such as a text description of what $\boldsymbol{x}_0$ looks like. Text could provide rich semantic information that significantly constrains the set of possible solutions, but it often requires working with more advanced latent DMs, which were not considered in Chapter 6. Can we develop a latent analogue of PnP-DM that can support state-of-the-art text-conditioned DMs?

Second, many imaging inverse problems are defined for videos, which introduce higher dimensionality and, more importantly, an extra time dimension compared to static images. How can we efficiently obtain expressive diffusion priors for videos? And how can we effectively incorporate them into the PnP-DM framework

to reconstruct videos that are consistent in both spatial and temporal dimensions?

Third, in many applications (especially those governed by partial differential equations (PDEs)), the forward model $\mathcal{A}$ may only permit forward evaluation but not differentiation. The likelihood step in Chapter 6 assumes access to the gradient of $\mathcal{A}$, making it incompatible with such black-box settings. Can we overcome this limitation and extend PnP-DM to handle non-differentiable forward models?

Fourth, many inverse problems have solution spaces that are inherently discrete, whereas PnP-DM is designed for continuous domains $\mathbb{R}^n$. Modeling data distributions in discrete spaces requires discrete DMs, which differ substantially from their continuous counterparts. Can we design an analogous version of PnP-DM for discrete spaces that also offers strong theoretical guarantees and empirical performance?

We answer all of these questions affirmatively by introducing four methods in the remainder of this chapter. To highlight their shared structure and enable future generalizations, we first present a unifying alternating-update framework that unifies all four approaches.

7.2 A Unified Diffusion-Based Posterior Estimation Framework

Algorithm 2 A Unified Diffusion-Based Posterior Estimation Framework

Require: initialization $\boldsymbol{u}^{(0)}$, total number of iterations $K > 0$, noise schedule $\{\eta_k > 0\}_{k=0}^{K-1}$, measurements $\boldsymbol{y} \in \mathbb{R}^m$, pre-trained diffusion model $\boldsymbol{s}_{\boldsymbol{\theta}}$.

1: **for** $k = 0, ..., K-1$ **do**
2: $\quad \boldsymbol{v}^{(k)} \leftarrow \texttt{LikelihoodStep}(\boldsymbol{u}^{(k)}, \boldsymbol{y}, \eta_k)$
3: $\quad \boldsymbol{u}^{(k+1)} \leftarrow \texttt{PriorStep}(\boldsymbol{v}^{(k)}, \boldsymbol{s}_{\boldsymbol{\theta}}, \eta_k)$
4: **end for**
5: **return** $\boldsymbol{u}^{(K)}$

We present the unified framework in Algorithm 2. In a nutshell, it alternates between two updates—the likelihood step and the prior step—until a final sample is obtained. The likelihood step enforces measurement consistency in a stochastic way, producing noisy output with noise level η_k. The prior step denoises the sample back from noise level η_k to the clean image manifold. The noise schedule $\{\eta_k > 0\}_{k=0}^{K-1}$ controls the progression of the reconstruction from a starting manifold (e.g., the manifold with noise level η_0) to an ending manifold with noise level $\eta_{K-1} \approx 0$.

In Section 7.3 and Section 7.4, we retain the likelihood step from PnP-DM and extend the prior step. Rather than unconditional image-space DMs for images

considered in Chapter 6, we investigate either text-conditioned latent-space DMs (Section 7.3) or latent-space video DMs (Section 7.4). Such extensions enable us to incorporate text description as a form of prior and solve video inverse problems, respectively.

In Section 7.5, we keep the prior step the same as in PnP-DM and generalize the likelihood step. Instead of using a gradient-based MCMC sampler for the likelihood step, we leverage the Ensemble Kalman technique [143] to implement the likelihood step via statistical linearization when the gradient of $\mathcal{A}$ is unavailable. This generalization enables us to solve inverse problems based on, for example, the Navier-Stokes equation.

Finally, in Section 7.6, we propose a discrete analogue of PnP-DM that requires re-designing both the likelihood and prior steps. We first construct a discrete version of Split Gibbs Sampler and then follow the same recipe in PnP-DM to obtain an implementable algorithm, where the likelihood step involves a Metropolis-Hasting sampler and the prior step involves a discrete DM.

7.3 Thrust 1: Incorporating More Information as Prior

Text descriptions provide rich semantic information for characterizing image distributions, which may be helpful as priors for solving inverse problems. For example, for a blurry photograph, we can more accurately resolve it into a sharper image if we are given a description of its content. However, the instantiation of PnP-DM in Chapter 6 assumes unconditional DMs and does not take text as an input. It remains an open question whether and how state-of-the-art text-conditioned DMs, such as StableDiffusion [249], can be effectively leveraged for solving inverse problems.

In this section, we show how to employ a text-conditioned DM in the prior updates of Algorithm 2 for incorporating text as a form of prior knowledge. Specifically, we present three instantiations of Algorithm 2 on the update rules of DCDP [181], PnP-DM (Section 6.3), and DAPS [341]. We show that they have highly similar likelihood steps and the same prior steps. Compared to existing text-guided diffusion-based techniques such as TReg [160], our approach does not require an additional CLIP model [236]. We demonstrate the effectiveness of our approach on three image restoration problems with severe ill-posedness. We find that our proposed methods can generate high-quality solutions while also being able to resolve ambiguity according to text prompts.

The appendix for this section is Appendix D.1.

7.3.1 Instantiation of Algorithm 2 with Text-Conditioned Diffusion Models

To incorporate text as a form of prior, we sample from the following posterior distribution conditioned on both measurements and text

$$p(\boldsymbol{x}_0 \mid \boldsymbol{y}, \boldsymbol{t}) \propto p(\boldsymbol{y} \mid \boldsymbol{x}_0, \boldsymbol{t})\, p(\boldsymbol{x}_0 \mid \boldsymbol{t}) = p(\boldsymbol{y} \mid \boldsymbol{x}_0)\, p(\boldsymbol{x}_0 \mid \boldsymbol{t}) \tag{7.1}$$

where the equality holds because $\boldsymbol{y}$ and $\boldsymbol{t}$ are independent conditioned on $\boldsymbol{x}_0$. The text-conditioned prior $p(\boldsymbol{x}_0 \mid \boldsymbol{t})$ can be modeled by text-conditioned DMs.

Solve Inverse Problems in Latent Space Since most text-conditioned DMs are latent-space models, we formulate inverse problems also in latent space. Assuming that the set of likely targets $\boldsymbol{x}_0$'s is in the range of a decoder $\mathcal{D}$, we have that $\exists\, \boldsymbol{z}_0$ s.t. $\boldsymbol{x}_0 = \mathcal{D}(\boldsymbol{z}_0)$ and can thus rewrite Equation (1.1) as:

$$\boldsymbol{y} = \mathcal{A}(\mathcal{D}(\boldsymbol{z}_0)) + \boldsymbol{n}. \tag{7.2}$$

It follows that the posterior $p(\boldsymbol{x}_0 \mid \boldsymbol{y})$ is the pushforward of the latent posterior $p(\boldsymbol{z}_0 \mid \boldsymbol{y})$ through $\mathcal{D}$. Then it suffices to first generate latent samples from $p(\boldsymbol{z}_0 \mid \boldsymbol{y})$ and then decode them by $\mathcal{D}$. This latent-space approach shares a similar spirit as the Deep Image Prior (DIP) approach [177], which (in our notations) chooses $\mathcal{D}$ to be a convolutional neural network and $\boldsymbol{z}_0$ to be its weights. However, there are two key distinctions. First, DIP is deterministic and only provides a single solution $\widehat{\boldsymbol{x}} = \mathcal{D}(\boldsymbol{z}_0)$, while we aim to sample from the posterior $p(\boldsymbol{x}_0 \mid \boldsymbol{y})$. Second, DIP employs an untrained network and optimizes its randomly initialized weights, while our decoder $\mathcal{D}$ is pre-trained in a VAE and fixed when we optimize its input $\boldsymbol{z}_0$. Similar to Equation (7.1), the latent posterior $p(\boldsymbol{z}_0 \mid \boldsymbol{y})$ can be written as

$$p(\boldsymbol{z}_0 \mid \boldsymbol{y}, \boldsymbol{t}) \propto p(\boldsymbol{y} \mid \boldsymbol{z}_0)\, p(\boldsymbol{z}_0 \mid \boldsymbol{t}) \tag{7.3}$$

where $p(\boldsymbol{y} \mid \boldsymbol{z}_0)$ is given by Equation (7.2) and $p(\boldsymbol{z}_0 \mid \boldsymbol{t})$ can be modelled by state-of-the-art text-conditioned DMs [240, 249].

Text-Conditioned Latent Diffusion Models (LDMs) Given a text prompt $\boldsymbol{t}$, text-conditioned LDMs sample from the text-guided prior distribution with the classifier-free guidance technique [132], which models the prior distribution as

$$p(\boldsymbol{z}_0 \mid \boldsymbol{t}) \propto p(\boldsymbol{z}_0) \cdot p(\boldsymbol{t} \mid \boldsymbol{z}_0)^w$$

where the parameter $w \geq 0$ controls the strength of the text guidance. During the reverse diffusion process, where the time-dependent score function $\nabla_{\boldsymbol{x}_0} \log p_t(\boldsymbol{x}_0 \mid \boldsymbol{t})$

is required, these models approximate it as

$$\begin{aligned}\nabla_{z_0} \log p_t(z_0 \mid \boldsymbol{t}) &= \nabla_{z_0} \log p_t(z_0) + w \cdot \nabla_{z_0} \log p_t(\boldsymbol{t} \mid z_0) \\ &\approx \boldsymbol{s}_{\boldsymbol{\theta}}(z_0, \emptyset, t) + w \cdot \boldsymbol{s}_{\boldsymbol{\theta}}(z_0, \boldsymbol{t}, t),\end{aligned}$$

where $\boldsymbol{s}_{\boldsymbol{\theta}}$ is a neural network that takes text as input and $\emptyset$ denotes an empty string. Note that setting $w = 0$ disables text conditioning, reducing the model to an unconditional diffusion model.

Overall Approach With the latent-space formulations of inverse problems and DMs in place, we can then follow the same procedure in Section 6.2 to get a latent version of the Split Gibbs Sampler in the format of Algorithm 2. Below we present three instantiations of the prior and likelihood steps based on the update rules of DCDP [181], PnP-DM (Section 6.3), and DAPS [341]. We show that they share the same prior step and only differ slightly for the likelihood step. We refer the readers to Appendix D.1.1 for a detailed derivation of each algorithm.

7.3.1.1 Prior Step via Text-Conditioned Denoising Diffusion

The prior step (Line 3 of Algorithm 2) generates a sample $\boldsymbol{u}^{(k+1)}$ from the following distribution

$$\exp\left(\log p(\boldsymbol{u} \mid \boldsymbol{t}) - \frac{1}{2\eta_k^2}\left\|\boldsymbol{u} - \boldsymbol{v}^{(k)}\right\|_2^2\right)/Z \tag{7.4}$$

where Z is a normalizing constant and $\boldsymbol{t}$ is the text prompt. As explained in Section 6.3, this sampling task is equivalent to solving a Gaussian denoising problem and can be implemented rigorously using the EDM framework given a text-conditioned DM for $p(\boldsymbol{u} \mid \boldsymbol{t})$ [151]. To sample from this distribution, we first find the timestep $t = t^*$ where $\sigma_{t^*} = \eta_k$ and set $\boldsymbol{u}_{t^*} = \boldsymbol{v}^{(k)}$. Then we run the following updated reverse diffusion probability flow ODE[1] from $t = t^*$ to $t = 0$

$$\mathrm{d}\boldsymbol{u}_t = \left[\frac{\dot{s}_t}{s_t}\boldsymbol{u}_t - s_t^2 \dot{\sigma}_t \sigma_t \boldsymbol{s}_{\boldsymbol{\theta}}\left(\frac{\boldsymbol{u}_t}{s_t}, \boldsymbol{t}, \sigma_t\right)\right]\mathrm{d}t \tag{7.5}$$

where the score function $\boldsymbol{s}_{\boldsymbol{\theta}}$ takes the text embedding $\boldsymbol{t}$ as input. The score term $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t/s_t, \boldsymbol{t}, \sigma_t)$ can be readily obtained using the built-in support for text prompts through classifier-free guidance of many state-of-the-art LDMs, such as StableDiffusion [249]. As illustrated in Figure 7.1, starting from a noisy iterate $\boldsymbol{v}^{(k)}$, the prior

[1] This is the ODE version of Equation (6.6). These two processes share the same marginal distribution at any time t.

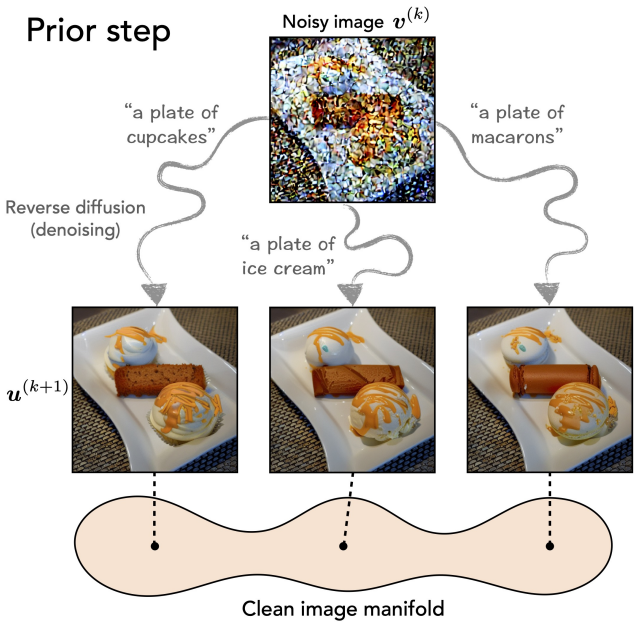


Figure 7.1: **Illustration of the prior step with different text inputs.** Starting from a noisy image $\boldsymbol{v}^{(k)}$, the prior step performs text-guided denoising through reverse diffusion to generate a cleaner sample $\boldsymbol{u}^{(k+1)}$ on the clean image manifold. The images are visualized after decoding. Different text prompts steer the denoising process toward distinct yet plausible modes of the image.

step can sample distinct yet plausible modes on the clean image manifold matching different text prompts. In scenarios where the measurement $\boldsymbol{y}$ is highly ambiguous due to the ill-posedness of the forward operator $\mathcal{A}$, the text prompt $\boldsymbol{t}$ serves as a powerful signal to guide the restoration process and enables controllability within the solution space. Our approach resembles the technique used in the SDEdit algorithm [218], where text prompts are used to guide the image generation process. Unlike prior methods such as TReg [160], our framework does not rely on a CLIP image encoder and introduces no additional hyperparameters besides those already in the LDMs (e.g., classifier-free guidance scale w).

7.3.1.2 Likelihood Step with Two Sub-Steps

For the likelihood step (Line 2 of Algorithm 2), we summarize the three instantiations in the following table.

Method	**Sub-Step 1**	**Sub-Step 2**
DCDP [181]	Find argmax of (7.6)	Add noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \eta_k^2 \boldsymbol{I})$
PnP-DM (Section 6.3)	Sample from (7.6)	—
DAPS [341]	Sample from (7.6)	Add noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \eta_k^2 \boldsymbol{I})$

Sub-Step 1: Enforcing Data Fidelity The first part of the likelihood step targets the following distribution

$$\exp\left(-\frac{1}{2\sigma_y^2}\left\|\mathcal{A}(\mathcal{D}(\boldsymbol{v}))-\boldsymbol{y}\right\|_2^2-\frac{1}{2\eta_k^2}\left\|\boldsymbol{u}^{(k)}-\boldsymbol{v}\right\|_2^2\right)/Z \tag{7.6}$$

where Z is a normalizing constant. This distribution balances fidelity to the observed measurements $\boldsymbol{y}$ with proximity to the current prior sample $\boldsymbol{u}^{(k)}$. The way this target distribution is handled varies slightly across different methods. DCDP performs MAP estimation by maximizing the log-posterior using gradient-based optimization. In contrast, PnP-DM and DAPS treat this as a sampling problem and use algorithms such as Langevin Monte Carlo or Hamiltonian Monte Carlo to draw samples from Equation (7.6). Although their formulations differ, they can be unified under our framework by identifying a correspondence between η_k and the method-specific hyperparameters used to control the relative weighting between data fidelity and prior proximity. Specifically, we have:

$$\eta_k \equiv \begin{cases} \sigma_y/\sqrt{\mu} & \text{for DCDP [181],} \\ \eta_k & \text{for PnP-DM (Chapter 6),} \\ r_t & \text{for DAPS [341] where } t \equiv k. \end{cases}$$

This correspondence reveals that all three methods are effectively targeting the same class of distributions, differing only in their choices of hyperparameters.

Sub-Step 2: Adding Noise The second part of the likelihood step involves adding Gaussian noise to the intermediate sample obtained in sub-step 1. The intuition is that Equation (7.6) is equivalent to solving the original inverse problem with a gaussian prior $\mathcal{N}(\boldsymbol{u}^{(k)}, \eta_k^2\boldsymbol{I})$, so ideally the output $\boldsymbol{v}^{(k)}$ should contain additional noise with a standard deviation of η_k upon $\boldsymbol{u}^{(k)}$. However, this may not be the case in practice, so DCDP and DAPS manually add the expected amount of noise to maintain a consistent noise level η_k across iterations, aligning the sample with the expected input distribution of the subsequent prior denoising step. On the other hand, PnP-DM strictly follows the Split Gibbs Sampling derivation and thus does not have this noise adding step. We find in the experiments that this noise-adding step is helpful empirically when using LDMs as priors.

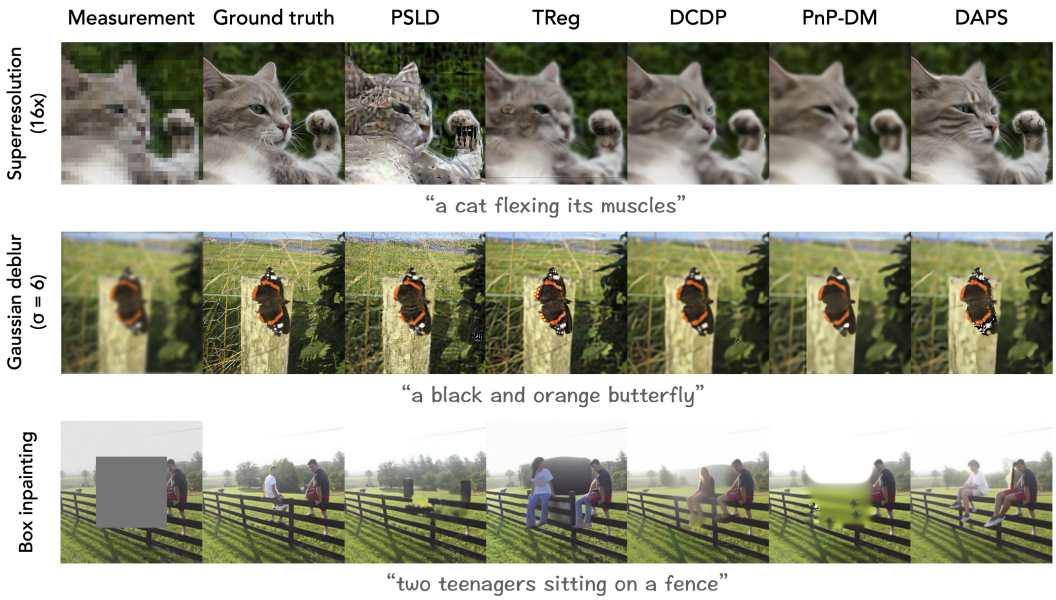


Figure 7.2: **Visual examples for the super-resolution (16×), gaussian deblur (σ = 6), and box inpainting tasks with matching prompts.** The proposed instantiations, DAPS and DCDP in particular, produce more detailed and perceptually coherent reconstructions, while the baseline methods suffer from artifacts and a decrease in quality.

Table 7.1: **Quantitative evaluation for the super-resolution (16×), gaussian deblurring (σ = 6), and box inpainting tasks with matching prompts. Bold:** best; Underline: second best. For each task, we report the mean performance and standard deviation in PSNR, SSIM, and LPIPS across 14 test images. The results show that methods based on our unified framework outperform the baseline approaches, TReg and PSLD, especially on the super-resolution and deblurring problems. The unified methods maintain strong performance without relying on additional components, highlighting their efficiency and generality.

Method	Super-resolution (16×)				Gaussian deblur (σ = 6)				Box inpainting			
	PSNR (↑)	SSIM (↑)	LPIPS (↓)	Data fit (↓)	PSNR (↑)	SSIM (↑)	LPIPS (↓)	Data fit (↓)	PSNR (↑)	SSIM (↑)	LPIPS (↓)	Data fit (↓)
TReg	20.96 (1.88)	0.48 (0.14)	0.42 (0.05)	1.99 (0.22)	23.81 (2.18)	0.60 (0.14)	**0.26** (0.06)	39.44 (5.57)	15.89 (1.37)	0.46 (0.11)	0.32 (0.04)	116.22 (32.73)
PSLD	14.74 (3.33)	0.24 (0.20)	0.53 (0.08)	3.52 (1.76)	20.42 (4.54)	0.41 (0.18)	0.44 (0.10)	79.83 (204.19) [2]	**19.72** (2.41)	**0.83** (0.03)	**0.16** (0.02)	88.51 (38.55)
DCDP	22.44 (2.44)	**0.55** (0.16)	0.45 (0.07)	**1.32** (0.10)	**24.98** (2.51)	0.62 (0.15)	0.34 (0.07)	**16.74** (1.30)	19.18 (2.34)	0.70 (0.11)	0.27 (0.07)	**56.57** (21.59)
PnP-DM	22.04 (2.80)	0.49 (0.19)	0.45 (0.08)	1.76 (0.50)	23.85 (2.43)	0.58 (0.16)	0.39 (0.06)	29.71 (4.73)	18.41 (2.83)	0.66 (0.13)	0.32 (0.06)	69.51 (19.44)
DAPS	**22.87** (2.43)	0.47 (0.15)	**0.39** (0.08)	1.46 (0.15)	24.33 (2.61)	**0.63** (0.15)	0.35 (0.09)	31.74 (2.78)	18.90 (2.57)	0.71 (0.11)	0.27 (0.07)	60.70 (22.39)

7.3.2 Experiments

7.3.2.1 Setup

Inverse Problems We test our methods on super-resolution, Gaussian deblurring, and box inpainting. For super-resolution, we consider 16× downsampling. For deblurring, we use a Gaussian blur kernel of 61×61 pixels and $\sigma = 6$. The box inpainting is done with a 256×256 pixel mask in the center of each test image. Each task is repeated with both a matching and non-matching text prompt, and measurement noise is assumed to be a zero-mean Gaussian with standard deviation $\sigma_y = 0.01$.

Datasets and Metrics For validation and testing, we use 512×512 images from the ImageNet dataset [81]. The output images are evaluated using the peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM) [310], and Learned Perceptual Image Patch Similarity (LPIPS) [349] metrics. Note that the images are resized to 256×256 using bicubic interpolation before being input to LPIPS. We also report the "data fit" for each task, defined as $\|\mathcal{A}(\widehat{\boldsymbol{x}}) - \boldsymbol{y}\|_2$ where $\widehat{\boldsymbol{x}}$ is the reconstructed image, which quantifies how well the reconstruction is consistent with the measurement $\boldsymbol{y}$.

Likelihood Step We perform the likelihood step according to the table in Section Section 7.3.1.2. In sub-step 1, sampling is done using Hamiltonian Monte Carlo (DAPS, PnP-DM) and optimization is performed using the stochastic gradient descent optimizer with momentum (DCDP). See Table D.1 for the inverse problem-specific hyperparameters used in each of these. Sub-step 2 varies by method, but does not need to change for different inverse problems.

Prior Step The prior step is performed with reverse diffusion using StableDiffusion 1.5 [249] with the `DPMSolverMultistepScheduler`. The number of steps in the scheduler is set to $K = 50$ in order to match the η-schedule.

Baselines We compare our methods with PSLD [251] and TReg [160]. The hyperparameters for these methods were selected per inverse problem through Bayesian optimization. TReg is supplied with the same text prompts as the other methods. PSLD is not given text prompts, as we found this generates significantly worse results.

7.3.2.2 Main Results

The quantitative results are shown in Table 7.1. A few qualitative samples for each method can be found in Figure 7.2. Overall, we see that the methods based on our unified framework outperform the two baseline approaches, PSLD and TReg, especially for super-resolution and Gaussian deblurring. Additionally, these methods avoid the added complexity introduced by external components such as CLIP, which is required by TReg to function.

We also observe that DAPS, PnP-DM, and DCDP work more robustly than TReg and PSLD. We see that the images recovered by PSLD tend to contain artifacts throughout the entire image. Additionally, as mentioned before, we were not able to use text prompts to guide PSLD, resulting in an inpainting result that does not contain “two teenagers" as specified in the prompt (in Figure 7.2). TReg performed well on deblurring, but struggled on box inpainting. Note that the original TReg paper used a diffusion model fine-tuned specifically for inpainting, while our models can perform inpainting with the same base StableDiffusion model used for all of the other tasks.

Within the unified methods, the reconstructions produced by PnP-DM appear blurrier and contain fewer details than those from DAPS and DCDP. We hypothesize that this is because PnP-DM is the only method that does not add noise in sub-step 2 of the likelihood step. Without this noise-adding step, we find that the output of the likelihood step, $\boldsymbol{v}^{(k)}$, in PnP-DM tends to exhibit less noise than expected. As a result, the prior step over-denoises the image—removing too many features under the assumed noise standard deviation—leading to overly smooth and blurry reconstructions. This issue was less prominent for solving inverse problems with pixel-space DMs in Chapter 6. However, it becomes more significant in this latent-space setting, where the decoder in Equation (7.2) introduces a high degree of nonlinearity.

7.3.2.3 Discussions

Through the unified framework, we can naturally control image generation via text guidance in a consistent manner. For convenience, we use DAPS as a representative to explore the following three aspects of the proposed approach.

[2]Note that PSLD failed to converge on two of the test images for the Gaussian deblur test, resulting in a high data fit term. If these outliers were removed, the data fit term would be 15.92 (1.82).

Figure 7.3: **Visual examples of DAPS on the 16× super-resolution task using different prompts.** The examples show that varying descriptions lead to semantically different yet feasible reconstructions. Despite the severe degradation of the input measurements, the restored images are of high quality and closely align with the information from the provided text prompts, while still consistent with the original measurement.

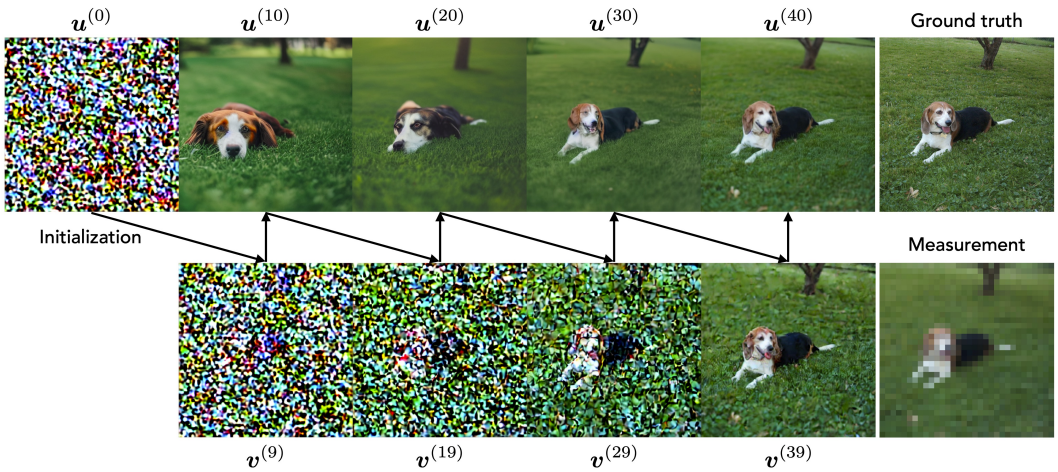


Figure 7.4: **Visualization of the image generation process of DAPS.** The top row shows outputs from the prior steps ($\boldsymbol{u}^{(k+1)}$ for $k \in \{9, 19, 29, 39\}$), which denoise and integrate text conditioning to guide the sample toward the target distribution, while the bottom row shows outputs from the likelihood steps ($\boldsymbol{v}^{(k)}$ for $k \in \{9, 19, 29, 39\}$), which enforce data consistency with added noise. Initialization starts from pure noise, and the two processes alternate to progressively generate a realistic image. The images are visualized after decoding.

Controllability with Text Guidance We first demonstrate its ability to reconstruct different image modes from significantly degraded measurements using both matching and non-matching text prompts, as shown in Figure 7.3. The matching text prompt was created as a simple description of the image, and the non-matching text prompt was created by making a slight modification to the matching text prompt. When there is enough ambiguity in the measurement, DAPS can generate detailed information that matches each of the given prompts. Sometimes this involves reinterpreting what the colors of the image represent: in the second row, the original image has a single cannoli, but when DAPS is run with a non-matching prompt, it is changed into two separate macarons.

Different Prompt Specificity We then investigate how the specificity of text prompts influences the posterior samples generated by DAPS, where the results are shown in Figure 7.5. With highly specific prompts, such as "two scoops of ice cream with a cannoli" or "a plate of macarons" (top right), DAPS effectively reconstructs the corresponding image modes in a controllable manner. As we provide increasingly generic prompts, such as "a plate of desserts" (middle) or "a plate of food" (bottom), the generated samples exhibit roughly the same level of diversity in appearance—i.e., the bottom two panels appear to be equally diverse at first

Figure 7.5: **Effect of prompt specificity on posterior samples generated by DAPS for 16× super-resolution.** We repeatedly generate samples using DAPS with prompts of varying specificity. As prompts become more generic, the samples exhibit greater semantic diversity—e.g., several yellow scoops in the "a plate of food" panel look like mashed potatoes, which are not seen in the more specific "a plate of desserts" panel. However, the overall diversity in appearance remains similar—i.e., the bottom two panels appear to be equally diverse at first glance. Modes with highly specific details, such as "two scoops of ice cream with a cannoli" or "a plate of macarons" (top right), are only recovered when given the corresponding prompts. These results underscore the importance of text conditioning in uncovering rare modes and improving mode coverage in posterior estimation.

glance—but an increased level of semantic diversity. For example, several yellow scoops in the "a plate of food" panel look like mashed potatoes, which are not seen in the more specific "a plate of desserts" panel. However, less likely modes such as "a plate of macarons" are not recovered by the samples of the more generic prompts. This highlights the importance of text conditioning in discovering less likely solutions and improving posterior mode coverage. In addition, samples from the most generic prompt ("a plate of food") tend to exhibit structurally incoherent elements, such as speckled or shredded textures in the brown and yellow objects. This is consistent with prior observations that conditional DMs often outperform unconditional ones, as conditioning reduces the complexity of the target distribution [16]. Hence, beyond improving mode coverage, text conditioning also enhances the quality of individual samples.

Visualization of Image Generation Process We also visualize the image generation process in Figure 7.4 where the bottom row contains the inputs to the prior steps $\boldsymbol{v}^{(k)}$ and the top row shows the outputs $\boldsymbol{u}^{(k+1)}$. The text prompt for this example is "A dog sitting on the grass." In the early iterations where η_k is large, the $\boldsymbol{v}$ iterates are noisier and the prior step is closer to text-conditioned image generation. As η_k anneals down to 0, the $\boldsymbol{u}$-iterates become increasingly similar to the target image.

7.4 Thrust 2: Harnessing Higher Dimension for Video Inverse Problems

Reconstructing high-quality videos from time-varying measurements is a core challenge in many scientific domains, such as black hole video reconstruction [92] and dynamic magnetic resonance imaging (MRI) [110]. In these applications, the accuracy of the recovered spatiotemporal features significantly affects downstream scientific analysis or medical interpretation. These problems are inherently difficult due to the high dimensionality of the underlying video and the severe loss of both spatial and temporal information during acquisition, necessitating a prior on both the spatial and temporal dimensions for meaningful recovery. However, the investigation in Chapter 6 was limited to image DMs. It remains an open question how to efficiently obtain DMs for videos and use them for solving video inverse problems (VIPs).

In this section, we introduce an instantiation of Algorithm 2 for solving high-dimensional scientific VIPs with **S**patio**Te**mporal video diffusion **P**riors (STeP). Existing methods for VIPs rely on extracting temporal consistency directly from measurements, which limits their effectiveness on scientific tasks with high spa-

tiotemporal uncertainty. We instead learn a spatiotemporal diffusion prior for videos. To do so efficiently, we are inspired by [303] to first train an image latent diffusion model with a 2D U-Net and then turn it into a spatiotemporal video diffusion model by adding a zero-initialized temporal module to each 2D convolution module in the U-Net. This enables us to fine-tune a spatiotemporal diffusion prior from a pre-trained IDM in a data- and time-efficient manner, using only hundreds to thousands of videos and a few hours of training on a single NVIDIA A100 GPU. STeP enforces learned priors over both spatial and temporal dimensions and does not rely on heuristics to ensure temporal consistency, making it well-suited for tasks with high spatiotemporal uncertainty. Due to its plug-and-play nature, STeP can handle general inverse problems with nonlinear forward models without the need for task-specific design or temporal heuristics to enforce temporal consistency.

We demonstrate the effectiveness of STeP on two challenging scientific video inverse problems: black hole video reconstruction (previewed in Figure 7.6) and dynamic MRI, where the underlying targets exhibit significantly different spatiotemporal characteristics. As Figure 7.6 illustrates, STeP not only achieves state-of-the-art results with improved spatial and temporal consistency but also effectively captures the multi-modal nature of highly ill-posed problems, recovering diverse plausible solutions from the posterior distribution. Notably, our approach achieves substantial improvements in both spatial and temporal consistency, significantly outperforming baselines across most evaluation metrics. For example, our method demonstrates 1.57 dB and 2.38 dB improvements in PSNR on black hole video reconstruction and dynamic MRI, respectively. More importantly, for the challenging black hole video reconstruction, while baseline methods have difficulty accurately recovering temporal dynamics, our method demonstrates spatiotemporal structure that closely aligns with ground truth video (Figure 7.9), under extremely sparse measurement conditions.

The appendix for this section is Appendix D.2. The code for the work presented in this section is available at `https://github.com/zhangbingliang2019/STeP`.

7.4.1 Background

Existing Approaches for Video Inverse Problems Existing approaches for video inverse problems have primarily focused on restoration and editing problems, such as super-resolution and JPEG artifact removal, for natural videos [53, 77, 82, 168, 169, 365]. Since it is computationally expensive to obtain a well-trained video

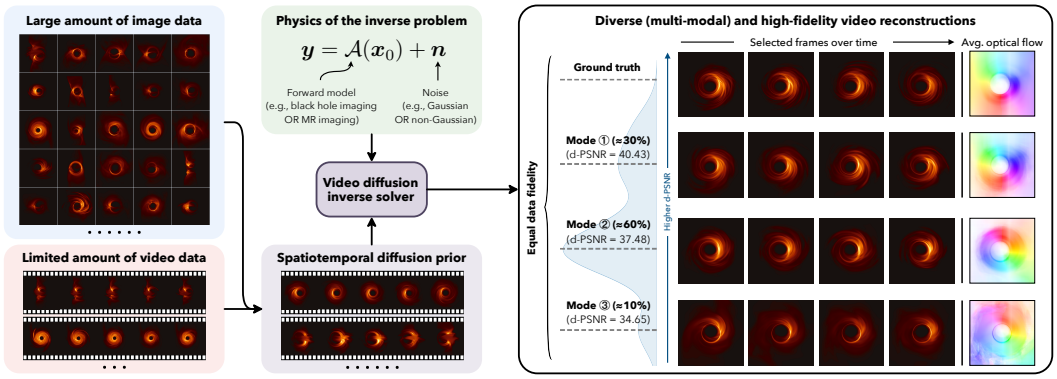


Figure 7.6: **An overview of our proposed framework with spatiotemporal diffusion priors, STeP, for scientific video inverse problems.** Left: STeP combines the physics model of the target problem with a spatiotemporal diffusion prior that directly characterizes the video distribution. We show that such a prior can be efficiently obtained by fine-tuning a pre-trained image diffusion model with limited video data. Right: STeP can generate diverse solutions to a black hole video reconstruction problem that exhibit equally good fidelity with the measurements.

diffusion model (VDM) for natural videos, existing methods instead rely on an image diffusion model (IDM) to process each video frame and propose various techniques to enforce temporal consistency. For example, the *batch-consistent sampling (BCS)* technique fixes the injected noise across all video frames [168, 169]. Another common approach, which we refer to as *noise warping*, first extracts optical flow from measurements and uses it to warp the injected noise across all video frames [53, 77, 82, 334]. It is observed empirically that the dynamic of the injected noise translates into that of the generated video. These methods work well in restoration problems where the dynamic is mostly preserved in the measurements or video editing tasks where the dynamic is given. However, these methods face challenges when dealing with more challenging problems in scientific domains because the measurements may belong to a different domain that is nontrivial to invert, or substantial spatiotemporal information may be lost in the measurement process.

Plug-and-Play Diffusion Priors for Video Inverse Problems As reviewed in Section 5.3.2, plug-and-play diffusion priors (PnPDP) constitute a family of methods that leverage diffusion models as priors for solving inverse problems. A major advantage of PnPDP is its ability to handle inverse problems in a general way without task-specific design. Video inverse problems fit into the PnPDP framework in principle, but existing PnPDP methods have mainly focused on the static image

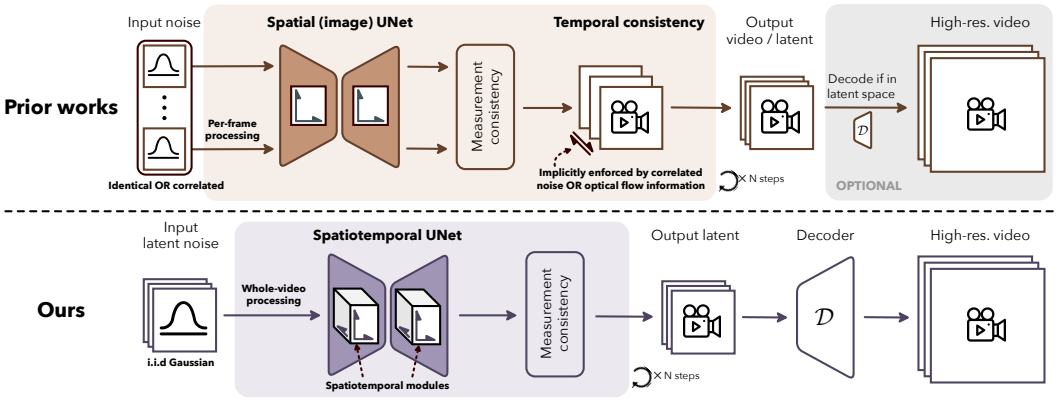


Figure 7.7: **A schematic comparison between prior works (top) and our STEP framework (bottom) for video inverse problems.** The bold texts highlight the key differences between them. While prior works use an image diffusion model and enforce temporal consistency using simple heuristics or warping noise with optical flow, we directly learn a spatiotemporal diffusion prior.

setting. The main reason is that characterizing the prior distribution $p(\boldsymbol{x}_0)$ for videos is challenging due to their high dimensionality and a potentially limited number of samples for training. Prior work [167] has explored the applicability of PnPDP to an optical scattering problem with a pixel-space video diffusion prior. Video diffusion models are commonly believed to be hard to train due to the computational overhead of 3D modules and the requirement of a large video dataset [24, 25, 133, 350, 358]. Many recent methods tend to solve video modeling by fine-tuning from a pre-trained IDM with a video dataset [24, 25, 317]. In the following sections, we propose a recipe for efficiently obtaining spatiotemporal diffusion prior and solving VIPs with an instantiation of Algorithm 2.

7.4.2 Instantiation of Algorithm 2 with Spatiotemporal Diffusion Priors

Instead of taking a per-frame processing approach using IDMs, we propose to directly learn the video distribution $p(\boldsymbol{x}_0)$ from data (Section 7.4.2.1) and draw samples from the posterior distribution $p(\boldsymbol{x}_0 \mid \boldsymbol{y}) \propto p(\boldsymbol{y} \mid \boldsymbol{x}_0)p(\boldsymbol{x}_0)$ via an instantiation of Algorithm 2 (Section 7.4.2.2). Figure 7.7 illustrates the conceptual difference between STEP and the prior approaches on video inverse problems. Designed mainly for restoration problems on natural videos, prior works rely on IDMs to incorporate spatial prior and enforce temporal consistency by injecting correlated noise in the diffusion process. STEP instead adopts a whole-video formulation and simultaneously handles spatial and temporal dimensions within a PnPDP framework. This enables us to deal with more general problems in which the extraction

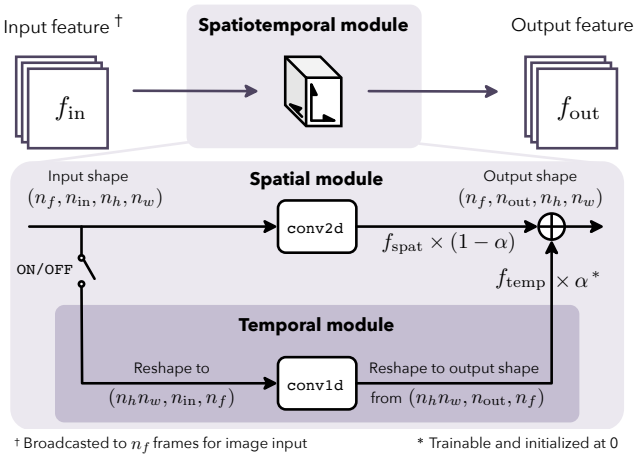


Figure 7.8: **Architecture of the spatiotemporal module.** Given a pre-trained image diffusion U-Net, we add a zero-initialized temporal module with an `ON/OFF` switch to each 2D spatial module and initialize the additive weight α to zero. Thus, it will have no effect at the start of fine-tuning and gradually learn from the video training data. The number of frames, height, and width are denoted by n_f, n_h, and n_w, respectively. The numbers of channels for input features (f_{in}) and output features (f_{out}) are denoted by n_{in} and n_{out}, respectively.

of temporal information directly from measurements is hard.

7.4.2.1 Efficient Training of Spatiotemporal Diffusion Priors

We propose to obtain spatiotemporal diffusion priors for scientific video inverse problems efficiently via the following three steps.

Step 1: Training a latent diffusion model (LDM) as an image prior We start by training a VAE [163] using the standard ℓ_1 reconstruction loss with a scaled KL divergence loss on an image dataset. The KL divergence scaling factor is set to much less than 1 to prevent excessive regularization of the latent space. This allows us to obtain an image encoder $\mathcal{E}$ and decoder $\mathcal{D}$. Once they are trained, we fix their parameters and train a 2D U-Net model $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z}_t; \sigma_t)$ using the standard denoising score matching loss [292]. Despite recent progress in 3D spatiotemporal encoders and decoders [55, 319, 332], we opt for a 2D spatial encoder and decoder to process each frame independently. This choice is due to efficiency considerations, as the decoder $\mathcal{D}$ will be called repeatedly during inference.

Step 2: Turning an image prior into a spatiotemporal prior Upon obtaining an LDM as an image prior, we use a spatiotemporal U-Net architecture to parameterize

the time-dependent video score function, i.e., $s_\theta(z_t; \sigma_t) \approx \nabla_{z_t} \log p_t(z_t; \sigma_t)$, leveraging recent advancements in video generation [133, 303]. The key component in the architecture is the spatiotemporal module for 3D modeling, as illustrated in Figure 7.8. Given a pre-trained IDM with a 2D U-Net, we introduce a zero-initialized temporal module for each 2D spatial module within the U-Net. Specifically, for an input feature f_{in}, let f_{out} be the output of the spatiotemporal module, with f_{spat} and f_{temp} representing the outputs of the spatial and temporal branches, respectively. These features are combined using an α-blending mechanism:

$$f_{\text{out}} = (1 - \alpha) \cdot f_{\text{spat}} + \alpha \cdot f_{\text{temp}}, \tag{7.7}$$

where $\alpha \in \mathbb{R}$ is a learnable parameter initialized as 0 in each spatiotemporal module. This design allows us to inherit the weights of the 2D spatial modules from the pre-trained IDM, significantly reducing training time. Additionally, by factorizing the 3D module into a 2D spatial module and a 1D temporal module, the spatiotemporal U-Net has marginal computational overhead compared to the original 2D U-Net, striking a good balance between model capacity and efficiency.

Step 3: Image-video joint fine-tuning To further improve performance and be compatible with both image and video inputs, we introduce an `ON/OFF` switch signal in the spatiotemporal module. When the switch is set to `OFF` (indicating image inputs), the temporal module is disabled (or equivalently $\alpha = 0$). This ensures that $f_{\text{out}} = f_{\text{spat}}$ and reduces the spatiotemporal module to the original 2D spatial module, which processes each frame independently. During training, we initialize the weights of the spatial modules based on a pre-trained image diffusion model and fine-tune all parameters of the spatiotemporal U-Net using both image and video data. During fine-tuning, the model receives video data with probability $p_{\text{joint}} \in [0, 1]$ and receives images (with switch set to OFF accordingly) with probability $1 - p_{\text{joint}}$. The probability p_{joint} is a tunable hyperparameter controlling the proportion of real video data in training. Pseudo-video regularization helps the spatiotemporal U-Net retain the spatial capabilities of the initialized spatial U-Net. This strategy, proven effective in previous work [303], stabilizes training and prevents overfitting to the video dataset.

7.4.2.2 Likelihood and Prior Steps

As shown in Section 7.3.1, DCDP [181], PnP-DM (Chapter 6), and DAPS [341] share the same prior step and have slight differences in the likelihood step. We

find that STEP works the best with the DAPS instantiation due to its better compatibility with latent diffusion models. See the last paragraph of Section 7.3.2.2 for a discussion on this. For the likelihood step (Line 2 of Algorithm 2), STEP first samples

$$\widetilde{\boldsymbol{v}}^{(k)} \sim \exp\left(-\frac{1}{2\sigma_{\boldsymbol{y}}^2}\left\|\mathcal{A}(\mathcal{D}(\boldsymbol{v})) - \boldsymbol{y}\right\|_2^2 - \frac{1}{2\eta_k^2}\left\|\boldsymbol{u}^{(k)} - \boldsymbol{v}\right\|_2^2\right)/Z \tag{7.8}$$

where Z is a normalizing constant, and then samples

$$\boldsymbol{v}^{(k)} \sim \mathcal{N}(\widetilde{\boldsymbol{v}}^{(k)}, \eta_k^2 \boldsymbol{I}). \tag{7.9}$$

For the prior step (Line 3 of Algorithm 2), STEP samples $\boldsymbol{u}^{(k+1)}$ from the following distribution

$$\boldsymbol{u}^{(k+1)} \sim \exp\left(\log p(\boldsymbol{u}) - \frac{1}{2\eta_k^2}\left\|\boldsymbol{u} - \boldsymbol{v}^{(k)}\right\|_2^2\right)/Z \tag{7.10}$$

where Z is a normalizing constant. As shown in Section 6.3, this sampling task is equivalent to solving a Gaussian denoising problem and can be implemented rigorously using the EDM framework given a pre-trained DM for $p(\boldsymbol{u})$ [151]. The pseudocode and more technical details of the STEP with DAPS are provided in Appendix D.2.1.

7.4.3 Experiments

7.4.3.1 Tasks and Setup

We consider two scientific video inverse problems: black hole video reconstruction [91] and dynamic MRI [110]. Although both are scientific imaging tasks, they have significantly different characteristics. Black hole video reconstruction involves simple spatial structures (usually a ring structure) but complex temporal dynamics that obey certain physical constraints. In contrast, dynamic MRI requires higher spatial fidelity with relatively simpler temporal dynamics, such as periodic heartbeat motion.

Black Hole Video Reconstruction We consider the problem of observing the Sagittarius A* black hole using the Event Horizon Telescope (EHT) array in 2017 [91]. The measurements consist of sparse vectors with a dimensionality of 1,856 derived from an underlying black hole video comprising 64 frames, each with a spatial resolution of 256×256 pixels, capturing the black hole's dynamics during the 100-minute observation period. These measurements are given by calculating

the closure quantities based on the complex visibilities (a detailed description of the problem is available in Appendix D.2.2.1). For the training dataset, we employ the General Relativistic MagnetoHydroDynamic (GRMHD) simulations [316] of Sagittarius A*, covering various black hole models and observational conditions. The dataset comprises 648 simulated black hole videos and 50,000 black hole images.

Dynamic MRI We consider a standard compressed sensing MRI setup with two acceleration scenarios: 8× acceleration with 12 auto-calibration signal (ACS) lines and 6× acceleration with 24 ACS lines. Further details are provided in Appendix D.2.2.2. We utilize the publicly available cardiac cine MRI dataset from the CMRxRecon Challenge 2023 [298] for training. This dataset includes 3,324 cardiac MRI sequences, each containing fully sampled, ECG-triggered k-space data from 300 patients, featuring various canonical cardiac imaging views. Each sequence is processed into a video consisting of 12 frames with a spatial resolution of 192×192 pixels. We construct the image dataset by extracting the individual frames from the videos, resulting in 39,888 images in total.

7.4.3.2 Baselines and Our Method

Recall from Section 7.4.2 that our method leverages a learned spatiotemporal prior to avoid explicitly estimating temporal dynamics from measurements at inference time. To evaluate the effectiveness of this strategy, we compare our method to existing IDM-based approaches. Based on how temporal information is incorporated, we categorize these baseline methods into two groups as follows.

Group 1: Simple Heuristics We consider two baselines: Batch Independent Sampling (BIS) and Batch Consistent Sampling (BCS) [168]. BIS reconstructs each frame independently, while BCS implicitly incorporates a static temporal prior to enforce consistency across frames.

Group 2: Noise Warping We include another line of baselines: $\int$-noise [53, 82], and GP-Warp [77]. These baselines enforce temporal consistency by warping the noise using optical flow estimated directly from the measurements. Besides, we include more conventional warping strategies such as *Bilinear*, *Bicubic*, *Nearest* from work [53]. For the black hole video reconstruction task, a direct inversion from the sparse measurement vector is infeasible. Thus, to demonstrate the upper-bound performance of these methods, we leverage the ground truth video as an oracle to

derive the optical flow. For the dynamic MRI task, we utilize a naïve inversion obtained via inverse Fourier transformation to estimate the optical flow. We follow the methodology of [53], leveraging a pre-trained model [278] to extract optical flow.

Ours: Two Variants of STeP We evaluate two variants differentiated by their spatiotemporal priors: STeP (video-only), which uses a spatiotemporal prior trained exclusively on video data, and STeP (image-video joint), which initializes with a pre-trained IDM and undergoes joint image-video fine-tuning. We follow the procedure in Appendix D.2.3 to train our priors until convergence. The detailed training hyperparameters are summarized in Table D.3. To make a fair comparison, we run all the baselines with DAPS [341]. The detailed implementation of each baseline can be found at Appendix D.2.1 and Appendix D.2.2.3.

7.4.3.3 Metrics

We evaluate our results on three aspects: (1) spatial similarity, (2) temporal consistency, and (3) measurement data fit.

Spatial Spatial similarity is assessed by calculating Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [310], and Learned Perceptual Image Patch Similarity (LPIPS) [349]. Metrics are computed per frame and averaged across all frames. We utilize implementations from `piq` [153], normalizing images to the range $[0, 1]$. For grayscale images, frames are replicated across three channels before computing LPIPS scores.

Temporal Temporal consistency is quantified using delta-based PSNR (d-PSNR) and delta-based SSIM (d-SSIM), measuring the similarity of normalized differences between consecutive frames, with results averaged over all frames. Additionally, we compute the Fréchet Video Distance (FVD) [287] between the reconstructed videos and our test dataset to evaluate distributional similarity.

Data Fit Lastly, we report measurement data fit using task-specific metrics. For the black hole video reconstruction task, we employ the unified average $\tilde{\chi}^2$ statistic (defined in Equation (D.14)), where values closer to 1 indicate better data fidelity. For dynamic MRI, we measure data misfit by computing the mean squared error in measurement space.

Table 7.2: **Quantitative results on black hole video reconstruction and dynamic MRI.** We compare our method against baselines by reporting the mean and standard deviation (shown in parentheses) of selected evaluation metrics computed over 10 test videos (FVD is reported without standard deviation since it evaluates the set of 10 videos collectively). The results clearly demonstrate that by leveraging the spatiotemporal prior, STEP consistently achieves improvements in both spatial quality and temporal consistency relative to baseline methods.

Tasks	Methods	PSNR (↑)	SSIM (↑)	LPIPS (↓)	d-PSNR (↑)	d-SSIM (↑)	FVD (↓)	Data Misfit (↓)
Black hole	BIS [168]	23.79 (1.41)	0.718 (0.047)	0.179 (0.031)	29.26 (1.51)	0.938 (0.015)	1429.42	1.719 (1.277)
	BCS [168]	27.66 (2.04)	0.816 (0.053)	0.124 (0.040)	41.71 (1.99)	**0.979** (0.008)	564.43	1.426 (0.784)
	Bilinear [53]	26.11 (2.05)	0.718 (0.067)	0.151 (0.044)	33.14 (2.33)	0.958 (0.013)	1335.95	1.384 (0.742)
	Bicubic [53]	25.68 (1.88)	0.730 (0.058)	0.163 (0.041)	31.70 (1.88)	0.952 (0.015)	1521.14	1.736 (1.259)
	Nearest [53]	25.29 (1.80)	0.754 (0.059)	0.164 (0.042)	30.76 (1.75)	0.943 (0.017)	1171.07	1.691 (1.020)
	$\int$-noise [53, 82]	24.90 (1.52)	0.745 (0.058)	0.163 (0.034)	31.87 (1.93)	0.945 (0.017)	1253.81	1.655 (1.020)
	GP-Warp [77]	23.98 (1.28)	0.721 (0.043)	0.176 (0.029)	29.21 (1.24)	0.938 (0.014)	1395.15	1.721 (1.225)
	STEP (video only)	28.71 (1.81)	0.802 (0.079)	0.120 (0.041)	41.41 (2.39)	0.975 (0.011)	238.36	1.124 (0.136)
	STEP (image-video joint)	**30.28** (2.71)	**0.865** (0.063)	**0.095** (0.039)	**42.09** (2.63)	0.976 (0.011)	**170.67**	**1.114** (0.154)
MRI (8×)	BIS [168]	35.04 (0.76)	0.889 (0.016)	0.100 (0.012)	38.91 (1.13)	0.918 (0.012)	82.30	9.203 (0.698)
	BCS [168]	35.43 (0.97)	0.893 (0.016)	0.099 (0.012)	39.91 (1.09)	0.931 (0.011)	95.68	9.225 (0.693)
	Bilinear [53]	35.30 (0.96)	0.896 (0.016)	0.098 (0.012)	39.87 (1.14)	0.931 (0.011)	87.90	9.157 (0.670)
	Bicubic [53]	35.31 (0.91)	0.896 (0.016)	0.098 (0.011)	40.11 (1.10)	0.934 (0.010)	108.90	9.189 (0.653)
	Nearest [53]	34.87 (0.90)	0.895 (0.018)	0.099 (0.013)	40.09 (1.14)	0.933 (0.011)	108.57	9.188 (0.656)
	$\int$-noise [53, 82]	35.55 (1.03)	0.892 (0.020)	0.099 (0.014)	39.77 (1.36)	0.929 (0.015)	89.59	9.208 (0.693)
	GP-Warp [77]	34.49 (0.65)	0.886 (0.016)	0.102 (0.012)	38.71 (1.14)	0.916 (0.013)	92.19	9.209 (0.659)
	STEP (video only)	37.00 (1.46)	0.927 (0.019)	0.086 (0.013)	43.50 (2.70)	0.963 (0.015)	**75.27**	8.817(0.650)
	STEP (image-video joint)	**39.38** (1.16)	**0.951** (0.009)	**0.078** (0.011)	**44.86** (1.92)	**0.974** (0.006)	78.60	**8.753** (0.603)

7.4.3.4 Main Results

We summarize the main results as the following four observations.

STEP reconstructs videos with better spatial and temporal coherence. We provide quantitative results in Table 7.2 and visual comparisons in Figure 7.9 and Figure 7.10. STEP outperforms baselines in overall video quality across most metrics and generates video reconstructions with significantly better spatiotemporal coherence. Figure 7.9 visualizes an x-t slice, representing the temporal evolution (horizontal axis) of a spatial slice (marked by the cyan vertical line). We find that the noise warping baselines fail to constrain temporal consistency, as indicated by the fluctuating ring diameters in the x-t slices, while BCS provides almost static reconstructions. In contrast, STEP exhibits closer alignment with the ground truth and improved temporal consistency. This is further illustrated by the averaged optical flow visualization, highlighting that STEP faithfully captures the underlying temporal dynamic. Similar trends can be observed in Figure 7.10.

Noise warping is less effective in scientific problems. We observe that the performance of noise-warping strategies in challenging scientific VIPs does not correlate with noise-warping accuracy. For example, although the $\int$-noise approach has demonstrated superior noise-warping capability given optical flow (as shown

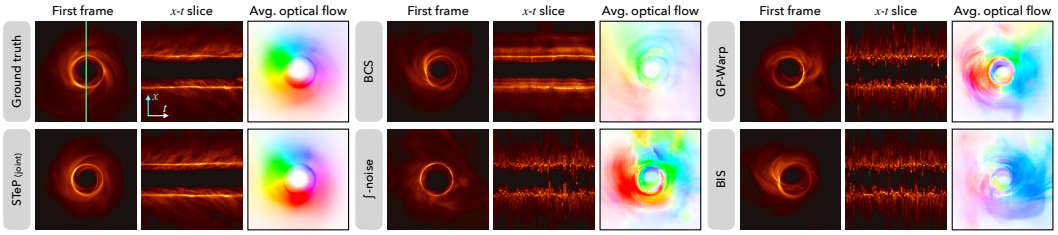


Figure 7.9: **Visual examples of STeP (bottom left) and baselines for black hole video reconstruction.** To facilitate analysis of the reconstructed spatiotemporal structures, we present results in three ways: (1) a single frame to illustrate spatial fidelity, (2) an x-t slice depicting temporal evolution of a vertical line to evaluate temporal consistency, and (3) the averaged optical flow visualized using the standard color scheme from [278] to assess spatiotemporal coherence jointly. Compared to baselines, STeP exhibits clearer alignment with ground truth videos across all aspects.

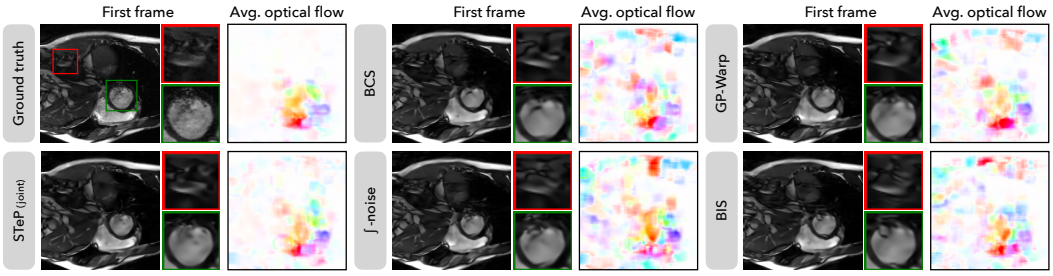
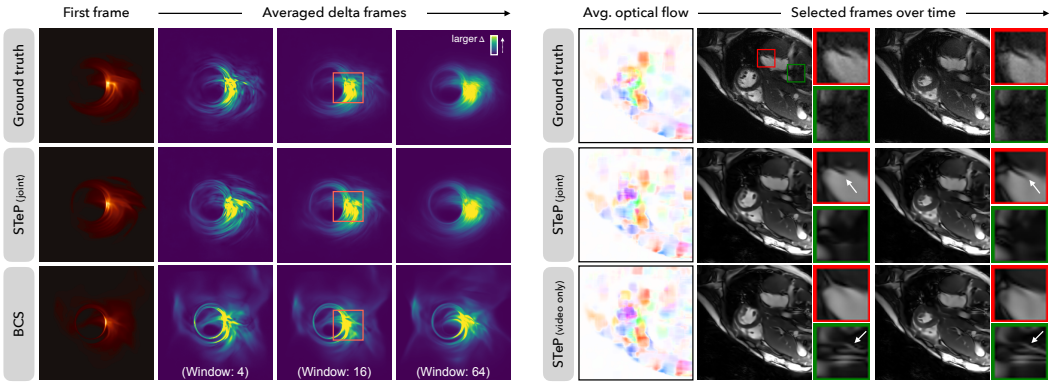


Figure 7.10: **Visual examples of STeP (bottom left) and baselines for dynamic MRI.** We visualize a representative frame along with two zoomed-in regions for each method to better illustrate spatial fidelity. Benefiting from its robust spatiotemporal prior, STeP provide reconstructions with fewer structural artifacts and temporal fluctuations, as indicated by its averaged optical flow aligning more closely with the ground truth. This demonstrates the effectiveness of our learned spatiotemporal prior in enhancing both spatial and temporal consistency.

in [53]), it underperforms simpler strategies such as *Bilinear* and *Bicubic* interpolation in challenging VIPs. This discrepancy arises primarily due to: (1) inaccuracies in the optical flow, and (2) the inherent difficulty of effectively manipulating noise in latent space through pixel space optical-flow-guided warping. Consequently, methods relying on precise optical flow and carefully designed warping strategies struggle with tasks requiring high temporal consistency, such as black hole video reconstruction. In contrast, our approach employs a data-driven spatiotemporal prior learned from training data, effectively overcoming these limitations and enabling broader applicability to scientific tasks with significant temporal uncertainty.



(a) better temporal consistency (b) less spatial hallucinations

Figure 7.11: **Detailed comparison on black hole video reconstruction and dynamic MRI.** Left: We compare STeP (joint) and the BCS baseline [168] by visualizing the averaged delta frames (difference images) over an expanding window. The delta frames given by STeP (joint) better align with the ground truth, indicating better temporal consistency. Right: We also compare the spatial fidelity between STeP (joint) and its variant STeP (video-only). Trained on both images and videos, STeP (joint) provide reconstructions with less spatial hallucinations compared to STeP (video-only).

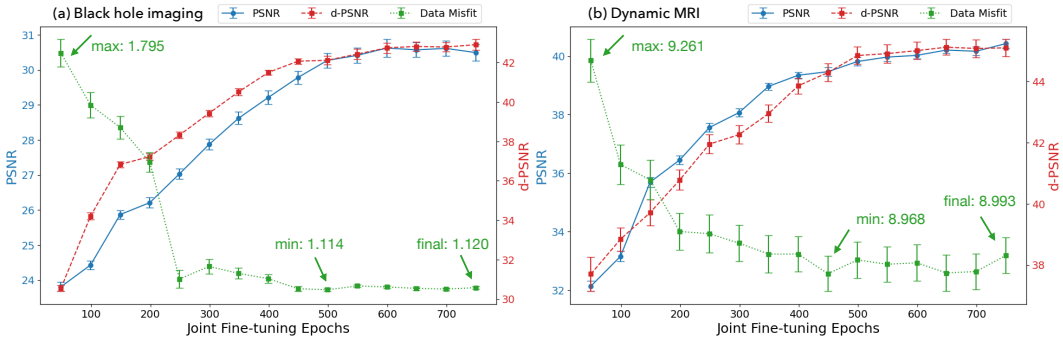


Figure 7.12: **Consistent improvement in image-video joint fine-tuning.** We evaluate intermediate checkpoints of (a) black hole video reconstruction and (b) dynamic MRI (8×). Spatial similarity (measured by PSNR), temporal consistency (measured by d-PSNR), and measurement data fit (measured by data misfit) all show steady improvement.

Joint fine-tuning benefits both spatial and temporal consistency. As shown in Table 7.2, STeP with an image-video jointly fine-tuned spatiotemporal prior outperforms both IDM-based baselines and STeP trained video data only. To further illustrate these improvements, we provide a detailed visual comparison in Figure 7.11. In Figure 7.11 (a), we analyze the temporal dynamics using averaged delta frames, which visualize the averaged differences between consecutive frames over a window. We observe that delta frames from BCS remain largely unchanged as the temporal window expands, indicating limited temporal variation. In contrast,

STEP (joint) effectively captures more complex and coherent temporal dynamics. In Figure 7.11 (b), we compare STEP (joint) and STEP (video only) with a focus on spatial fidelity. Although their reconstructions have similar average optical flow, STEP (joint) have fewer artifacts and hallucinations as pointed out by the arrows. Such an improvement highlights the value of using image data in both the pre-training and fine-tuning stages.

STEP provides diverse and equally plausible solutions. Due to the highly ill-posed and extremely sparse measurements in black hole video reconstruction, STEP can produce multiple semantically diverse reconstructions, each visually plausible and consistent with measurement data. Since the true posterior distribution is unknown, directly quantifying mode coverage is infeasible. Instead, we draw 100 *i.i.d.* samples and cluster them based on spatial appearance and temporal dynamics. As illustrated in Figure 7.6, we identify three distinct modes with nearly identical data fidelity (see Table D.5 exact data misfit values). One recovered mode aligns closely with the ground truth, while the others differ in rotation direction or spatial structure. This demonstrates that our method not only accurately reconstructs the underlying ground truth but also discovers additional plausible solutions.

7.4.3.5 Ablation on Image-Video Joint Finetuning

To show the effectiveness of the proposed image-video joint finetuning technique, we show the quantitative results of using checkpoints of the spatiotemporal U-Net that were fine-tuned for different numbers of epochs. We assess performance using PSNR (blue curve), d-PSNR (red curve), and a data-fitting metric (green curve), as shown in Figure 7.12. Since the spatiotemporal U-Net is initialized from a pre-trained image diffusion model, these curves indicate steady improvement in spatiotemporal consistency and data fitting as the prior is fine-tuned.

7.4.3.6 Discussion on the Choice of Inference Algorithm

In this section, we use DAPS rather than PNP-DM as the base algorithm for inference due to the former's better compatibility with latent DMs (as discussed in Section 7.3.2.2 for text-conditioned DMs). Here we deepen the investigation by running STEP with PNP-DM as the inference algorithm and compare it to the DAPS-based variant. As shown in Figure 7.13, DAPS leads to significantly better visual quality in terms of both improved spatial consistency in individual frames (top row)

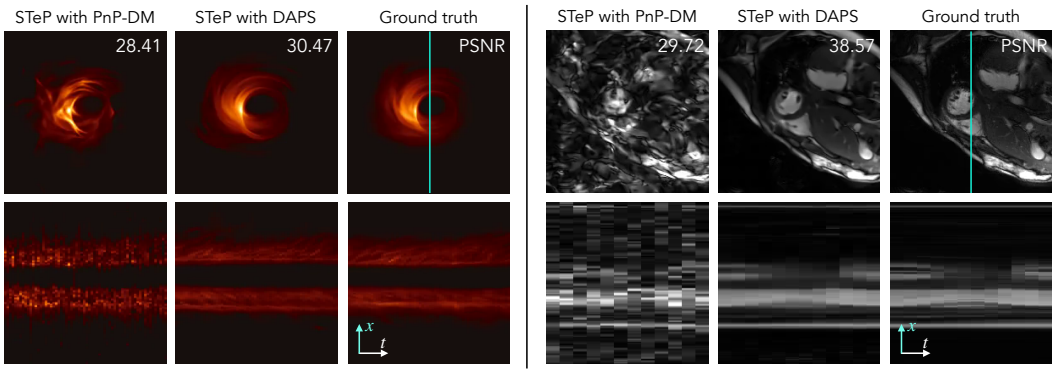


Figure 7.13: **Comparison between running STeP with DAPS versus PnP-DM as the inference backbone.** The DAPS version of STeP exhibits significantly better spatial consistency (shown by the first frames in the first row) and temporal consistency (shown by the x-t slice in the second row). This comparison illustrates the better compatibility of DAPS with latent video diffusion models than PnP-DM.

and enhanced temporal coherence in the x–t slices (bottom row). Quantitatively, DAPS achieves higher PSNR in both the black hole video (30.47 dB vs. 28.41 dB) and dynamic MRI reconstruction (38.57 dB vs. 29.72 dB), highlighting its superior compatibility with latent video diffusion models over PnP-DM.

7.5 Thrust 3: Accommodating Black-Box Forward Models

Many inverse problems in physical sciences involve partial differential equations (PDEs), such as weather forecasting [313], geophysics [192], and fluid reconstruction [94]. The forward models in these problems may involve complicated numerical algorithms where reliable computation of derivatives is challenging or even infeasible. These problems are beyond the scope of PnP-DM in Chapter 6, which assumes the differentiability of $\mathcal{A}$. In this section, we introduce Blade, an instantiation of Algorithm 2 for derivative-free, ensemble-based posterior estimation. We provide a convergence analysis of Blade and provide explicit bounds on the sampling error. We also show the performance of Blade on a challenging nonlinear fluid dynamics problem. Blade achieves both higher predictive accuracy and better probabilistic calibration compared to competing methods. The appendix for this section is Appendix D.3.

7.5.1 Instantiation of Algorithm 2 with Derivative-Free Likelihood Step

The name Blade is derived from the key components of the algorithm: **B**ayesian inversion, **L**inearization, **A**lternating updates, **D**erivative-free, and **E**nsemble. We refer readers to Appendix D.3.2 for a brief review of the relevant background. In

Algorithm 3 BLADE for Derivative-Free Diffusion-Based Posterior Estimation

Require: initial ensemble $\boldsymbol{X}^{(0)} = \{\boldsymbol{x}^{(j)} \in \mathbb{R}^n\}_{j=1}^J$, number of iterations K, $\{\eta_k\}_{k=0}^{K-1}$, likelihood potential $f(\cdot\,; \boldsymbol{y})$ with measurements $\boldsymbol{y} \in \mathbb{R}^m$, pre-trained diffusion model $\boldsymbol{s}_{\boldsymbol{\theta}}$.

1: **for** $k = 0, ..., K-1$ **do**
2: $\quad \mathbf{Z}^{(k)} \leftarrow \texttt{LikelihoodStep}(\boldsymbol{X}^{(k)}, \boldsymbol{y}, \eta_k)$ ▷ Algorithm 8 (Section 7.5.1.1)
3: $\quad \boldsymbol{X}^{(k+1)} \leftarrow \texttt{PriorStep}(\mathbf{Z}^{(k)}, \boldsymbol{s}_{\boldsymbol{\theta}}, \eta_k)$ ▷ Algorithm 9 (Section 7.5.1.2)
4: **end for**
5: **return** $\boldsymbol{X}^{(K)}$

a nutshell, BLADE iteratively updates an ensemble of interacting particles by alternating between a derivative-free likelihood sampling step and a denoising diffusion prior step. We provide pseudocode for the complete sampling algorithm in Algorithm 3. Section 7.5.1.1 details the derivative-free likelihood step achieved through statistical linearization using an ensemble of particles. Section 7.5.1.2 describes the prior step with a denoising DM. At the same time, the noise schedule η_k is gradually annealed towards zero. Further details are deferred to Appendix D.3.3.

7.5.1.1 Derivative-Free Likelihood Step via Statistical Linearization

Let $\boldsymbol{X}^{(k)} = \{\boldsymbol{x}^{(j)}\}_{j=1}^J$ denote the ensemble of J particles at k-th alternating iteration of the SGS framework. Recall from Chapter 6 that, in the likelihood step, we aim to sample $\boldsymbol{z}^{(j)}$ from $\pi^{Z|X=\boldsymbol{x}^{(j)}}(\boldsymbol{z}) \propto \exp(-f(\boldsymbol{z}; \boldsymbol{y}) - \frac{1}{2\eta^2}\|\boldsymbol{z} - \boldsymbol{x}^{(j)}\|_2^2)$ for each $j \in \{1, \ldots, J\}$ where $f(\boldsymbol{z}; \boldsymbol{y}) = \frac{1}{2\sigma_{\boldsymbol{y}}^2}\|\mathcal{A}(\boldsymbol{z}) - \boldsymbol{y}\|_2^2$ for inverse problems of form Equation (5.1) assuming that $\boldsymbol{n} \sim \mathcal{N}(\mathbf{0}, \sigma_{\boldsymbol{y}}^2 \boldsymbol{I})$.

Statistical Linearization Following [111, 143], we consider the covariance-preconditioned Langevin dynamics with the large particle limit

$$\mathrm{d}\boldsymbol{z}_t^{(j)} = -C_t \nabla \left(f(\boldsymbol{z}_t^{(j)}; \boldsymbol{y}) + \frac{1}{2\eta^2}\|\boldsymbol{z}_t^{(j)} - \boldsymbol{x}^{(j)}\|_2^2 \right) \mathrm{d}t + \sqrt{2C_t}\mathrm{d}\boldsymbol{w}_t, \tag{7.11}$$

where $C_t := \mathbb{E}_{q_t}[(\boldsymbol{z}_t - \bar{\boldsymbol{z}}_t)(\boldsymbol{z}_t - \bar{\boldsymbol{z}}_t)^T]$ with $\bar{\boldsymbol{z}}_t := \mathbb{E}_{q_t}[\boldsymbol{z}_t]$ and q_t is the particle distribution. Since the gradient of f with respect to z involves the gradient of $\mathcal{A}$ with respect to z, we approximate the gradient of f using the statistical linearization

technique, i.e.,

$$\nabla f(z_t^{(j)}; \boldsymbol{y}) = \frac{1}{\sigma_{\boldsymbol{y}}^2}(D^T\mathcal{A})(\mathcal{A}(z_t^{(j)}) - \boldsymbol{y}) \approx \frac{1}{\sigma_{\boldsymbol{y}}^2} C_t^{-1}\mathbb{E}_{q_t}[\widetilde{z}_t(\mathcal{A}(z_t) - \mathbb{E}_{q_t}\mathcal{A}(z_t))^T](\mathcal{A}(z_t^{(j)}) - \boldsymbol{y}). \tag{7.12}$$

where $D^T\mathcal{A}$ denotes the adjoint of the Jacobian of $\mathcal{A}$ and $\widetilde{z}_t = z_t - \bar{z}_t$. The approximation is given by substituting $D^T\mathcal{A}$ with the adjoint (i.e., transpose) of

$$\boldsymbol{A}_t := \mathbb{E}_{q_t}[(\mathcal{A}(z_t) - \mathbb{E}_{q_t}\mathcal{A}(z_t))(z_t - \bar{z}_t)^T]C_t^{-1}, \tag{7.13}$$

which is the least-square linear approximation of $\mathcal{A}$. Substituting Equation (7.12) into Equation (7.11) gives

$$\mathrm{d}z_t^{(j)} = -\left[\frac{1}{\sigma_{\boldsymbol{y}}^2}\mathbb{E}_{q_t}[\widetilde{z}_t(\mathcal{A}(z_t) - \mathbb{E}_{q_t}\mathcal{A}(z_t))^T](\mathcal{A}(z_t^{(j)}) - \boldsymbol{y}) + \frac{1}{\eta^2}C_t(z_t^{(j)} - \boldsymbol{x}^{(j)})\right]\mathrm{d}t + \sqrt{2C_t}\mathrm{d}\boldsymbol{w}_t. \tag{7.14}$$

Thanks to the covariance preconditioner, the dynamics in Equation (7.14) avoid computing C_t^{-1} in $\boldsymbol{A}_t$. Further, Equation (7.14) eliminates the reliance on the forward model's derivatives, allowing us to run the algorithm with only black-box access to $\mathcal{A}$.

Practical Implementation We then implement the dynamics in Equation (7.14) with a finite-particle system in practice and ensure that the invariant measure of the finite-particle system remains the same as that of Equation (7.14). As shown in [226], the covariance-preconditioned stochastic process requires an additional correction term as the diffusion term depends on the evolving particle. For the j-th particle, we add a correction term to the drift of Equation (7.14), yielding

$$\mathrm{d}z_t^{(j)} = -\left[\frac{1}{\sigma_{\boldsymbol{y}}^2}C_t\boldsymbol{A}_t^T(\mathcal{A}(z_t^{(j)}) - \boldsymbol{y}) + \frac{1}{\eta^2}C_t(z_t^{(j)} - \boldsymbol{x}^{(j)})\right]\mathrm{d}t + \sqrt{2C_t}\mathrm{d}\boldsymbol{w}_t + \frac{n+1}{J}(z_t^{(j)} - \bar{z}_t)\mathrm{d}t \tag{7.15}$$

where n is the dimensionality of z and J is the ensemble size. Lemma D.3.7 verifies that Equation (7.15) has an invariant measure that is identical to that of Equation (7.14). Intuitively, the correction term $\frac{n+1}{J}(z_t^{(j)} - \bar{z}_t)$ pushes the particles away from each other and vanishes when $J \gg n$. For the computation of $\sqrt{C_t}$, we use the construction proposed in [112] where $\sqrt{C_t} = \frac{1}{\sqrt{J}}\left[z_t^{(1)} - \bar{z}_t, \ldots, z_t^{(J)} - \bar{z}_t\right] \in \mathbb{R}^{n\times J}$, which avoids explicit matrix square roots. Further implementation details can be found in Appendix D.3.3.1, and the pseudocode is provided in Algorithm 8.

7.5.1.2 Ensemble-Based Prior Step via Denoising Diffusion

Let $\boldsymbol{Z}^{(k)} = z^{(j)}{}_{j=1}^{J}$ denote the ensemble of J particles at the k-th alternating iteration of the SGS framework. Recall from Chapter 6 that, in the prior step, our goal is to sample $\boldsymbol{x}^{(j)}$ from $\pi^{X|Z=z^{(j)}}(\boldsymbol{x}) \propto \exp\left(-g(\boldsymbol{x}) - \frac{1}{2\eta^2}\|\boldsymbol{x} - \boldsymbol{z}^{(j)}\|_2^2\right)$ for each $j \in 1, \ldots, J$, where $g(\boldsymbol{x}) := -\log p(\boldsymbol{x})$ denotes the potential function of the prior distribution. This corresponds to applying the prior step of PnP-DM independently to each particle in $\boldsymbol{Z}^{(k)}$[3], and we therefore refer the readers to Section 6.3.2 for details. Further implementation details are provided in Appendix D.3.3.2 and the pseudocode is available in Algorithm 9.

7.5.2 Convergence Analysis

In this sub-section, we analyze the non-asymptotic behavior of the proposed algorithm through the lens of its continuous-time and large particle limit for the ease of understanding. In practice, the proposed algorithm incurs two bias terms: ϵ_{model} from the statistical linearization and ϵ_{score} from the learned prior. By extending the existing interpolation proof techniques from Chapter 6, our analysis also quantifies how these errors affect the deviation from the reference process over K iterations. The technical definitions and notations are collected in Appendix D.3.1.1.

Theorem 7.5.1. *Given $\eta > 0$, consider the following two processes that alternate between the likelihood step with horizon $t^\dagger$ and the prior step with horizon t^*, where $\sigma(t^*) = \eta$:*

- *The approximate process that implements the likelihood step as in Equation* (7.14) *(with forward model approximation) and the prior step as in Equation* (6.6) *(with diffusion model score approximation). Let $\widetilde{\mu}_\tau$ denote its distribution at time τ, C_τ the associated covariance matrix, λ_τ^* the smallest non-zero eigenvalue of C_τ.*
- *The reference process that starts from the stationary distribution π^{XZ} and implements the likelihood step as Equation* (7.11) *with the preconditioner C_τ, and the prior step as Equation* (6.6)*, assuming exact knowledge of both the prior score function and forward model derivative. Let μ_τ denote its distribution at time τ.*

Let $T_k = k(t^\dagger + t^*), k = 0, \ldots, K$, $\lambda^* = \inf_{t \in \cup_k [T_k, T_k + t^\dagger]} \lambda_t^*$, and $\delta = \inf_{t \in [0, t^*]} \delta(t)$ where $\delta(t)$ is the diffusion term defined in Equation (D.21)*. We denote by ϵ_{score}*

[3]For Blade , we set $s(t) = 1$ for simplicity.

the score approximation error of the diffusion model defined in Assumption D.3.1, and ϵ_{model} *the forward model derivative approximation error defined in Assumption D.3.2. Assuming that* $\mathsf{KL}(\pi^X||\mu_0) < +\infty$ *and Assumption D.3.3 holds, for K iterations of* BLADE*, we have*

$$\underbrace{\frac{1}{T_K}\int_0^{T_K} \mathsf{FI}(\mu_\tau||\widetilde{\mu}_\tau)\mathrm{d}\tau}_{\substack{\textit{average Fisher divergence}\\ \textit{over K iterations of BLADE}}} \leq \underbrace{\frac{c\mathsf{KL}(\pi^X||\widetilde{\mu}_0)}{K}}_{\substack{\textit{convergence}\\ \textit{from initialization}}} + \underbrace{ct^\dagger\epsilon_{\text{model}}}_{\textit{model error}} + \underbrace{ct^*\epsilon_{\text{score}}}_{\textit{score error}} \tag{7.16}$$

where $c := \frac{4}{\min(\lambda^*,\delta)(t^\dagger+t^*)}$ *is a constant and* FI*,* KL *are Fisher divergence and KL divergence, respectively.*

The proof is provided in Appendix D.3.1.3. Theorem 7.5.1 accounts for the effect of the two generally unavoidable approximations: the statistical linearization of the forward model and the learned diffusion prior. Equation (7.16) indicates that the time-average Fisher divergence between the approximate process and reference process decays at an $O(1/K)$ rate up to a weighted sum of two approximation errors. While this bound is structurally similar to that of PNP-DM (Theorem 6.4.1), there are two key distinctions. First, our method considers fundamentally different likelihood step dynamics (with linearization and covariance preconditioning); consequently, the resulting error term depends on additional factors such as covariance matrix eigenvalues and the linearization error. Second, unlike the analysis in Section 6.4 that assumes exact likelihood steps, our analysis incorporates the effects of model error ϵ_{model} and a finite time horizon $t^\dagger$. Furthermore, this analysis offers theoretical guarantees with explicit bounds on the approximation errors, which has not been done in the existing works on the derivative-free algorithms with diffusion priors.

7.5.3 Experiments

We evaluate BLADE on the inverse problem of recovering the initial vorticity field in the 2D Navier–Stokes equations from partial, noisy observations taken at a later time. This setting mirrors many practical inverse problems in science and engineering, including weather data assimilation [313], geophysics [192], and fluid reconstruction [94].

Problem Setup We follow the general experimental setup established in INVERSEBENCH [357] (Chapter 8), using its publicly released dataset and pre-trained diffusion prior for all experiments. The true initial vorticity $\boldsymbol{x}_0$ in resolution 128×128

Table 7.3: **Comparison on the Navier-Stokes inverse problem.** Metrics are abbreviated as follows: Rel ℓ_2 (relative ℓ_2 error), CRPS (continuous ranked probability score), and SSR (spread-skill ratio). − indicates either that probabilistic metrics are inapplicable (deterministic models) or that it is too expensive to generate enough samples from the algorithm for reliable calculation. **Bold** and Underline indicate best and second best among methods that accommodate unpaired data, respectively.

	$\sigma_{\boldsymbol{y}} = 0$			$\sigma_{\boldsymbol{y}} = 1.0$			$\sigma_{\boldsymbol{y}} = 2.0$		
	Rel $\ell_2 \downarrow$	CRPS $\downarrow$	SSR $\rightarrow 1$	Rel $\ell_2 \downarrow$	CRPS $\downarrow$	SSR $\rightarrow 1$	Rel $\ell_2 \downarrow$	CRPS $\downarrow$	SSR $\rightarrow 1$
Paired data									
CDM	1.362	2.900	0.983	1.409	2.872	1.059	1.542	2.993	1.087
U-Net	0.585	–	–	0.702	–	–	0.709	–	–
Unpaired data									
EKI	0.577	2.303	0.012	0.586	2.350	0.118	0.673	2.700	0.011
EKS + DM	0.539	1.900	0.181	0.606	2.088	0.218	0.685	2.255	0.280
DPG	0.325	–	–	0.408	–	–	0.466	–	–
SCG	0.961	–	–	0.928	–	–	0.966	–	–
EnKG	0.120	0.395	0.164	**0.191**	0.651	0.154	**0.294**	1.032	0.144
BLADE (ours)	**0.110**	**0.216**	**0.955**	0.229	**0.453**	**0.950**	0.306	**0.608**	**0.949**

is evolved forward with a numerical solver, then subsampled and corrupted with Gaussian noise of standard deviation $\sigma_{\boldsymbol{y}} = 0, 1, 2$. The measurements $\boldsymbol{y}$ thus constitute a partial, noisy snapshot of the flow field. More details can be found in Appendix D.3.5.

Baselines We compare our algorithm against two classes of methods. The first class is the methods that only require diffusion prior trained on unpaired data, including DPG [277], SCG [141], EnKG [356], EKI [143], EKS [111] (initialized from diffusion prior). The second class is provided as reference points, which requires additional training on paired data, including a conditional diffusion model (CDM) and an end-to-end UNet. The conditional diffusion learns the posterior distribution through conditional score matching. The UNet directly learns to predict the ground truth from the observation. For each noise regime, we re-train both the conditional diffusion model and the U-Net from scratch, using the same training configuration. More details can be found in Appendix D.3.4.

Evaluation Metrics For comprehensive evaluation, we consider three different metrics in the literature [241, 357] to assess the performance from both deterministic and probabilistic perspectives: relative ℓ_2 error, continuous ranked probability score (CRPS), and spread-skill ratio (SSR). CRPS is a proper scoring rule that rewards both sharp predictive distribution and well-calibrated predictions, whereas SSR diagnoses calibration only. An SSR near one is desirable, but must be interpreted

in conjunction with the other error metrics. Formal definitions and implementation details of these metrics are in Appendix D.3.5.2.

Results Table 7.3 summarizes performance under three observation noise levels. BLADE offers the best calibrated ensemble predictions: its CRPS is the best among all, and its SSR remains close to one. The other competing methods are too confident (SSR < 0.2) and their predictions do not represent the true uncertainty. The CRPS of CDM has a very high CRPS despite SSR near one, which means that it produces an overly diffuse distribution with large errors. Overall, BLADE achieves both sharp predictive performance and reliable uncertainty calibration.

7.6 Thrust 4: Tackling Inverse Problems in Discrete Spaces

Finally, we present SGDD, an instantiation of Algorithm 2 for posterior estimation in discrete spaces using discrete diffusion models (DMs). While existing works on diffusion-based posterior estimation have been focusing on continuous domains [64, 320, 341], designing analogous techniques to discrete-state spaces remains challenging due to the lack of well-defined gradients in both the likelihood and prior. To address this, we formulate a discrete version of the Split Gibbs Sampler (SGS) with a generalized regularization potential $D(\boldsymbol{x}, \boldsymbol{z}; \eta)$, extending the standard ℓ_2 term to functions compatible with discrete DMs under certain limiting conditions. This relaxed formulation preserves the alternating update structure of Algorithm 2 and enables seamless integration of pre-trained discrete DMs. We prove theoretical convergence guarantees for SGDD, with error bounds accounting for discretization and score approximation. Empirically, we show the effectiveness of SGDD on inverse problems on discrete images, where SGDD outperforms baselines by more than 8 dB in PSNR. The appendix for this section is Appendix D.4.

7.6.1 Discrete Diffusion Models

Recent works on discrete diffusion models extend score-based generative methods from modeling continuous distributions in Euclidean spaces to categorical distributions in discrete-state spaces [11, 40, 198, 259]. Specifically, when the data distribution lies in a finite support $\mathcal{X} = \{1, \dots, N\}$, one can evolve a family of categorical distributions p_t over $\mathcal{X}$ following a continuous-time Markov chain

$$\partial_t p_t = \boldsymbol{Q}_t^{\mathrm{fw}} p_t, \tag{7.17}$$

where $p_0 = p_{\mathrm{data}}$ and $\boldsymbol{Q}_t^{\mathrm{fw}} \in \mathbb{R}^{N \times N}$ are diffusion matrices with a simple stationary distribution. To reverse this continuous-time Markov chain, it suffices to learn the

concrete score $s(\boldsymbol{x}; t) := \left[\frac{p_t(\tilde{\boldsymbol{x}})}{p_t(\boldsymbol{x})}\right]_{\tilde{\boldsymbol{x}} \neq \boldsymbol{x}}$, as the reverse process is given by

$$\partial_t p_{T-t} = \boldsymbol{Q}_{T-t} p_{T-t} \tag{7.18}$$

with $\boldsymbol{Q}_t^{[i,j]} = \frac{p_t(\boldsymbol{x}_i)}{p_t(\boldsymbol{x}_j)} \boldsymbol{Q}_t^{\mathrm{fw}[j,i]}$ and $\boldsymbol{Q}_t^{[i,i]} = -\sum_{\boldsymbol{x}_j \neq \boldsymbol{x}_i} \boldsymbol{Q}_t^{[j,i]}$.

For sequential data $\boldsymbol{x} \in \mathcal{X}^n$, there are N^n states in total. Instead of constructing an exponentially large diffusion matrix, we use a sparse matrix $\boldsymbol{Q}_t^{\mathrm{fw}}$ that perturbs tokens independently in each dimension [198].

Example: Uniform Kernel An example of such a diffusion matrix is

$$\boldsymbol{Q}_t^{\mathrm{fw}} = \dot{\sigma}_t \boldsymbol{Q}^{\mathrm{uniform}} = \dot{\sigma}_t (\mathbf{1}\mathbf{1}^T / N - \boldsymbol{I}), \tag{7.19}$$

where $\sigma_t \equiv \sigma(t)$ is a predefined noise schedule with $\sigma(0) = 0$ and $\sigma(T) = \sigma_{\max}$. This uniform kernel transfers any distribution to a uniform distribution as $\sigma \to \infty$. Moreover,

$$p_t = \exp\left(\int_0^t \boldsymbol{Q}_\tau^{\mathrm{fw}} \mathrm{d}\tau\right) p_0 = \exp(\sigma_t \boldsymbol{Q}^{\mathrm{uniform}}) p_0 = \left[e^{-\sigma_t} \boldsymbol{I} + (1 - e^{-\sigma_t}) \frac{\mathbf{1}\mathbf{1}^T}{N}\right] p_0.$$

When $\boldsymbol{x} \in \mathcal{X}^n$ is a discrete object of n dimensions, we have $p_t(\boldsymbol{x}_t \mid \boldsymbol{x}_0) \propto \beta_t^{d(\boldsymbol{x}_t, \boldsymbol{x}_0)} (1 - \beta_t)^{n - d(\boldsymbol{x}_t, \boldsymbol{x}_0)}$, where $d(\cdot, \cdot)$ is the Hamming distance between two sequences and $\beta_t = \frac{N-1}{N}(1 - e^{-\sigma_t})$.

7.6.2 Instantiation of Algorithm 2 for Discrete-Space Inverse Problems

7.6.2.1 A Generalized Split Gibbs Sampler

Recall that our goal is to sample from the posterior distribution

$$p(\boldsymbol{x} \mid \boldsymbol{y}) \propto p(\boldsymbol{y} \mid \boldsymbol{x}) p(\boldsymbol{x}) = \exp(-f(\boldsymbol{x}; \boldsymbol{y}) - g(\boldsymbol{x})), \tag{7.20}$$

where $f(\boldsymbol{x}; \boldsymbol{y}) = -\log p(\boldsymbol{y} \mid \boldsymbol{x})$ and $g(\boldsymbol{x}) = -\log p(\boldsymbol{x})$. The Split Gibbs Sampler (SGS) relaxes this sampling problem by introducing an auxiliary variable $\boldsymbol{z}$, allowing sampling from an augmented distribution

$$\pi(\boldsymbol{x}, \boldsymbol{z}; \eta) \propto \exp(-f(\boldsymbol{z}; \boldsymbol{y}) - g(\boldsymbol{x}) - D(\boldsymbol{x}, \boldsymbol{z}; \eta)), \tag{7.21}$$

where $D(\boldsymbol{x}, \boldsymbol{z}; \eta)$ measures the distance between $\boldsymbol{x}$ and $\boldsymbol{z}$, and $\eta > 0$ is a parameter that controls the strength of regularization.

While PnP-DM (Chapter 6) and other prior works [69, 326] consider $D(\boldsymbol{x}, \boldsymbol{z}; \eta) = \frac{\|\boldsymbol{x} - \boldsymbol{z}\|_2^2}{2\eta^2}$, we generalize the potential function to any continuous function D, such that

$D(\boldsymbol{x}, z; \eta) \to \infty$ as $\eta \to 0$ for any $\boldsymbol{x} \neq z$. As shown in Appendix D.4.1.3, this ensures that both marginal distributions,

$$\pi^X(\boldsymbol{x}; \eta) := \int \pi(\boldsymbol{x}, z; \eta) \mathrm{d}z, \text{ and } \pi^Z(z; \eta) := \int \pi(\boldsymbol{x}, z; \eta) \mathrm{d}\boldsymbol{x}$$

converge to the posterior $p(\boldsymbol{x} \mid \boldsymbol{y})$. The decoupling of the prior and likelihood in Equation (7.21) enables Gibbs sampling, which alternates between two steps:

1. **Likelihood step:** sample $z^{(k)} \sim \pi(\boldsymbol{x} = \boldsymbol{x}^{(k)}, z; \eta) \propto \exp(-f(z; \boldsymbol{y}) - D(\boldsymbol{x}^{(k)}, z; \eta))$
2. **Prior step:** sample $\boldsymbol{x}^{(k+1)} \sim \pi(\boldsymbol{x}, z = z^{(k)}; \eta) \propto \exp(-g(\boldsymbol{x}) - D(\boldsymbol{x}, z^{(k)}; \eta))$

A key feature of SGS is that it does not rely on the gradient of the guidance term $f(z; \boldsymbol{y})$, which is highly desirable in our setting with discrete data. The key challenge that we tackle is to develop an effective approach for discrete-space DMs that is easy to implement and enjoys rigorous guarantees (i.e., sampling from the correct posterior distribution).

7.6.2.2 Prior Step with Discrete Diffusion Models

Suppose $p(\boldsymbol{x})$ is a discrete-state distribution over $\mathcal{X}^n$ modeled by a diffusion process. We consider a discrete DM with uniform transition kernel $\boldsymbol{Q}_t^{\mathrm{fw}} = \frac{1}{N}\mathbf{1}\mathbf{1}^T - \boldsymbol{I}$. To connect SGS to discrete DMs, we specify the potential function $D(\boldsymbol{x}, z; \eta)$ as

$$D(\boldsymbol{x}, z; \eta) := d(\boldsymbol{x}, z) \log \frac{1 + (N-1)e^{-\eta}}{(N-1)(1 - e^{-\eta})} \tag{7.22}$$

where $d(\boldsymbol{x}, z)$ denotes the Hamming distance between $\boldsymbol{x}$ and z. When $\eta \to 0^+$, the regularization potential $D(\boldsymbol{x}, z; \eta)$ goes to infinity unless $d(\boldsymbol{x}, z) = 0$, ensuring the convergence of marginal distributions to $p(\boldsymbol{x} \mid \boldsymbol{y})$. Given Equation (7.22), the prior step can be written as

$$\boldsymbol{x}^{(k+1)} \sim \pi(\boldsymbol{x}, z = z^{(k)}; \eta) \propto p_0(\boldsymbol{x}) \left(\frac{\widetilde{\beta}}{1 - \widetilde{\beta}}\right)^{d(z^{(k)}, \boldsymbol{x})} \tag{7.23}$$

where $\widetilde{\beta} = \frac{N-1}{N}(1 - e^{-\eta})$. On the other hand, the distribution of clean data $\boldsymbol{x}_0$ conditioned on $\boldsymbol{x}_t$ for discrete DMs is given by

$$p(\boldsymbol{x}_0 \mid \boldsymbol{x}_t) \propto p_0(\boldsymbol{x}_0) p(\boldsymbol{x}_t \mid \boldsymbol{x}_0) \propto p_0(\boldsymbol{x}_0) \beta_t^{d(\boldsymbol{x}_t, \boldsymbol{x}_0)} (1 - \beta_t)^{n - d(\boldsymbol{x}_t, \boldsymbol{x}_0)} \tag{7.24}$$

where $\beta_t = \frac{N-1}{N}(1 - e^{-\sigma_t})$. Note that the term $(1 - \beta_t)^n$ does not depend on $\boldsymbol{x}_0$ and can thus be dropped. Therefore, sampling from Equation (7.23) is equivalent

to unconditional generation from $p(\boldsymbol{x}_0 \mid \boldsymbol{x}_t)$ when $\widetilde{\beta} = \beta_t$, i.e., $\eta = \sigma_t$. We can then solve the prior sampling problem by simulating a partial discrete diffusion sampler that starts from $\sigma_t = \eta$, $\boldsymbol{x}_t = \boldsymbol{z}^{(k)}$ and solves backward to $t = 0$. This is analogous to the continuous-space setting in Chapter 6 (i.e., $D(\boldsymbol{x}, \boldsymbol{z}; \eta) = \frac{\|\boldsymbol{x}-\boldsymbol{z}\|_2^2}{2\eta^2}$), where the prior step can be formulated as a Gaussian denoising problem solvable by a continuous-space DM.

7.6.2.3 Likelihood Step with Metropolis-Hastings

With the potential function $D(\boldsymbol{x}, \boldsymbol{z}; \eta)$ specified in Section 7.6.2.2, at iteration k, the likelihood sampling step can be written as:

$$\boldsymbol{z}^{(k)} \sim \pi(\boldsymbol{x} = \boldsymbol{x}^{(k)}, \boldsymbol{z}; \eta) \propto \exp\left(-f(\boldsymbol{z}; \boldsymbol{y}) - d(\boldsymbol{x}, \boldsymbol{z}) \log \frac{1 + (N-1)e^{-\eta}}{(N-1)(1-e^{-\eta})}\right). \quad (7.25)$$

Since the unnormalized probability density function of $\pi(\boldsymbol{x} = \boldsymbol{x}^{(k)}, \boldsymbol{z}; \eta)$ is available, we can efficiently sample from Equation (7.25) using the Metropolis-Hastings algorithm [125, 220].

7.6.2.4 Overall Algorithm

Algorithm 4 **S**plit **G**ibbs **D**iscrete **D**iffusion Posterior Sampling (SGDD)

Require: initialization $\boldsymbol{x}^{(0)} \in \mathcal{X}^N$, total number of iterations $K > 0$, noise schedule $\{\eta_k\}_{k=0}^{K-1}$, likelihood potential $f(\,\cdot\,; \boldsymbol{y})$ with measurements $\boldsymbol{y} \in \mathbb{R}^m$, pre-trained discrete diffusion model $\boldsymbol{s}_{\boldsymbol{\theta}}$.

1: **for** $k = 0, \ldots, K-1$ **do**
2: $\quad \boldsymbol{z}^{(k)} \leftarrow \texttt{LikelihoodStep}(\boldsymbol{x}^{(k)}, \boldsymbol{y}, \eta_k)$ ▷ Sample (7.25) (Section 7.6.2.3)
3: $\quad \boldsymbol{x}^{(k+1)} \leftarrow \texttt{PriorStep}(\boldsymbol{z}^{(k)}, \boldsymbol{s}_{\boldsymbol{\theta}}, \eta_k)$ ▷ Sample (7.23) (Section 7.6.2.2)
4: **end for**
5: **return** $\boldsymbol{x}^{(K)}$

We now summarize the complete SGDD algorithm. Like PnP-DM, SGDD alternates between a likelihood step and a prior step while employing an annealing schedule $\{\eta_k\}$, which starts at a large η_0 and gradually decays to $\eta_k \to 0$. This annealing scheme accelerates the mixing time of the Markov chain and ensures the convergence of $\pi^X(\boldsymbol{x}; \eta)$ and $\pi^Z(\boldsymbol{z}; \eta)$ to $p(\boldsymbol{x} \mid \boldsymbol{y})$ as $\eta \to 0$. We present the complete pseudocode of our method in Algorithm 4.

7.6.3 Convergence Analysis

We provide theoretical guarantees on the convergence of SGDD. For two probability measures in a finite domain $\mathcal{X}$, we consider the *Kullback-Leibler (KL) divergence*

and the *Fisher divergence (or relative Fisher information)*, respectively, as (where we define $f := \mu/\pi$)

$$\mathsf{KL}(\mu\|\pi) := \mathbb{E}_\mu\left[\log\frac{\mu}{\pi}\right],$$
$$\mathsf{FI}_{\boldsymbol{Q}}(\mu\|\pi) := \sum_{\boldsymbol{x}_i,\boldsymbol{x}_j\in\mathcal{X}} \pi(\boldsymbol{x}_i)\boldsymbol{Q}^{[j,i]}\left(f(\boldsymbol{x}_j) - f(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)\log\frac{f(\boldsymbol{x}_j)}{f(\boldsymbol{x}_i)}\right).$$

Both divergences are nonnegative and equal to zero if and only if $\mu = \pi$, when $\boldsymbol{Q}$ is irreducible. For the continuous case in Section 6.4 and Section 7.5.2, Fisher divergence can be written as a quadratic form $\mathsf{FI}(\mu\|\pi) = \mathbb{E}_\mu\|\nabla\log(\mu/\pi)\|^2$. However, it does not have an analogous quadratic form in finite spaces, which poses additional challenges to the analysis of SGDD. To address this challenge, we adopt a generalized definition of Fisher divergence [27, 130] and analyze the convergence of SGDD to the stationary distribution using a more general technique that encompasses both continuous and discrete settings.

We define a distribution μ_τ over $\mathcal{X}$ that evolves according to likelihood steps and prior steps alternatively where τ is the index for time over K iterations of SGDD. We assume each likelihood step is implemented with the Metropolis-Hastings algorithm, and that each prior step is solved by the Euler method with an approximated score function. We compare μ_τ to the continuous-time stationary distribution π_τ, which alternates between π^X and π^Z. The definitions of μ_τ and π_τ are provided in Appendix D.4.1.

Theorem 7.6.1. *Consider running K iterations of SGDD with a fixed $\eta > 0$ and an estimated concrete score $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t; t)$, and suppose that each prior step is solved by an H step Euler method. Let $t^* > 0$ with $\sigma(t^*) = \eta$. Let $T_k = k(t^*+1)+1$ be the starting time of the k-th prior step. Define π_τ and μ_τ as stationary and non-stationary distributions. Over K iterations of SGDD, the average Fisher divergence between μ_τ and π_τ satisfies*

$$\frac{1}{K}\sum_{k=0}^{K-1}\underbrace{\frac{1}{t^*}\int_{T_k}^{T_k+t^*}\mathsf{FI}_{\boldsymbol{Q}_\tau}(\mu_\tau\|\pi_\tau)\mathrm{d}\tau}_{\substack{\textit{average Fisher divergence}\\ \textit{over the k-th prior step}}} \le \underbrace{\frac{2\mathsf{KL}(\mu_0\|\pi_0)}{Kt^*}}_{\substack{\textit{convergence}\\ \textit{from initialization}}} + \underbrace{\frac{4M\epsilon}{c}}_{\textit{score error}} + \underbrace{\frac{2MLt^*}{cH}}_{\textit{discretization error}}. \tag{7.26}$$

where $\|\frac{\boldsymbol{s}_{\boldsymbol{\theta}}(\cdot;t)-\boldsymbol{s}(\cdot;t)}{\boldsymbol{s}(\cdot;t)}\|_\infty \le \epsilon < 1$, and L, M, c are positive constants defined in Appendix D.4.1.

The proof is provided in Appendix D.4.1. Theorem 7.6.1 states that the average Fisher divergence of the non-stationary process with respect to the stationary process in all prior steps converges at the rate of $O(1/K)$, up to a constant error term. This result extends our theoretical understanding of diffusion posterior sampling by generalizing the analysis in [273] and Section 6.4 on SDEs to general Markov processes using the free-energy-rate-functional-relative-Fisher-information (FIR) inequality [129]. Moreover, compared to existing analyses for continuous diffusion models, our analysis accounts not only for the imperfect score function but also for the discretization error in solving continuous-time Markov chains as well.

7.6.4 Experiments

We apply SGDD to inverse problems on discrete images to show its effectiveness on posterior estimation in discrete domains. Implementation details for the experiments are provided in Appendix D.4.2.3.

Problem Setup We convert the MNIST dataset [172] to binary strings by discretizing the images, and train a discrete diffusion prior on 60k training data using the SEDD [198] model with the uniform transition kernel. We consider AND and XOR operators as examples of linear and nonlinear forward models on binary strings. We randomly pick γn pairs of positions (i_p, j_p) over $\{1, \dots, n\}$, and compute

$$\mathcal{A}_{\text{AND}}(\boldsymbol{x}) = [\boldsymbol{x}_{i_p} \wedge \boldsymbol{x}_{j_p}]_{p=1,\dots,\gamma n},\ \mathcal{A}_{\text{XOR}}(\boldsymbol{x}) = [\boldsymbol{x}_{i_p} \oplus \boldsymbol{x}_{j_p}]_{p=1,\dots,\gamma n}. \tag{7.27}$$

The likelihood function is defined as $p(\boldsymbol{y} \mid \boldsymbol{x}) \propto \exp(-\|\mathcal{A}(\boldsymbol{x}) - \boldsymbol{y}\|_0/\sigma_{\boldsymbol{y}})$.

Baselines We compare SGDD to existing approaches that apply to discrete diffusion posterior sampling: DPS [64], SVDD-PM [182], and SMC [318]. Details on how we adapt these methods to our setting are provided in Appendix D.4.2.2.

Evaluation Metrics We use 1,000 binary images from the test set of MNIST and calculate the Peak Signal-to-Noise Ratio (PSNR) of the reconstructed image. Furthermore, we train a simple convolutional neural network on MNIST as a surrogate and report the *classifier accuracy* of the generated samples.

Results As shown in Table 7.4, SGDD outperforms baseline methods by a large margin in both XOR and AND tasks. We present the samples generated by SGDD for the XOR task in Figure 7.14. The reconstructed samples are visually consistent

Table 7.4: **Quantitative results for XOR and AND problems on discretized MNIST.** We report the mean and standard deviation (shown in parentheses) of *PSNR* and *class accuracy* across 1,000 generated samples. SGDD demonstrates superior performance on both tasks.

	XOR		AND	
	PSNR ↑	Accuracy (%) ↑	PSNR ↑	Accuracy (%) ↑
SVDD-PM [182]	11.81 (2.54)	51.4	10.04 (1.49)	33.7
SMC [318]	10.05 (1.54)	27.8	10.25 (1.63)	24.4
DPS [64]	9.04 (1.21)	30.0	8.67 (0.91)	24.5
SGDD	**20.17** (3.47)	**91.2**	**17.25** (3.82)	**79.4**

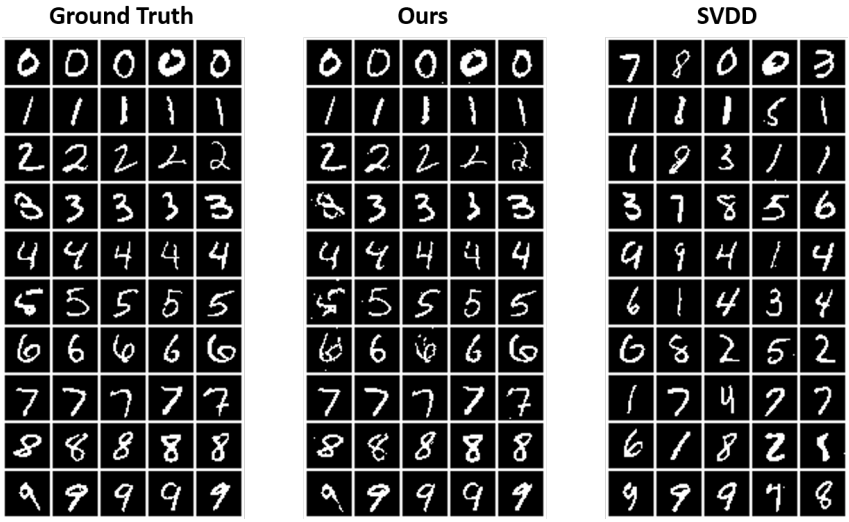


Figure 7.14: **Sampling results of the XOR task on the discretized MNIST dataset.** SGDD faithfully recovers the structural information of the ground truth signal.

with the underlying ground truth signal, which explains the high class accuracy of SGDD shown in Table 7.4. Furthermore, we demonstrate that SGDD generates diversified samples when the measurement $\boldsymbol{y}$ is sparse. For example, when a digit is masked with a large box, as shown in Figure 7.15, the measurement lacks sufficient information to fully recover the original digit. In this scenario, SGDD generate samples from multiple plausible modes, including digits 1, 4, 7, and 9. This highlights the ability of SGDD to produce diverse samples from the posterior distribution while preserving consistency with the measurement.

7.7 Conclusion

In this chapter, we presented four instantiations of Algorithm 2, each designed to extend the applicability of diffusion-based posterior estimation to broader classes

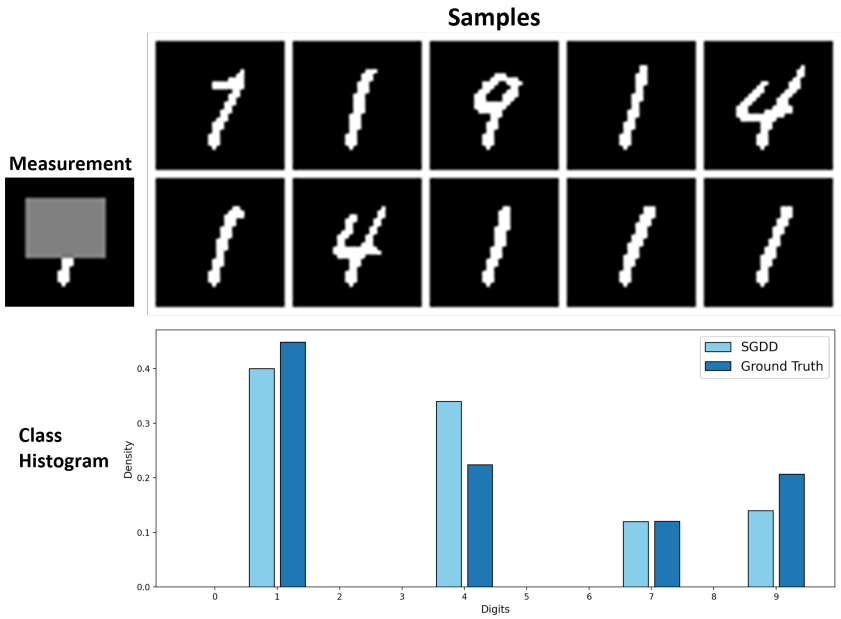


Figure 7.15: **Diversified samples when the measurement y is sparse.** Samples are generated by SGDD when solving an MNIST inpainting task.

of inverse problems. These methods address key limitations of the original PnP-DM method in Chapter 6 by incorporating richer forms of prior information, scaling to higher-dimensional video data, accommodating non-differentiable forward models, and enabling inference in discrete domains. Despite targeting distinct challenges, all four instantiations share a common alternating-update structure that provides a modular template for future extensions. Collectively, these developments demonstrate the versatility of our unified PnPDP framework, which we believe paves the way for future advances in diffusion-based posterior estimation.

Chapter 8

INVERSEBENCH: BENCHMARKING DIFFUSION-BASED METHODS FOR SCIENTIFIC INVERSE PROBLEMS

Plug-and-play diffusion priors (PnPDP), including the methods we presented in Chapter 6 and Chapter 7, have emerged as a promising paradigm for solving inverse problems. While many existing methods claim to handle general inverse problems, they were investigated in different domains (primarily natural image restoration). They have never been compared in a controlled manner, and it is unclear how they perform on scientific inverse problems in computational imaging. To address this gap, we introduce InverseBench, a framework that evaluates diffusion models across five distinct scientific inverse problems. These problems present unique structural challenges that differ from existing benchmarks, arising from critical scientific applications such as optical tomography, medical imaging, black hole imaging, seismology, and fluid dynamics. With InverseBench, we benchmark 14 inverse problem algorithms that use plug-and-play diffusion priors against strong, domain-specific baselines, offering valuable new insights into the strengths and weaknesses of existing algorithms. To facilitate further research and development, we open-source the codebase, along with datasets and pre-trained models, at `https://devzhk.github.io/InverseBench/`.

This chapter is based on our work [357], published as a Spotlight paper in the *Proceedings of the 13th International Conference on Learning Representations 2025 (ICLR 2025)*. The appendix for this chapter is Appendix E. The code for the work presented in this chapter is available at `https://github.com/devzhk/InverseBench`.

8.1 Introduction

The existing PnPDP methods are primarily evaluated and compared on a fairly narrow set of image restoration tasks—such as inpainting, super-resolution, and deblurring [44, 64, 147, 214, 262, 306]. These problems differ greatly from those in science and engineering applications such as geophysics [294], astronomy [233], oceanography [47], and many other fields, which have very different structural challenges arising from the underlying physics. It is unclear how much insight can be carried over from image restoration to scientific inverse problems.

In this chapter, we introduce InverseBench, a comprehensive benchmarking framework designed to evaluate PnPDP methods in a systematic and easily extensible manner. We curate a diverse set of five inverse problems from distinct scientific domains: optical tomography, black hole imaging, medical imaging, seismology, and fluid dynamics. These problems present structural challenges that differ significantly from natural image restoration tasks (cf. Figure 8.1 and Table 8.2), and encompass a broad spectrum of complexities across multiple scientific fields. Most notably, the forward model (which maps the target image to measurements) is defined using various types of physics-based models, which can be highly nonlinear and difficult to evaluate.

We select 14 representative PnPDP algorithms proposed for solving inverse problems, providing a thorough comparison of their performance across different scientific inverse problems and further insights into their efficacy and limitations. Additionally, we establish strong, domain-specific baselines for each inverse problem, providing a meaningful reference point for assessing the effectiveness of diffusion model-based approaches against traditional methods.

Through extensive experiments, we find that PnPDP methods generally exhibit strong performance given a suitable dataset for training a diffusion prior. This performance is consistent even as we vary the forward model (which is a strength of a PnP approach), given appropriate tuning. However, for forward models that require certain constraints on the input (e.g., use a PDE solver), performance can be very sensitive to hyperparameter tuning. Moreover, the strength of using a diffusion prior can also be a limitation, as PnPDP methods have difficulty when the source image is out of the prior distribution (i.e., the use of diffusion models makes it difficult to recover "surprising" results). Additionally, we find that PnP methods that use multiple queries of the forward model tend to outperform simpler methods like DPS, at the cost of requiring additional tuning and computation, which points to an interesting direction for future method development.

InverseBench is implemented as a highly modular framework that can interface with new inverse problems and algorithms to run experiments at scale. We open-source the codebase, along with datasets and pre-trained models, at `https://devzhk.github.io/InverseBench/`.

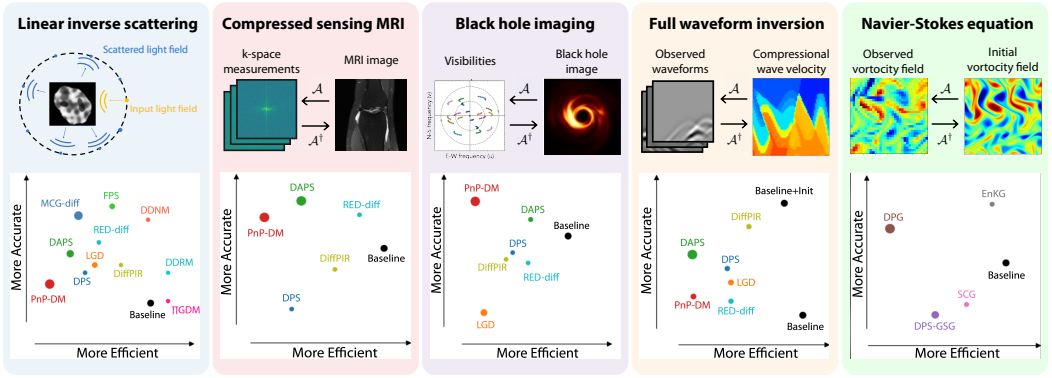


Figure 8.1: **Illustration of five benchmark problems in the InverseBench.** $\mathcal{A}$ represents the forward model that produces measurements from the underlying target. $\mathcal{A}^\dagger$ represents the inverse map. In the linear inverse scattering problem (left two), the measurements are the recorded data from the receivers, and the unknown source we aim to infer is the permittivity map of the object. The bottom panel displays the efficiency and accuracy plots for our benchmarked algorithms. Certain characteristics of the problem cause the efficiency and accuracy trade-offs of each algorithm to vary across tasks. In these plots, the larger radius of the points indicates greater interaction with the forward function $\mathcal{A}$, as measured by the number of forward model evaluations.

Table 8.1: **Requirements on the forward model of the algorithms evaluated in our experiments.**

Category	Method	SVD of $\mathcal{A}$	Pseudo inverse of $\mathcal{A}$	Linear $\mathcal{A}$	Gradient
Linear guidance	DDRM [155]	✓	✓	✓	–
	DDNM [306]	✗	✓	✓	–
	ΠGDM [262]	✗	✓	✗	–
General guidance	DPS [64]	✗	✗	✗	✓
	LGD [263]	✗	✗	✗	✓
	DPG [277]	✗	✗	✗	✗
	SCG [141]	✗	✗	✗	✗
	EnKG [356]	✗	✗	✗	✗
Variable-splitting	DiffPIR [361]	✗	✗	✗	✓
	PnP-DM [320]	✗	✗	✗	✓
	DAPS [341]	✗	✗	✗	✓
Variational Bayes	RED-diff [214]	✗	✗	✗	✓
Sequential Monte Carlo	FPS [89]	✗	✗	✓	–
	MCGdiff [44]	✓	✓	✓	–

8.2 Plug-and-Play Diffusion Priors for Inverse Problems

We use the term *Plug-and-Play Diffusion Prior* (PnPDP) to refer to the class of recent methods that use diffusion models (or the denoising network within) as plug-and-play priors [291] for solving inverse problems. See Section 5.3.2 for a brief overview. Table 8.1 lists the 14 representative PnPDP methods we selected and notes

their different requirements on the forward model $\mathcal{A}$. Broadly speaking, existing PnPDP approaches can be grouped into four categories described below.

Guidance-Based Methods Arguably the most popular approach to solving inverse problems with a pre-trained diffusion model is guidance-based methods [64, 155, 251, 262, 306], which modify Equation (5.8) by adding a likelihood score term, $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y} \mid \boldsymbol{x}_t)$, along the diffusion trajectory using Equation (5.10). This term is related to the forward model $\mathcal{A}$ if the final clean $\boldsymbol{x}_0$ is a candidate source, in which case $p(\boldsymbol{y} \mid \boldsymbol{x}_0)$ can be estimated by querying $\mathcal{A}$. However, as we discussed, $\log p_t(\boldsymbol{y} \mid \boldsymbol{x}_t)$ is generally intractable so various approximations have been proposed [34, 64, 262, 265].

Variable Splitting Variable splitting is a widely used strategy for solving regularized optimization problems and conducting Bayesian inference [176, 296]. The core idea is to split the inference into two alternating steps. The first step uses the forward model to update or sample in the neighborhood of the most recent $\boldsymbol{x}_t$. The second step runs unconditional inference on $p(\boldsymbol{x}_t)$, which amounts to running Equation (5.8) for a small amount of time. Example methods include those presented in Chapter 6, Chapter 7, and others in the literature [181, 260, 326, 341, 361].

Variational Bayes Variational Bayes methods approximate intractable distributions such as $p(\boldsymbol{x} \mid \boldsymbol{y})$ using some simpler parameterized distribution q_θ [342]. The key idea is to find a q_{θ^*} that both fits the measurements $\boldsymbol{y}$ and agrees with the prior $p(\boldsymbol{x})$ in a KL divergence sense. Instead of directly sampling according to Equation (5.8), it uses the diffusion model as a prior within a variational inference framework [101, 103, 214].

Sequential Monte Carlo Sequential Monte Carlo (SMC) methods draw samples iteratively from a sequence of probability distributions. These methods represent probability distributions by a set of particles with associated weights, which asymptotically converge to a target distribution following a sequence of proposal and reweighting steps. Recent works have extended SMC methods to the sequential diffusion sampling process [44, 89, 283, 318], enabling zero-shot posterior sampling with diffusion priors. However, these methods are typically applicable only to inverse problems with linear forward models.

Table 8.2: **Characteristics of different inverse problems in InverseBench.** From left to right: whether the forward model is linear, whether one can efficiently compute the SVD from the forward model, the domain in which the inverse problem is defined, whether the forward model can be solved in closed form, whether one can access gradients from the forward model, and the noise type.

Problem	Linear $\mathcal{A}$	Efficient SVD of $\mathcal{A}$	Domain	Closed-form $\mathcal{A}$	Gradient of $\mathcal{A}$	Noise type
Linear inverse scattering	✓	✓	$\mathbb{C}^n$	✓	✓	Gaussian
Compressed sensing MRI	✓	✗	$\mathbb{C}^n$	✓	✓	Real-world
Black hole imaging	✗	✗	$\mathbb{R}^n$	✓	✓	Non-additive
Full waveform inversion	✗	✗	$\mathbb{R}^n$	✗	✓	Noise-free
Navier-Stokes equation	✗	✗	$\mathbb{R}^n$	✗	✗	Gaussian

8.3 InverseBench

In this section, we introduce the formulation and specific challenges of the five scientific inverse problems considered in InverseBench: linear inverse scattering, compressed sensing MRI, black hole imaging, full waveform inversion, and the Navier-Stokes equation. The characteristics of these inverse problems are summarized in Table 8.2. Their computational characteristics are summarized in Figure E.1. Detailed descriptions and formal definitions can be found in Appendix E.2.

Linear Inverse Scattering Inverse scattering is an inverse problem that arises from optical microscopy, where the goal is to recover the unknown permittivity contrast $\boldsymbol{x}_0 \in \mathbb{R}^n$ from the measured scattered lightfield $\boldsymbol{y}_{\text{sc}} \in \mathbb{C}^m$. We consider the following formulation of inverse scattering

$$\boldsymbol{y}_{\text{sc}} = \boldsymbol{H}(\boldsymbol{f}_{\text{tot}} \odot \boldsymbol{x}_0) + \boldsymbol{n} \in \mathbb{C}^m \quad \text{where} \quad \boldsymbol{f}_{\text{tot}} = \boldsymbol{G}(\boldsymbol{f}_{\text{in}} \odot \boldsymbol{x}_0). \tag{8.1}$$

Here $\boldsymbol{G} \in \mathbb{C}^{n\times n}$ and $\boldsymbol{H} \in \mathbb{C}^{m\times n}$ are the discretized Green's functions that model the responses of the optical system, $\boldsymbol{f}_{\text{in}}$ and $\boldsymbol{f}_{\text{tot}}$ are the input and total lightfields, $\odot$ is the elementwise product, and $\boldsymbol{n}$ is the measurement noise. Since this problem is a linearized version of the general nonlinear inverse scattering problem based on the first Born approximation, we refer to it as linear inverse scattering. This problem allows us to test algorithms designed specifically for linear problems.

Compressed Sensing MRI Compressed sensing MRI is a technique that accelerates the scan time of MRI via subsampling. We consider the parallel imaging (PI) setup of CS-MRI (same as Section 2.2), which is widely adopted in research and practice. Mathematically, PI CS-MRI can be formulated as an inverse problem that aims to recover an image $\boldsymbol{z} \in \mathbb{C}^n$ from

$$\boldsymbol{y}_j = \boldsymbol{M}\boldsymbol{F}\boldsymbol{S}_j\boldsymbol{x}_0 + \boldsymbol{n}_j \in \mathbb{C}^m \quad \text{for } j = 1, ..., J$$

where $\boldsymbol{M} \in \{0,1\}^{m \times n}$ is a subsampling operator and $\boldsymbol{F}$ is the Fourier transform; $\boldsymbol{y}_j$, $\boldsymbol{S}_j$, and $\boldsymbol{n}_j$ are the measurements, sensitivity map, and the noise of the j-th coil, respectively. Compressed sensing MRI is a linear problem, but it poses significant challenges due to its high-dimensional nature, involvement of priors in the complex domain, and attention to fine-grained details.

Black Hole Imaging Black hole imaging (BHI) is a technique of imaging black holes using Very Long Baseline Interferometry (VLBI). Each measurement for BHI, $\boldsymbol{V}^t_{\{a,b\}} \in \mathbb{C}$ (often referred to as *visibility*) is given by cross-correlating the recorded scalar electric fields of a pair of telescopes $\{a, b\}$ at time t and measures a Fourier component of the target image $\boldsymbol{x}_0$. However, the measurements are corrupted by the gain errors and phase errors due to the atmosphere. To correct for these errors, multiple noisy visibilities can be combined into quantities that are invariant to these errors, which are called *closure phase* and *log closure amplitude* measurements [23, 50]

$$\begin{aligned}
\boldsymbol{y}^t_{\text{cp},\{a,b,c\}} &:= \angle(\boldsymbol{V}^t_{\{a,b\}} \boldsymbol{V}^t_{\{b,c\}} \boldsymbol{V}^t_{\{a,c\}}) := \mathcal{A}^t_{\text{cp},\{a,b,c\}}(\boldsymbol{x}_0) \in \mathbb{R}, \\
\boldsymbol{y}^t_{\text{logca},\{a,b,c,d\}} &:= \log\left(\frac{\left|\boldsymbol{V}^t_{\{a,b\}}\right|\left|\boldsymbol{V}^t_{\{c,d\}}\right|}{\left|\boldsymbol{V}^t_{\{a,c\}}\right|\left|\boldsymbol{V}^t_{\{b,d\}}\right|}\right) := \mathcal{A}^t_{\text{logca},\{a,b,c,d\}}(\boldsymbol{x}_0) \in \mathbb{R},
\end{aligned}$$

where $\angle$ computes the angle of a complex number. Additionally, because the closure quantities do not constrain the total flux (i.e., summation of the pixel values) of the underlying black hole image, we add a constraint on the total flux defined as

$$\boldsymbol{y}_{\text{flux}} := \int_\rho \int_\delta \boldsymbol{x}_0(\rho, \delta) \mathrm{d}\rho \mathrm{d}\delta \in \mathbb{R}. \tag{8.2}$$

Aggregating data over time intervals and telescope combinations, the overall forward model of BHI can be expressed as

$$\boldsymbol{y} := \left[\mathcal{A}_{\text{cp}}(\boldsymbol{x}_0), \mathcal{A}_{\text{logca}}(\boldsymbol{x}_0), \mathcal{A}_{\text{flux}}(\boldsymbol{x}_0)\right] := \left[\boldsymbol{y}_{\text{cp}}, \boldsymbol{y}_{\text{logca}}, \boldsymbol{y}_{\text{flux}}\right], \tag{8.3}$$

where $\boldsymbol{y}_{\text{cp}} = \left[\boldsymbol{y}^t_{\text{cp},\{a,b,c\}}, \forall t \in \mathcal{T}, \{a,b,c\}\right]$ is the set of all closure phase measurements and $\boldsymbol{y}_{\text{cp}} = \left[\boldsymbol{y}^t_{\text{logca},\{a,b,c,d\}}, \forall t \in \mathcal{T}, \{a,b,c,d\}\right]$ is the set of all log closure amplitude measurements over the observation period $\mathcal{T}$ and combinations of telescopes. Since the closure quantities are nonlinear transformations of the visibilities, the forward model of BHI is non-convex. The inverse problem is further complicated by the need for super-resolution imaging beyond the intrinsic resolution of

the observations (i.e., maximum probed spatial frequency), as well as phase ambiguities, which can lead to multiple modes in the posterior distribution [269, 273]. Another challenge of BHI is that measurement noise is non-additive due to the usage of the closure quantities. We refer readers to Appendix C.2 for more details on the BHI setup.

Full Waveform Inversion Full waveform inversion (FWI) aims to infer subsurface physical properties—such as compressional and shear wave velocities—using the full information of recorded waveforms. We consider the problem of recovering the discrete wave velocity $\boldsymbol{x}_0 \in \mathbb{R}^n$, or equivalently, the square slowness of the wave $\boldsymbol{m} := \frac{1}{\boldsymbol{x}_0^2}$, from measurements of the pressure wavefield. The measurements $\boldsymbol{y}$ are given by

$$\boldsymbol{y} = \boldsymbol{P}_r \boldsymbol{g} = \boldsymbol{P}_r \boldsymbol{A}(\boldsymbol{m})^{-1} \boldsymbol{P}_s^T \boldsymbol{q}_s = \boldsymbol{P}_r \boldsymbol{A}\left(\frac{1}{\boldsymbol{x}_0^2}\right)^{-1} \boldsymbol{P}_s^T \boldsymbol{q}_s \in \mathbb{R}^m, \tag{8.4}$$

where $\boldsymbol{P}_r$ is the sampling operator at the receiver locations, $\boldsymbol{P}_s^T$ is the injection operator at the source locations, sampling operator at the receiver locations, $\boldsymbol{g}$ is the discretized synthetic pressure wavefield (which is a function of location and time), and $\boldsymbol{q}_s$ is the corresponding pressure source. Matrix $\boldsymbol{A}\left(\frac{1}{\boldsymbol{x}_0^2}\right)$ is the discretized operator for the acoustic (scalar) wave equation that models seismic wave propagation in heterogeneous acoustic media with constant density

$$\frac{1}{\boldsymbol{x}_0^2} \partial_t^2 g - \nabla_{\boldsymbol{x}}^2 g = q_s, \tag{8.5}$$

where v, $g := g(\boldsymbol{x}, t)$, and q_s are the continuous counterpart of $\boldsymbol{x}_0$, $\boldsymbol{g}$, and $\boldsymbol{q}_s$. So, $\boldsymbol{A}\left(\frac{1}{\boldsymbol{x}_0^2}\right)$ is the discretization of operator $\frac{1}{\boldsymbol{x}_0^2}\partial_t^2 - \nabla_{\boldsymbol{x}}^2$, where $\nabla_{\boldsymbol{x}}^2$ is the Laplacian operator. Since we only have measurements at the free surface, the inverse problem has non-unique solutions. One of the major challenges for FWI is the prohibitive computational expense, especially for large problems, as it usually requires numerous calls to the forward modeling process. Moreover, the conventional method for FWI, called the adjoint-state method, casts it as a local optimization problem [293, 294]. This means that a sufficiently accurate initial model is required, as the solution is only sought in its vicinity. FWI conventionally needs to start with a smoothed model derived from simpler ray-based methods [194, 211], which imposes a significantly strong prior. A general method with less reliance on initialization is highly desired.

Navier-Stokes Equation Navier-Stokes equation is a classic benchmarking problem from fluid dynamics [143]. Its applications range from ocean dynamics to

climate modeling, where measurements of the atmosphere are used to calibrate the initial condition for the downstream numerical forecasting. We consider the forward model that is given by the following 2D Navier-Stokes equation for a viscous, incompressible fluid in vorticity form on a torus.

$$\begin{aligned} \partial_t \omega(\boldsymbol{x}, t) + \boldsymbol{v}(\boldsymbol{x}, t) \cdot \nabla_{\boldsymbol{x}} \omega(\boldsymbol{x}, t) &= \nu \nabla_{\boldsymbol{x}}^2 \omega(\boldsymbol{x}, t) + f(\boldsymbol{x}), && \boldsymbol{x} \in (0, 2\pi)^2, t \in (0, T] \\ \nabla_{\boldsymbol{x}} \cdot \boldsymbol{v}(\boldsymbol{x}, t) &= 0, && \boldsymbol{x} \in (0, 2\pi)^2, t \in [0, T] \\ \omega(\boldsymbol{x}, 0) &= \omega_0(\boldsymbol{x}), && \boldsymbol{x} \in (0, 2\pi)^2 \end{aligned} \tag{8.6}$$

where $\boldsymbol{v} \in C\left([0, T]; H^r_{\text{per}}((0, 2\pi)^2; \mathbb{R}^2)\right)$ for any $r > 0$ is the velocity field, $\omega = \nabla_{\boldsymbol{x}} \times \boldsymbol{v}$ is the vorticity, $\omega_0 \in L^2_{\text{per}}\left((0, 2\pi)^2; \mathbb{R}\right)$ is the initial vorticity, $\nu \in \mathbb{R}_+$ is the viscosity coefficient, and $f \in L^2_{\text{per}}\left((0, 2\pi)^2; \mathbb{R}\right)$ is the forcing function. The solution operator $\mathcal{G}$ is defined as the operator mapping the vorticity from the initial vorticity to the vorticity at time T, i.e., $\mathcal{G} : \omega_0 \to \omega_T$. We consider the problem of recovering the initial vorticity field $\boldsymbol{x}_0 := \omega_0$ from the noisy partial measurements $\boldsymbol{y}$ of the vorticity field $\boldsymbol{w}_T$ at time T given by

$$\boldsymbol{y} = \boldsymbol{P}\boldsymbol{G}(\boldsymbol{x}_0) + \boldsymbol{n}$$

where $\boldsymbol{P}$ is the sampling operator, $\boldsymbol{n}$ is the measurement noise, and $\boldsymbol{G}(\cdot)$ is the discretization of $\mathcal{G}$. The Navier-Stokes equation does not admit a closed-form solution, so there is no closed-form gradient available for the solution operator. We implement the forward model using a pseudo-spectral solver with adaptive time stepping [127]. Obtaining an accurate numerical gradient via automatic differentiation through the numerical solver is challenging due to the extensive computation graph expanded after thousands of discrete time steps.

8.4 Experiments

8.4.1 Experimental Setup

Here we provide a summary of our experimental setup. The details about each inverse problem and its corresponding datasets can be found in Appendix E.2. For each problem, we train a diffusion model on the training set using the pipeline from [151], and use the same checkpoint for all PnPDP methods on each problem for a fair comparison. Technical details of DM pre-training can be found in Appendix E.3.

Linear Inverse Scattering We create a dataset of fluorescence microscopy images using the online simulator [314]. The training set consists of 10,000 HL60 nucleus

permittivity images. The test and validation sets contain 100 and 10 permittivity images, respectively. We curate the test and validation samples so that all test samples have less than 0.6 cosine similarities to those in the training set.

Compressed Sensing MRI We use the multi-coil raw k-space data from the fastMRI knee dataset [338]. We exclude the first and last 5 slices of each volume for training and validation, since they do not contain much anatomical information, and resize all images down to 320×320 following the preprocessing procedure of [145]. In total, we use 25,012 images for training, 6 images for hyperparameter search, and 94 images for testing.

Black Hole Imaging We leverage a dataset of General Relativistic MagnetoHydroDynamic (GRMHD) [223] simulated black hole images as our training data. The training set consists of 50,000 resized 64×64 images. Since this dataset is not publicly available, we generate synthetic images from a pre-trained diffusion model for both the validation and test datasets. Specifically, we use 5 sampled images for the validation set and 100 sampled images for the test set.

Full Waveform Inversion We adapt the CurveFaultB dataset [80], which presents the velocity maps that contain faults caused by shifted rock layers. We resize the original data to a resolution of 128×128 with bilinear interpolation and anti-aliasing. The training set consists of 50,000 velocity maps. The test and validation sets contain 10 and 1 velocity maps, respectively.

Navier-Stokes Equation We create a dataset of non-trivial initial vorticity fields by first sampling from a Gaussian random field and then evolving Equation (8.6) for five time units. The equation setup follows Iglesias, Law, and Stuart [143] and Li et al. [185]. We set the Reynolds number to 200 and the spatial resolution to 128×128. The training set consists of 10,000 samples. The test and validation sets contain 10 and 1 samples, respectively.

8.4.2 Evaluation Metrics

Accuracy Metrics We use the Peak Signal-to-Noise Ratio (PSNR) and Structure Similarity Index Measure (SSIM) [310] as the generic ways to quantify recovery of the true source. For all the problems except for black hole imaging, we use the ℓ_2 error $\|\mathcal{A}(\widehat{\boldsymbol{x}}) - \boldsymbol{y}\|_2$ to measure the consistency of a reconstruction $\widehat{\boldsymbol{x}}$ with the measurements $\boldsymbol{y}$. For black hole imaging, the closure quantities are invariant

under translation, and so we measure the best fit under any shift alignment. We also assess the Blur PSNR, where images are blurred to match the target resolution of the telescope. We evaluate data misfit via the χ^2 statistic on two closure quantities: the closure phase (χ^2_{cp}) and the log closure amplitude (χ^2_{logca}). A χ^2 value close to 1 indicates better data fitting. To facilitate a comparison between underfitting ($\chi^2 > 1$) and overfitting ($\chi^2 < 1$), we report a unified metric defined as

$$\tilde{\chi}^2 = \chi^2 \cdot \mathbb{1}\{\chi^2 \geq 1\} + \frac{1}{\chi^2} \cdot \mathbb{1}\{\chi^2 < 1\}. \tag{8.7}$$

For FWI and Navier-Stokes experiments, we also use the relative ℓ_2 error $\|\widehat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2 / \|\boldsymbol{x}_0\|_2$ as it is a commonly used primary accuracy metric in PDE problems [143].

Efficiency Metrics We define a set of efficiency metrics in Table E.7 to evaluate the computational complexity of inverse algorithms more thoroughly. These metrics fall into two categories: (1) total metrics that measure the overall computational cost; (2) sequential metrics that help identify bottlenecks where forward model or diffusion model queries cannot be parallelized.

Ranking Score To assess the relative ranking of different PnP diffusion models across various problems, we define the following ranking score for each problem. Given a set of accuracy or efficiency metrics $\{h_k\}_{k=1}^K$, we rank the algorithms according to each individual metric. Suppose algorithm l has the rank $R_k(l)$ out of L algorithms under the metric k. Its ranking score on this metric is given by $\text{score}_k(l) = 100 \times (L - R_k(l) + 1)\,/L$. For each problem, we calculate the average ranking score to assess overall performance:

$$\text{score}^{\text{problem}}(l) = \frac{1}{K} \sum_{k=1}^{K} \text{score}_k(l).$$

8.4.3 Main Findings

The full experimental results for each problem are provided in Appendix E.1 as tables. Below, we highlight some key insights distilled from these results.

How do PnPDP methods work compared to conventional baselines? Our primary finding is that, given a suitable dataset for training a DM prior, PnPDP methods generally outperform conventional baselines. This is evident in Figure 8.1, where

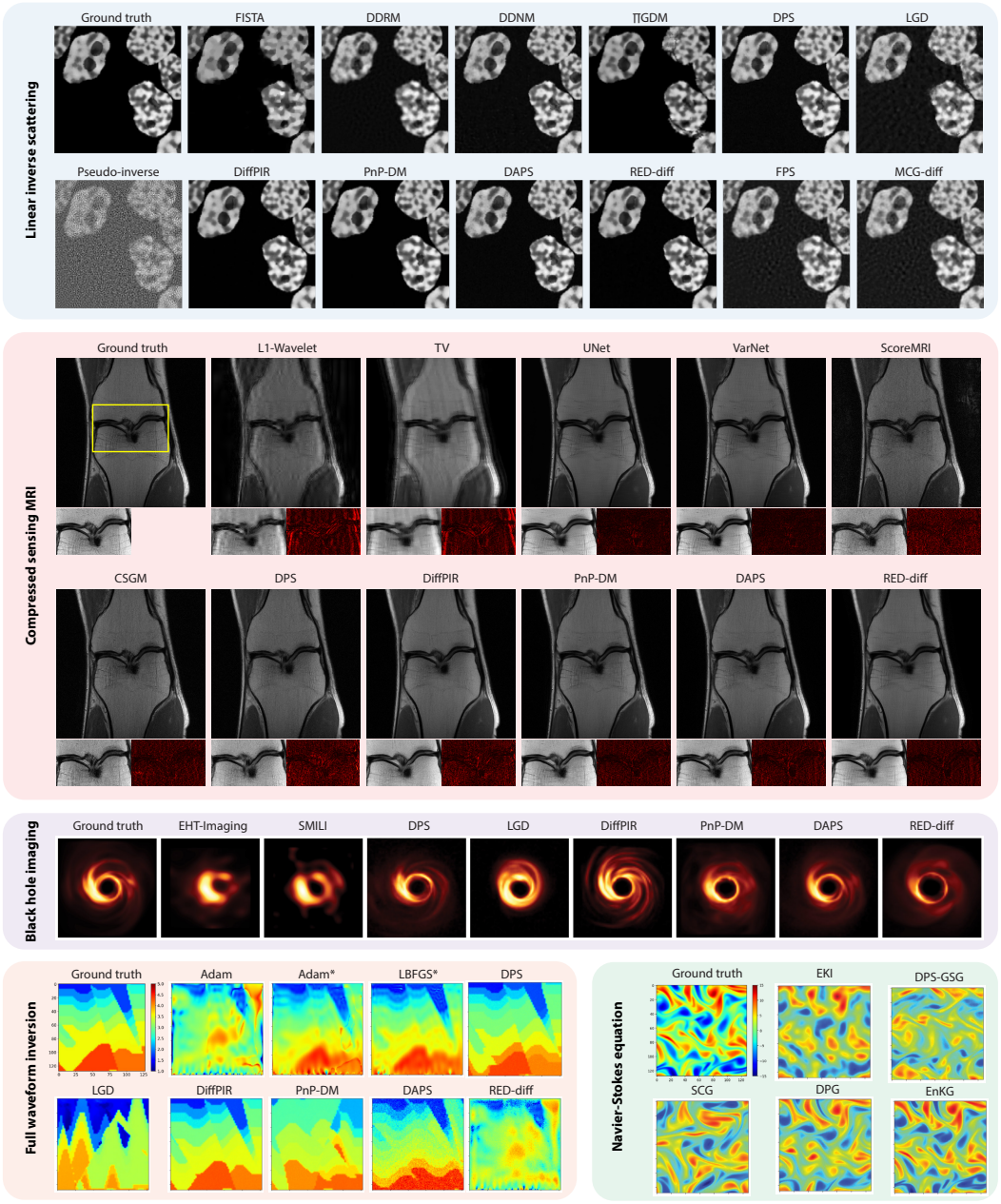


Figure 8.2: **Qualitative comparison showing representative examples of PnP-DP methods and domain-specific baselines across five inverse problems.** Note that for full waveform inversion, Adam* and LBFGS* are initialized with Gaussian-blurred ground truth, serving as references.

the PnPDP methods generally lie higher along the vertical axis. This finding is as expected, given that the baselines do not incorporate such strong prior information.

However, if the classic optimization baselines are initialized well, then they sometimes outperform PnPDP methods, most of which cannot naturally incorporate an

initialization beyond white noise. For example, in FWI, PnPDP methods clearly outperform the classic baseline methods if the baselines are initialized randomly or from a constant. But if initialized with a good guess (e.g., a heavily blurred ground truth image), they consistently outperform the current PnPDP methods. That being said, the fact that PnPDP methods rely much less on initialization than the traditional optimization methods is already an intriguing property. See qualitative comparison in Figure 8.1 and quantitative comparison in Table E.5.

How do PnPDP methods compare with each other? In the problems where the forward model has a closed-form expression, methods that require more gradient queries, such as DAPS and PnP-DM, tend to be more accurate. However, since they have more queries to the forward model, they are also more expensive, as shown in Figure 8.1. Additionally, these methods require more careful tuning as they usually have larger hyperparameter spaces, as shown in Table E.10[1].

In the problems where the forward model has no closed-form expression, particularly a forward model defined by a PDE system and implemented as a numerical PDE solver, this trend does not hold. In fact, DAPS and PnP-DM perform poorly, as shown in Figure 8.1 and Table E.5. These methods also demonstrate an increased level of numerical instability and sensitivity to hyperparameters, as shown in Figure 8.3: minor adjustments in step size can lead to either unconditional generation results that ignore measurements (with slightly smaller steps) or complete failure (with slightly larger steps). This performance degradation stems from a critical limitation in many current PnPDP algorithms: they do not account for stability conditions required to query a forward model. For example, in the FWI and Navier-Stokes equation, the input of the forward model must satisfy the Courant–Friedrichs–Lewy (CFL) condition [76] to produce stable solutions. This issue is particularly pronounced for methods like DAPS and PnP-DM, which incorporate Langevin Monte Carlo (LMC) as a subroutine, because LMC introduces additional Gaussian noise at each step and thus exacerbates instability compared to other PnPDP methods.

How does the performance vary with different levels of measurement sparsity? As measurement sparsity increases, making the inverse problem more ill-posed, we observe an increasingly wide performance gap between PnPDP methods and baselines. Figure 8.4 illustrates this trend across three problems, showing that

[1]Note that tuning the hyperparameters of PnPDP approaches is still much more efficient than retraining a neural network that is typically required for end-to-end approaches.

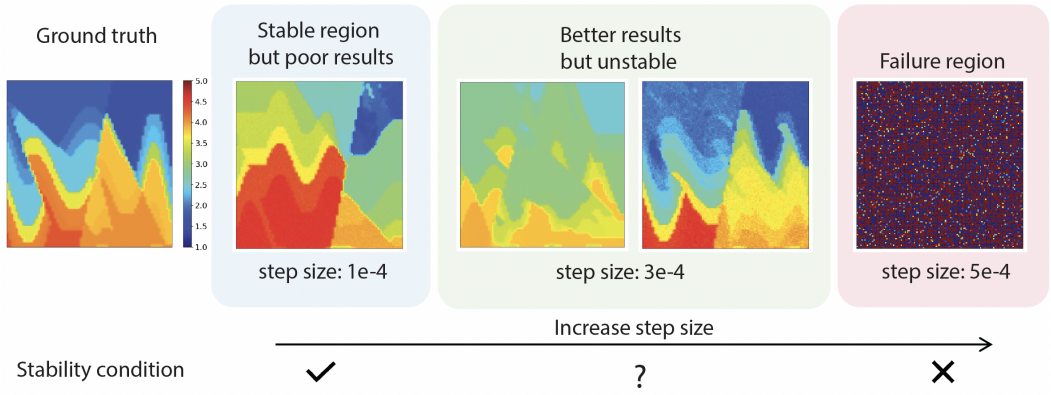


Figure 8.3: **Illustration of the failures of PnPDP methods (DAPS) as an example) on full waveform inversion.** With a small learning rate, DAPS is numerically stable but does not solve the inverse problem effectively. With a slightly larger learning rate, DAPS produces a noisy velocity map that breaks the stability condition of the PDE solver, resulting in a complete failure.

the average performance gain of top PnPDP methods over baselines grows with increasing measurement sparsity.

How well do PnPDP methods deal with different forward models? For linear inverse problems, our results demonstrate that PnPDP methods can effectively deal with varying forward models without the need for parameter tuning. To validate this, we conduct a controlled experiment in CS-MRI, where we maintain a consistent measurement sparsity while altering the subsampling pattern (from vertical to horizontal lines). We assess the average performance variation across three method categories: traditional baselines, end-to-end approaches, and PnPDP methods. The average absolute performance change for PnPDP methods is 0.48dB (PSNR) and 0.016 (SSIM), comparable to the traditional baseline methods at 1.62dB (PSNR) and 0.027 (SSIM), but significantly smaller than the end-to-end methods, which exhibit changes of 9.58dB (PSNR) and 0.21 (SSIM). These findings indicate that PnPDP methods are more robust than both baseline and end-to-end methods when handling different forward models.[2]

How well do PnPDP methods handle out-of-distribution sources? In general, if the unknown source falls outside the diffusion prior distribution, PnPDP methods tend to generate solutions that are biased toward the prior. As illustrated in Figure 8.5a, most solutions produced by PnPDP methods exhibit a black hole ring

[2]For end-to-end approaches, this is considered as an out-of-distribution test.

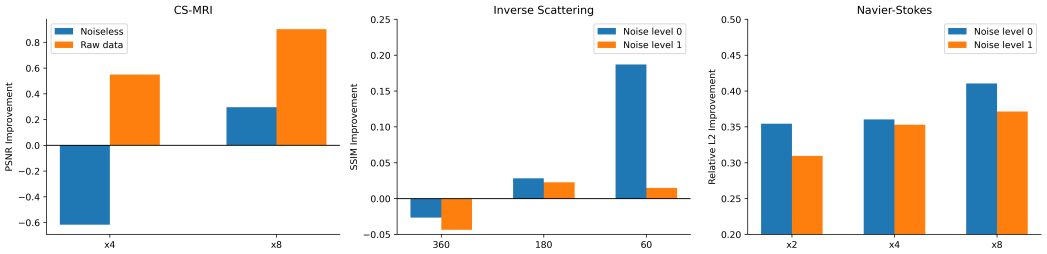
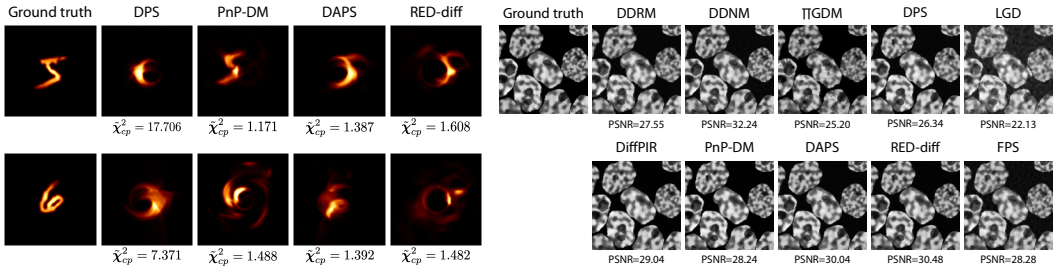


Figure 8.4: **Relative performance of plug-and-play diffusion prior methods compared with traditional baselines under different levels of measurement sparsity on different tasks.** Metrics are averaged over multiple PnPDP methods. The performance difference increases in general as the measurement becomes sparser.



(a) (b)

Figure 8.5: **PnPDP methods on out-of-distribution test samples.** (a) Black-hole imaging problem on digit inputs; and (b) inverse scattering on sources that contain 9 cells, while the prior model is trained on images with 1 to 6 cells.

feature characteristic of the diffusion prior. This suggests that while PnPDP approaches are flexible in capturing high-dimensional priors, they are limited in their ability to reliably recover "surprising" sources that lie outside the support of the diffusion prior distribution. However, when the unknown source is close to the diffusion prior distribution, PnPDP methods can recover it effectively, as demonstrated in Figure 8.5b.

8.5 Conclusion

We conclude this chapter by highlighting key research opportunities for advancing PnPDP methods in solving inverse problems. In this chapter, we only consider the setting where the forward model is exact and explicit, while the forward model in real-world problems often involve approximations. Studying the robustness of PnPDP methods when there is a mismatch between the assumed and actual forward model would be an essential step toward their practical deployment. This includes evaluating how well they handle imperfect and/or partially known forward model representations. Another research challenge we identify is that the current PnPDP

methods do not account for stability conditions required to query a forward model, which leads to degraded performance and numerical instability. However, many scientific inverse problems are based on PDE systems that require certain conditions on the inputs to simulate numerically and violating these constraints can result in meaningless solutions. Another direction for improvement is inference speed. As shown in Figure 8.1, almost all the PnPDP methods are less computationally efficient than the conventional baselines. There remains substantial room for optimization. We hope that INVERSEBENCH can serve as a systematic benchmark and catalyst for developing more advanced diffusion-based posterior samplers capable of addressing these challenges.

Chapter 9

CONCLUSION

Computational imaging systems push the boundaries of imaging techniques by enhancing physical sensors with computational algorithms. In this thesis, we presented two lines of research that improve *sampling* in the computational imaging pipeline using machine learning (ML)—one around measurement acquisition with physical sensors and another around posterior estimation with computational algorithms. Together, these contributions address key challenges and unlock new capabilities for existing computational imaging systems. This chapter concludes the thesis by highlighting the key takeaways and proposing some promising directions for future research.

9.1 Key Takeaways

In Part I, we investigated how to improve sampling strategies for compressed sensing MRI (CS-MRI) through end-to-end ML approaches. Traditional CS-MRI methods use fixed, predetermined subsampling patterns that are agnostic to both the underlying target and downstream tasks. In contrast, we showed that ML-based approaches can effectively learn sampling strategies that are adaptive to the underlying target or specific to the downstream objectives:

- **Sequential sampling:** In Chapter 3, we proposed a method for learning sequential sampling strategies for CS-MRI, leveraging the fact that measurements in CS-MRI must be taken sequentially rather than simultaneously. Our learned multi-step strategies decide which measurements to acquire based on the previously observed samples, thereby making better selections compared to non-adaptive baselines. In addition, our method outperforms reinforcement learning–based approaches in both accuracy and training efficiency.
- **Task-specific sampling:** In Chapter 4, we developed a framework that directly optimizes sampling patterns of CS-MRI for a downstream task, rather than reconstruction metrics that may not align with clinical or diagnostic objectives. While this may sound like an easy tweak, we showed that the naïve end-to-end optimization approach does not lead to a robust strategy. Instead, we introduced a two-stage training procedure: first pre-training a backbone model for reconstruc-

tion, followed by task-specific fine-tuning with a prediction head—analogous to modern foundation model workflows. We showed that our approach consistently outperforms existing methods that do not consider downstream tasks in terms of task-specific metrics. We also implemented the learned sampling strategies on a real MRI scanner and validated their effectiveness with experimentally collected data.

In Part II, we introduced a new approach of using diffusion models (DMs) as priors for estimating the posterior distributions of inverse problems in Bayesian inference. The proposed approach addresses several core challenges:

- **Principled method:** In Chapter 6, we proposed PnP-DM, a posterior sampling method based on the Split Gibbs Sampler (SGS) [296]. Unlike previous methods, our method avoids making uncontrolled approximations that could lead to significant sampling errors, thereby enabling more accurate posterior estimation. We drew a key connection between SGS and the EDM framework [151], which provides a unified and principled way of incorporating DMs as priors for solving inverse problems.

- **Unified framework:** In Chapter 7, we built on the core ideas of PnP-DM and extended them into a unified sampling framework for diffusion-based posterior estimation. We presented four instantiations of this framework, each designed to address one limitation in PnP-DM. These include incorporating text as a semantic prior, scaling to high-dimensional video data, handling non-differentiable (black-box) forward models, and enabling posterior inference in discrete spaces. While addressing diverse challenges, all four methods share a common alternating-update structure inspired by PnP-DM, and they exhibit both strong theoretical guarantees and robust empirical performance across a range of tasks.

- **Comprehensive benchmark:** In Chapter 8, we presented a large-scale benchmark designed to rigorously evaluate diffusion-based approaches for solving scientific inverse problems. The benchmark encompasses 14 widely used methods and 5 representative inverse problems spanning multiple scientific fields. Through comprehensive evaluations and ablation studies, we provided critical insights into the capabilities and shortcomings of current techniques and highlighted future opportunities for improving posterior sampling in computational imaging.

9.2 Futures Directions

While the contributions above advance both measurement acquisition and posterior estimation in computational imaging, many open problems remain. We outline a few promising directions below.

9.2.1 Measurement Acquisition (Part I)

Optimizing Measurement Acquisition for Posterior Sampling Methods The proposed methods in Part I rely on end-to-end network architectures, which require paired data to train and could be prone to overfitting due to a lack of uncertainty quantification in the decision-making process. Incorporating posterior sampling methods like PnP-DM from Part II could address problems where paired data is unavailable and lead to better performance and robustness. It remains a challenge to optimize the sampling patterns with posterior sampling methods that do not permit backpropagation.

Implementation of Sequential Strategies While our learned sequential sampling strategies showed significant improvements in retrospective experiments, it remains an open question whether these improvements translate to real-world performance when deployed on an actual MRI scanner. A key implementation challenge lies in integrating deep learning-based sequential policies into MRI sampling sequences without causing disruptions for the underlying acquisition protocols.

Extension to Other Imaging Modalities While our methods are developed for CS-MRI, they may generalize to other modalities. For example, in computed tomography (CT), one could optimize the angular views of X-ray acquisition for downstream diagnostic tasks. Exploring these extensions could broaden the impact of adaptive and task-driven sampling strategies.

9.2.2 Posterior Estimation (Part II)

Partially-Known or Mismatched Forward Models The proposed methods in Part II assume the forward model $\mathcal{A}$ to be accurate, which is sometimes unrealistic in practice. Many real-world problems, such as photographic deblurring and radio astronomy, involve uncertainty or a mismatch in the forward process. Developing posterior estimation methods that can handle these imperfections remains an open challenge.

More General Noise Distributions The current framework is designed under the assumption of additive Gaussian measurement noise. However, many real-world imaging problems involve structured or non-Gaussian noise. Extending the framework to accommodate more general and complex noise distributions (e.g., Poisson, impulse, structured noise) is an important direction.

4D Imaging While the proposed framework can handle inverse problems on 3D videos, many real-world imaging problems involve inherently 4D data. Scaling up the current framework to handle the spatiotemporal evolution of 3D volumes would be an interesting future direction. For example, one potentially viable idea is to partition 3D volumes into lower-dimensional slabs or cubes that are more computationally tractable.

Compatibility with Autoregressive Generative Models Autoregressive models have recently achieved state-of-the-art performance on generative modeling [281]. Even though they are not strictly DMs, they share a similar iterative generation procedure over a sequence of image manifolds. It would be worthwhile to explore whether the insights and techniques we developed for DM-based posterior estimation can be leveraged for designing more powerful techniques with autoregressive models.

9.3 Concluding Remarks

As computational imaging continues to push the boundaries of science and engineering, the opportunities to improve how we see and interpret the world are expanding rapidly. This thesis contributes to that progress by proposing new approaches for more intelligent measurement acquisition and principled posterior estimation—two foundational stages of the imaging pipeline. We believe that our results not only address concrete challenges and demonstrate practical impact, but also open the door to promising future directions that further enhance the performance, robustness, and efficiency of computational imaging systems. Realizing these advances will require even deeper integration between physical models and learning-based algorithms. Looking ahead, we hope the insights and frameworks developed in this work will serve as building blocks for the next generation of intelligent imaging systems.

Part III

Appendices

Appendix A

APPENDIX FOR CHAPTER 3

A.1 Model Architectures

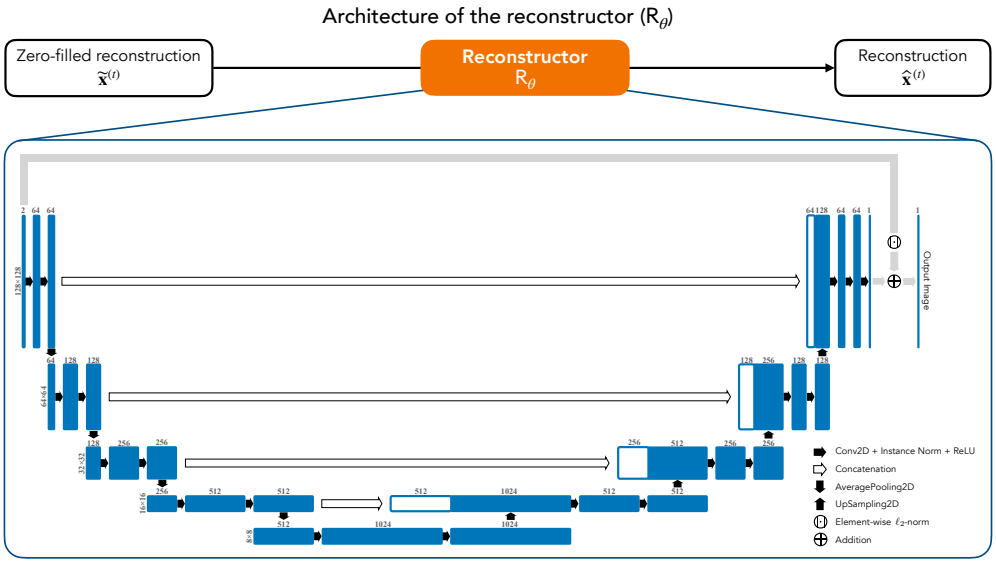


Figure A.1: **Flow diagram of the reconstructor, $\mathsf{R}_\theta(\cdot)$, in the proposed framework.** We use a residual U-Net reconstructor for all of our models.

A.1.1 Reconstructor

In Figure A.1, we provide the network diagram for the reconstructor. As stated in the main paper, we use the standard U-Net architecture following [12, 13, 338]. The input to the reconstructor is the complex-valued zero-filled image, and the output is a single-channel real-valued image. The initial convolutional layer has 64 channels, which are doubled after every downsampling layer. The reconstructor uses skip-connections, depicted as white horizontal arrows, that concatenate feature maps at different levels for easier optimization.

A.1.2 Samplers

Figure A.2 shows the detailed architecture of our neural samplers. On top, we show the sampler architecture for the 1D line sampling setting, which is a five-layer Multilayer Perceptron (MLP) network with 512 channels in the hidden layers. The input to the sampler includes the up-to-date sampling mask $\boldsymbol{M}^{(t)}$, k-space

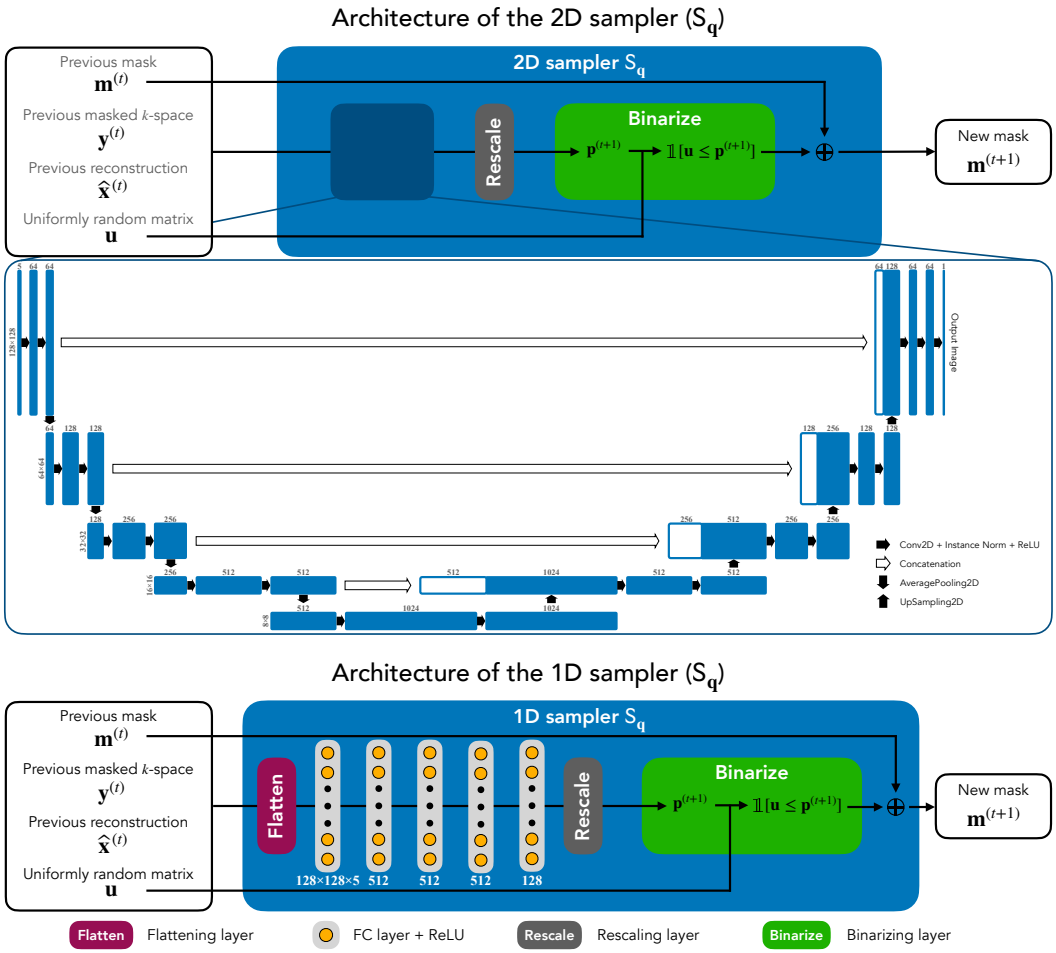


Figure A.2: **Flow diagram of the samplers, $\mathsf{S}_q(\cdot)$, in the proposed framework.** We use a Multilayer Perceptron for 1D line sampling and a U-Net for 2D point sampling. Both networks take previous observations as inputs and output a probability map, which is rescaled and binarized into the final sub-sampling mask at the next iteration.

measurements $\boldsymbol{y}^{(t)}$, and the k-space of reconstruction $\boldsymbol{F}\widehat{\boldsymbol{x}}^{(t)}$. A uniformly random vector $\boldsymbol{u}$ is also given as an input for Bernoulli sampling. The output is a binary sampling mask $\boldsymbol{m}^{(t+1)}$ generated through stochastic binarization with the random vector $\boldsymbol{u}$. The bottom shows the 2D point sampling architecture. For the 2D setting, the sampler uses a U-Net architecture [250]. Inputs and outputs are the same as the 1D line sampler.

A.2 Baseline Details

A.2.1 Random

Random subsampling is a widely used k-space sampling pattern that utilizes stochastic subsampling for creating incoherent artifacts that can be easily recognized and

removed through post-processing techniques [110]. In our implementation, we first pre-select the central low-frequency k-space region and then uniformly sample from the remaining lines or points until exhausting the sampling budget. We pair this sampler with a U-Net reconstructor trained with this random sampling pattern for 50 epochs. The reconstructor architecture and training schedule are the same as those of our sequential models.

A.2.2 Equispaced

Equispaced subsampling is another widely used 1D line sampling baseline [338]. Lines are sampled equidistantly from each other with an offset to achieve the desired sampling budget. We choose the equispaced baseline due to its ease of implementation on existing MRI scanners [338].

A.2.3 Spectrum

Spectrum is a data-driven k-space sampling approach introduced in [289]. The spectrum method utilizes the fact that k-space samples with higher power often contain more information about the image's large-scale structure. To identify the k-space samples, we average the magnitude spectrum of all fully-sampled k-space data in the training set. We then select samples with the largest average power, which will form the final subsampling mask. We pair this sampler with a U-Net reconstructor trained using measurements acquired according to this learned sampling pattern.

A.2.4 LOUPE

LOUPE [12] is the state-of-the-art single-shot sampling method. It jointly optimizes an subsampling pattern along with an image reconstruction network. We follow the official implementation in [12] but replace the binarization function in the subsampling mask generation with a straight-through estimator following [20, 344]. The same modification is applied to our method as described in Section 3.3.1. The reconstructor architecture and other hyperparameters are the same as those of our sequential methods.

A.2.5 PG-MRI

PG-MRI [13] formulates the k-space sample selection as a partially observable Markov decision process and learns a sequential sampling policy using the policy gradient algorithm [18]. According to their evaluations, PG-MRI outperforms multiple baseline approaches, including uniform random [110], equispaced sampling [338] and another Monte-Carlo tree search-based reinforcement learning ap-

proach [146]. We use the author's official code for our implementation. The reconstructor is pre-trained using the uniform random policy. We then plug the pre-trained reconstructor into the pipeline to evaluate the image reconstruction reward for sampling policy training. All other hyperparameters are the same as the original paper [13].

A.2.6 Evaluator

[351] proposed a greedy acquisition framework that trains a ResNet to reconstruct the anatomical image simultaneously with an Evaluator network trained to select the most uncertain measurements in k-space. As there is no official code available for [351], we use the reimplementation in [231]. The reconstructor uses a cascade ResNet architecture with four cascade blocks, each composed of three residual bottleneck layers [126] followed by a data consistency layer [255]. The evaluator contains four convolutional blocks, and each consists of a 4×4 convolution, instance normalization, and a LeakyReLU activation layer [209]. We use a batch size of 128 and train the model for 200 epochs with a learning rate of $1e-4$ using the Adam optimizer [162].

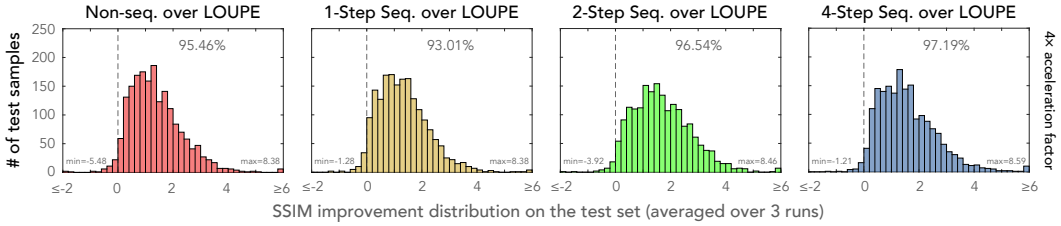


Figure A.3: **Histograms of pair-wise SSIM differences on all 1,851 test images using 1D line sampling with 4× acceleration factor.** We calculate the improvement of our model with different sequential steps over LOUPE. Our sequential model and non-sequential baseline significantly outperform LOUPE for most subjects.

A.3 Further Analyses

A.3.1 Pair-Wise Comparison for 1D Line Sampling

We report extended pair-wise SSIM comparisons for 1D line sampling on the test set. Figure A.3 and Figure A.4 show the SSIM improvement distribution on the test set. Here, we compare our method with the previous sequential sampling approach Evaluator [351] and PG-MRI [13] by measuring their improvements over the state-of-the-art single-shot sampling baseline LOUPE [12]. Our model outperforms LOUPE for 97.19% of the targets while both previous sequential sampling baselines perform substantially worse than the LOUPE baseline, with only 1.58% and 5.64%

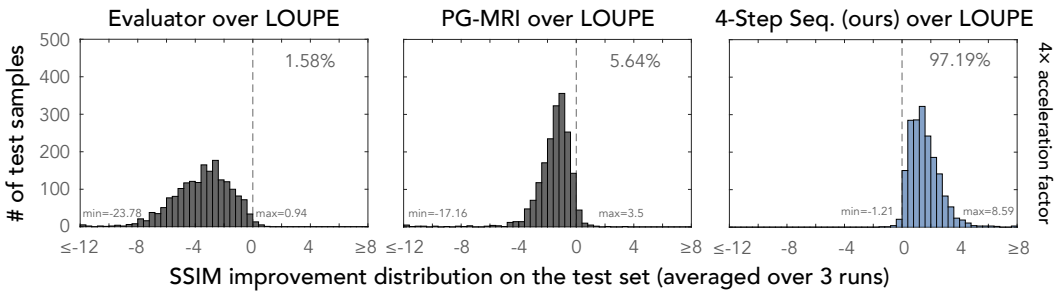


Figure A.4: **Histograms of pair-wise SSIM differences on all 1,851 test images using 1D line sampling with 4× acceleration factor.** We calculate the improvement of the Evaluator (left), PG-MRI (middle), and our best sequential model (4-step sequential) (right) over LOUPE. Our 4-step sequential model significantly outperforms LOUPE, while the other two baselines are substantially worse than LOUPE for most subjects.

of the subjects outperforming LOUPE for Evaluator and PG-MRI, respectively. This highlights the importance of combining co-design and sequential sampling in an end-to-end fashion for MR k-space sampling.

A.3.2 Pair-Wise Comparison for 2D Point Sampling

In Figure A.5, we show the SSIM improvement distribution for different methods compared to the LOUPE baseline. The histograms across each row show that, for all three acceleration factors, the non-sequential model has marginal improvement over the LOUPE baseline; in contrast, our sequential model significantly outperforms LOUPE as we increase the number of sequential sampling steps. By inspecting Figure A.5 down each column, our models demonstrate increasingly larger advantages over LOUPE as the number of sampled measurements increases from 16× to 4× (i.e., the acceleration factor decreases).

A.4 Additional Reconstruction Examples

We present some additional reconstruction examples in Figure A.6 and Figure A.7.

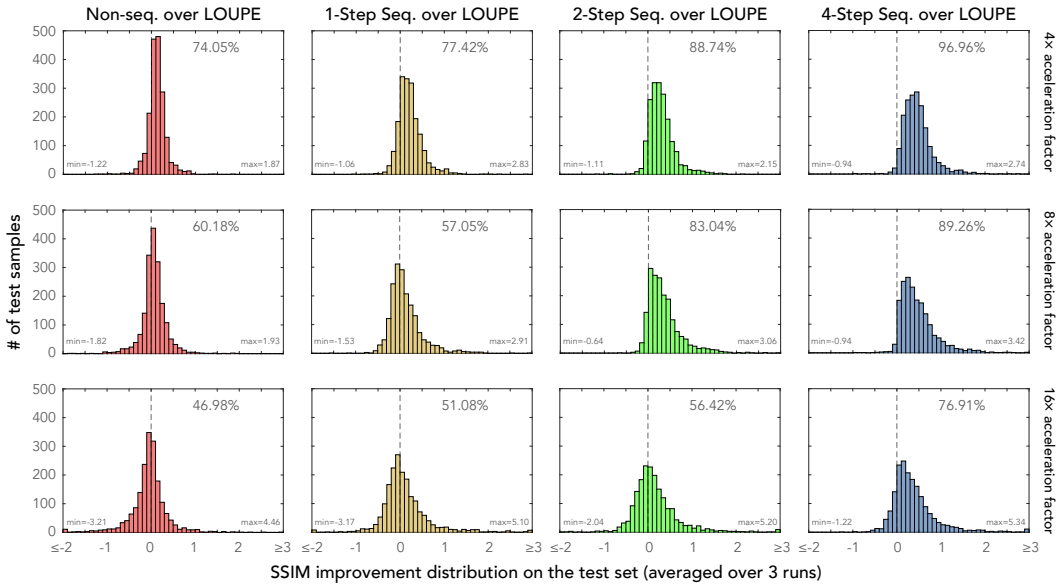


Figure A.5: **Histograms of pair-wise SSIM differences on all 1,851 test images using 2D line sampling with 4× (first row), 8× (second row), and 16× (third row) acceleration factors.** We calculate the improvement of our model with different sequential steps over LOUPE in each column. For all three acceleration factors, our sequential model outperforms the non-sequential baseline and LOUPE on an increasing percentage of test samples as the number of sequential steps increases. Our sequential models also have increasingly larger advantages over LOUPE as the number of sampled measurements increases (i.e., the acceleration factor decreases).

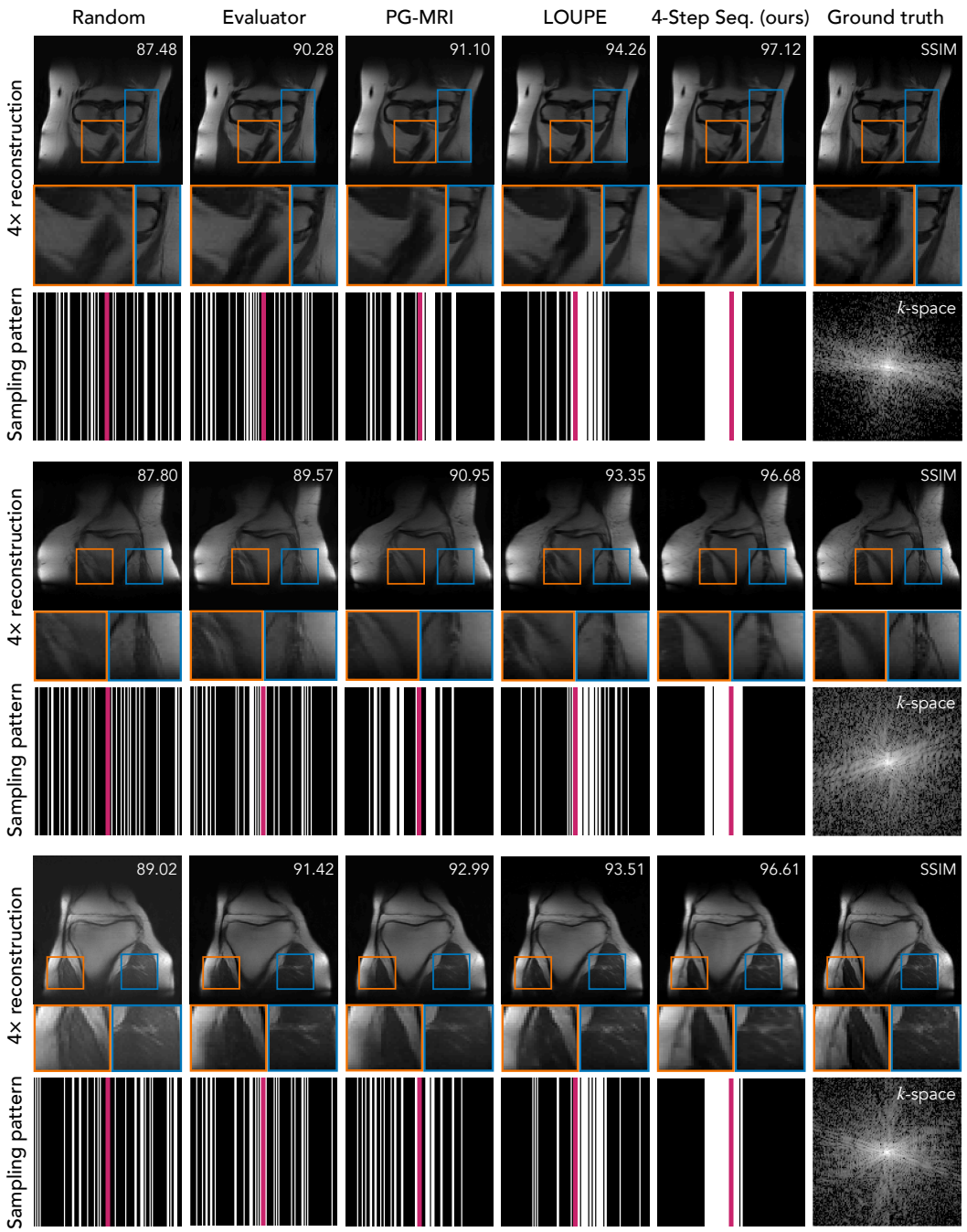


Figure A.6: **Visualizations of the reconstructions of the 394$^{\text{th}}$ (top), 1083$^{\text{th}}$ (middle), 1506$^{\text{th}}$ (bottom) test images with an acceleration factor of 4× for 1D line sampling.** Zoomed-in image patches highlight our significant improvement over previous methods. We find that our learned masks for the 1D line sampling case usually consist of adjacent low-frequency samples. However, only a few of the learned samples have their conjugate symmetric points sampled as well. Our learned policy appears to leverage the conjugate symmetry of the k-space and trade off taking more measurements with taking fewer measurements with higher SNR (by effectively sampling the same measurement twice).

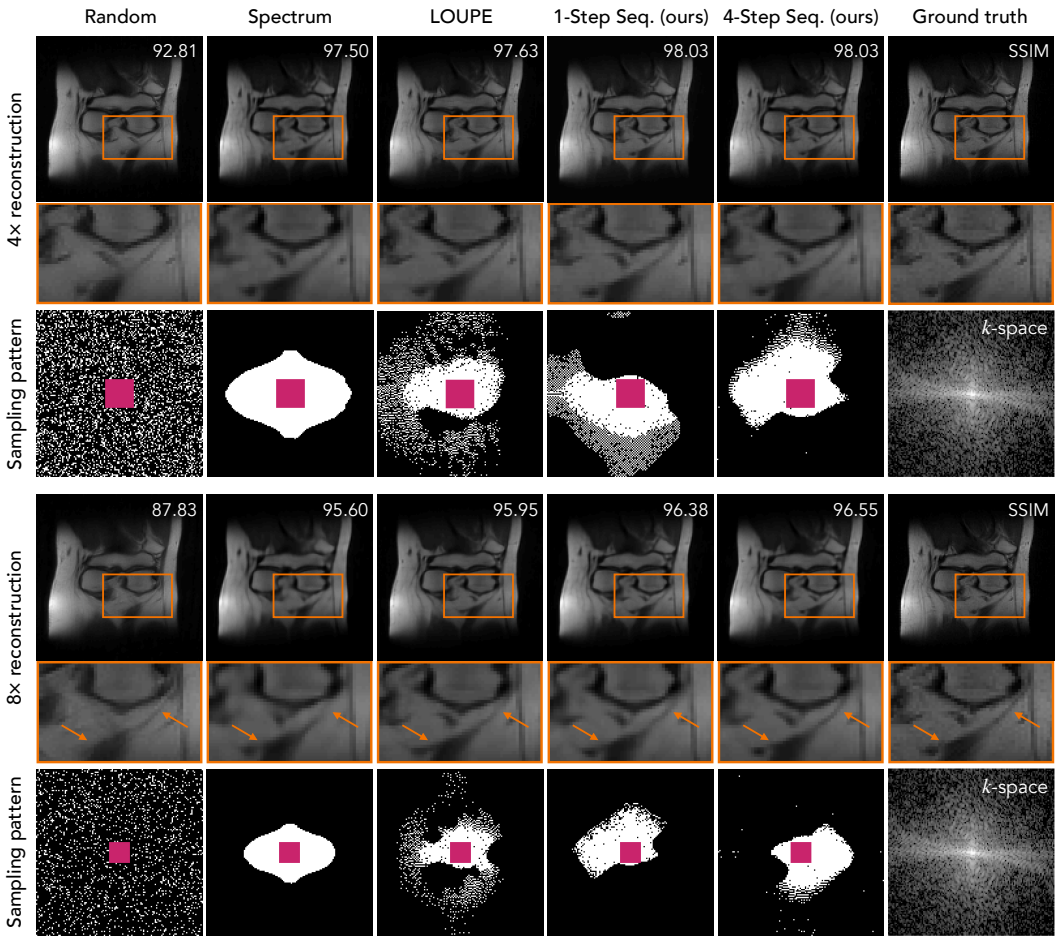


Figure A.7: **Visualizations of the reconstructions of the 1355$^{\text{th}}$ test sample with an acceleration factor of 4× (top) and 8× (bottom) for 2D point sampling.** A zoomed-in image patch is shown along with the cumulative k-space measurements selected by each policy. Orange arrows point out the regions where our sequential approach provides more accurate and detailed local structures.

Appendix B

APPENDIX FOR CHAPTER 4

B.1 Implementation Details

In this section, we describe the implementation details of TACKLE and the baseline methods.

B.1.1 Further Information on Datasets and Their Preparation

For each dataset in Section 4.4, we randomly split the data into training, validation, and test sets *on the patient level*, which means that each validation or test slice comes from a patient whose images are not used for training.

B.1.1.1 ROI-Oriented Reconstruction

For this task, we use all images with Meniscus Tear (MT) annotations in the fastMRI+ dataset [338, 354]. We follow the specific data splitting in [231], which results in 4,158 slices for training, 210 slices for validation, and 201 slices for testing. We crop the center of the k-space of each image and adjust the size and position of each bounding box accordingly.

B.1.1.2 Brain Tissue Segmentation

For this task, we use the 109th coronal slice of each volume in the OASIS dataset [213]. The access to the dataset can be found here[1]. Specifically, we use the 4-label tissue-type segmentation maps, which include segments of the cortex, the white matter, the subcortical gray matter, and the cerebrospinal fluid (CSF). We split the data into 248 slices for training (≈60%), 82 slices for validation (≈20%), and 84 slices for testing (≈20%).

B.1.1.3 Knee Tissue Segmentation

For this task, we use all the sagittal slices in the SKM-TEA dataset [83] that contains all four segmentation labels (the patellar cartilage, the femoral cartilage, the tibial cartilage, and the meniscus). We split the data into 2,935 slices for training (≈60%), 1,040 slices for validation (≈20%), and 987 slices for testing (≈20%).

[1] `https://github.com/adalca/medical-datasets/blob/master/neurite-oasis.md`

B.1.1.4 Pathology Classification

For this task, we use all the images acquired by the FLAIR sequence in the BRATS dataset [219] to detect the existence of the Glioma tumor. FLAIR stands for *fluid attenuated inversion recovery*, a kind of inversion recovery sequence that is commonly used for detecting various brain lesions due to its ability to suppress the CSF signal and enhance lesion-to-background contrast [154]. Empirically, we find that it is more accurate to detect the existence of the Glioma on FLAIR images than on images with the other contrasts in the BRATS dataset. We split the data into 30,495 slices for training (≈60%), 9,996 slices for validation (≈20%), and 10,353 slices for testing (≈20%).

B.1.2 Training and Implementation Details of TACKLE

B.1.2.1 Training Details

For all experiments, the models were trained by the Adam [162] optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$ on a single NVIDIA A6000 GPU. We choose the best learning rate among {1e-2, 1e-3, 1e-4}, and trained all models until convergence (i.e., no improvement for 10 epochs on the validation set according to the task-specific evaluation metric). For instance, if a $\text{TACKLE}_{\text{seg.}}$ model achieves a higher Dice score on the validation set than all previous epochs on epoch 42, the model will be saved as a checkpoint. If it has no further improvement until epoch 52, then the training will be terminated, and the saved checkpoint on epoch 42 will be used for reporting the final results.

The training of our proposed framework is conducted by retrospective subsampling on fully sampled measurements. The first module is the sampler, which requires no input and directly learns a matrix that contains the probability of sampling each k-space frequency. The output of the sampler is the subsampling mask $\boldsymbol{m}$, in which 1 represents the measurements to be sampled and 0 represents those not to be sampled. Sampling amounts to taking the element-wise product between $\boldsymbol{m}$ and the fully sampled measurements $\boldsymbol{k}$, which gives us the subsampled measurements $\boldsymbol{y} := \boldsymbol{m} \odot \boldsymbol{k}$. The retriever will then take the two-channel complex measurements $\boldsymbol{y}$ as the input and output a single-channel real image $\widehat{\boldsymbol{x}}$. In the multi-coil case, $\boldsymbol{y}$ contains signals from multiple coils with different sensitivity maps, and $\widehat{\boldsymbol{x}}$ is reconstructed by taking the root sum of squares across all coils. For reconstruction tasks (full-FOV reconstruction and ROI-oriented reconstruction), $\widehat{\boldsymbol{z}} = \widehat{\boldsymbol{x}}$ will be the final output for loss calculation and back-propagation. For downstream tasks beyond reconstruction,

we feed $\widehat{\boldsymbol{x}}$ into an additional predictor which gives a prediction $\widehat{\boldsymbol{z}}$. In this case, $\widehat{\boldsymbol{z}}$ will be the final output for loss calculation and back-propagation.

B.1.2.2 Retriever Architecture

Following the E2E-VarNet architecture [267], our retriever operates in k-space and contains 12 refinement steps, each of which includes a U-Net [250] with independent weights from each other. The update rule of the t-th refinement step is

$$\boldsymbol{k}^{(t+1)} = \boldsymbol{k}^{(t)} - \eta^{(t)} \boldsymbol{m} \odot \left(\boldsymbol{k}^{(t)} - \boldsymbol{y} \right) + \mathsf{G}^{(t)} \left(\boldsymbol{k}^{(t)} \right)$$

where $\boldsymbol{m}$ is the subsampling mask, $\boldsymbol{y}$ is the measurement, $\boldsymbol{k}^{(t)}$ is the reconstructed k-space, $\eta^{(t)}$ is a data consistency parameter, and $\mathsf{G}^{(t)}$ is the refinement module defined as

$$\mathsf{G}^{(t)} \left(\boldsymbol{k}^{(t)} \right) := \boldsymbol{F} \boldsymbol{E} \left(\mathsf{UN}^{(t)} \left(\boldsymbol{R} \boldsymbol{F}^{-1} \boldsymbol{k}^{(t)} \right) \right).$$

Here, $\boldsymbol{E}$ and $\boldsymbol{R}$ are the expand and reduce operations across all coils (see [267] for more details), and $\mathsf{UN}^{(t)}$ is the U-Net model at the t-th step. Specifically, we use the standard U-Net [250] architecture with 2 input and output channels, 4 average down-pooling layers, and 4 up-pooling layers. The model starts with an 18-channel output for the input layer and doubles the number of channels with each downsampling layer. Between every two pooling layers are two convolution modules, each of which consists of a 3×3 convolution, an instance normalization [286], and a LeakyReLU activation with negative slope of 0.2. The input to each U-Net is first normalized to zero mean and standard deviation of 1 before being fed into the network, and will be normalized back to the original mean and standard deviation after passing through the network. After 12 refinement steps, the final output layer of the retriever is an inverse Fourier transform followed by a root-sum-squares reduction for each pixel over all coils. The output of the retriever is a batch of single-channel images. For reconstruction tasks, a loss function will be directly applied to the output. For non-reconstruction tasks, there is an additional predictor module.

B.1.2.3 Predictor Architecture

For tissue segmentation tasks, the predictor is a U-Net model that has the same architecture as the refinement network described above, except for the following differences: There is 1 input channel and c output channels (where c is the number of segmentation classes). The model starts with a 64-channel output for the input layer. The convolution modules use the Parametric ReLU activation. There is no

normalization after the output. We use the U-Net implementation in the `MONAI` package [45]. For the pathology classification task, the predictor is a ResNet18 model with 1 input channel and 2 output dimensions. We normalize the input to zero mean and a standard deviation of 1 before feeding it into the network. We use the ResNet implementation in the `torchvision` package [212].

B.1.3 Pre-Select Region and Sensitivity Map Estimation

Among all the datasets considered in this manuscript, fastMRI+ [338, 354] and SKM-TEA [83] contain multi-coil k-space measurements. Reconstruction from multi-coil k-space data requires estimation of the coil sensitivity maps, i.e., $\boldsymbol{S}_i$ in Equation (2.2), using the central low-frequency region of the k-space, called the Auto-Calibration Signal (ACS). We set the ACS region as a square around the DC component that contains $1/8$ of the subsampling budget. For example, if a dataset contains k-space measurements of size 256×256, for $8\times$ acceleration, we select the center 32×32 low frequencies as the ACS. We also include the pre-determined ACS region for single-coil k-space experiments because we find that it stabilizes the training of some baselines.

Given the ACS, we estimate coil sensitivity maps using the Sensitivity Map Estimation (SME) module introduced in [267]. In contrast to the ESPIRiT algorithm [285], SME estimates the sensitivity maps with a CNN applied to each coil image independently. The architecture of the CNN in SME is the same as the U-Net in each E2E-VarNet cascade, except with an 8-channel output instead of an 18-channel output for the input layer.

B.1.4 Further Details on the Implementation of SemuNet [309]

For the brain and knee segmentation tasks, we compare the proposed method with SemuNet [309]. Originally demonstrated for a brain segmentation task, it also aims to jointly optimize a subsampling mask, a reconstructor, and a task predictor for the downstream accuracy. SemuNet uses a hybrid of ℓ_1 loss for reconstruction and cross-entropy loss for segmentation. Since the code of SemuNet has not been released, we have tried to reproduce the results in the original paper to our best efforts. Specifically, we follow their proposed loss function and architecture of the sampler, the residual U-Net reconstructor, and a U-Net predictor. We follow the original paper to use an Adam optimizer [162] and not pre-select low-frequency measurements. However, since our tasks and datasets are different from those in [309], we empirically find that the learning rate and the parameter λ that adjusts

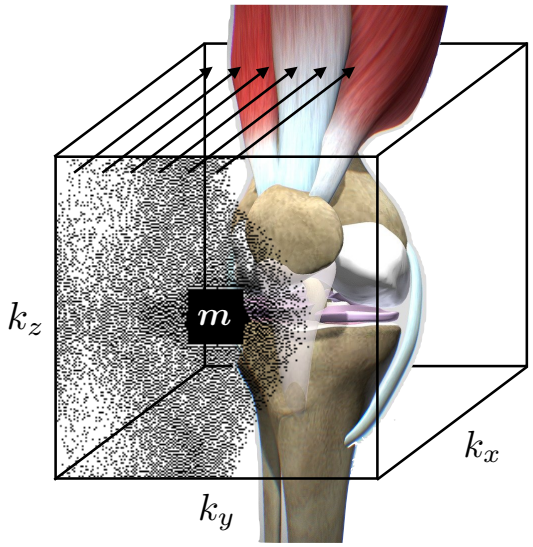


Figure B.1: **Conceptual illustration of the subsampling setup with a knee example.** The back dots on the k_y-k_z plane represent k-space trajectories along k_x, which are illustrated by the black arrows. We consider subsampling in the two phase-encoding dimensions (k_y and k_z) of a 3D Cartesian sequence, where the subsampling pattern $\boldsymbol{m}$ is learned from data for some specific downstream task.

the trade-off between the two losses are suboptimal for our settings. Therefore, we conduct a grid search on the learning rate in $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 5\}$ and $\lambda \in \{0.0001, 0.001, 0.01\}$. For both 16× and 64× accelerations, we choose the best combination of parameters based on the performance on the validation set, and report the Dice score on the test set.

B.2 Subsampling Setup and Implementation

In this work, we optimize the subsampling mask $\boldsymbol{m}$ over all 2D subsampling patterns. We implement 2D subsampling patterns in practice by subsampling in the two phase encoding dimensions of a 3D Cartesian sequence based on the 2D pattern, as illustrated in Figure B.1. We denote the number of trajectories along k_y and k_z (the two phase encoding directions) as n_{k_y} and n_{k_z}, respectively. For the fully sampling scenario, one needs to sequentially sample a total of $n_{k_y} n_{k_z}$ trajectories, which could take a long time to acquire in practice. Given a 2D subsampling mask $\boldsymbol{m}$, we subsample in the k_y-k_z plane according to $\boldsymbol{m}$. If $\boldsymbol{m}$ has an acceleration ratio of R, the subsampling sequence only takes $n_{k_y} n_{k_z}/R$ trajectories, and the acquisition time will be reduced by a factor R in practice. One can obtain the slice-wise 2D k-space measurements $\boldsymbol{y}$ by taking the 1D inverse Fourier Transform of the raw 3D k-space data along k_x. We have implemented a 4×-accelerated version of the sequence

Table B.1: **Comparison of TACKLE$_{\text{ROI}}$ on in- and out-of-distribution samples under different acceleration ratios (R).**

Metric	R	Samples w/ MT (in-distribution)	Samples w/o MT (out-of-distribution)
PSNR (dB)	4	37.65	36.92
	8	33.28	32.88
	16	32.06	31.74

that we use in Section 4.5 on the Siemens IDEA sequence programming platform, using the subsampling scheme we describe above. Figure 4.12 demonstrates that the prospective subsampling version of our learned sequence achieves the same level of visual quality as the fully sampling version but only takes a quarter of the scan time. This result highlights the real-world practicality of our approach.

B.3 Additional Validation on Out-of-Distribution Data

In Chapter 4, we show that TACKLE$_{\text{ROI}}$ improves the reconstruction of ROIs that contain the meniscus tear (MT). In practice, it is likely that a healthy subject or someone with a different pathological lesion from the meniscus tear will get scanned. So it is important that the learned sequence should also generalize to out-of-distribution subjects. Here we take our trained TACKLE$_{\text{ROI}}$ models with 4×, 8×, and 16× accelerations from our ROI reconstruction experiments, and directly test them on images that do not contain the meniscus tear, without additional fine-tuning. The results are summarized in Table B.1.

Although it is not surprising that TACKLE$_{\text{ROI}}$ performs better on the in-distribution data (samples w/ MT), we want to point out that the two numbers above correspond to two different test sets and are thus not directly comparable. The main takeaway is that TACKLE$_{\text{ROI}}$ can robustly recover samples without MT, even if it is trained on samples with MT. As discussed in Section 4.4.1, TACKLE$_{\text{ROI}}$ improves the ROI reconstruction by trading off k-space frequencies for the local anatomy to attain improved resolution. We find that such a strategy can lead to satisfactory reconstruction quality even when the underlying subject does not contain the target pathology.

B.4 Additional Results

We provide a box-plot comparison for the Meniscus Tear ROI reconstruction in Figure B.2. We also provide some visual examples for the tumor classification task

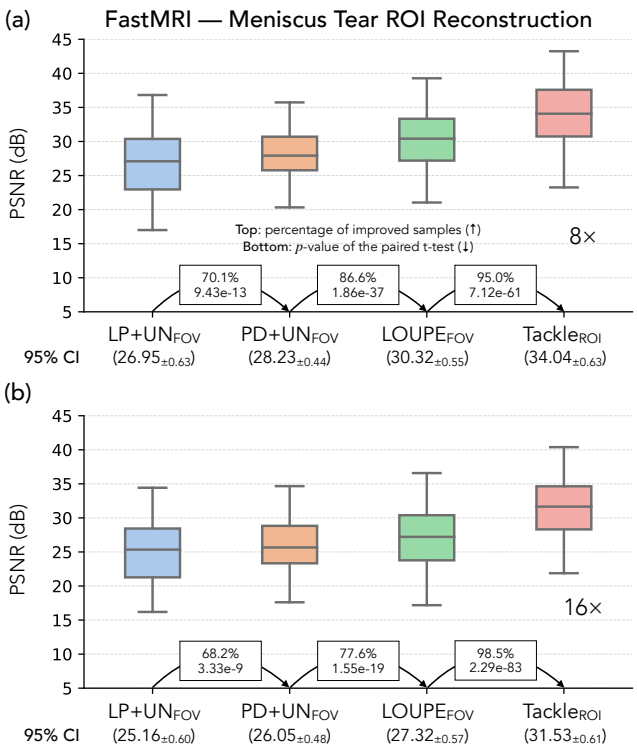


Figure B.2: **Box plots of the Meniscus Tear ROI reconstruction results under 8× (a) and 16× (b) accelerations.** Within the rectangle between each pair of methods, the top number is the percentage of samples that get improved, and the bottom number is the p-value given by the paired samples t-test. A higher percentage and a lower p-value indicate a more significant improvement. The 95% confidence intervals for all methods are given below their names.

in Figure B.3. For the three reconstruction-oriented baselines (left three columns), the inputs to the predictor network are typical reconstructions. Optimized end-to-end for classification accuracy, the retriever of TACKLE$_{\text{class.}}$ learns a feature map that highlights the region where a tumor could exist. Similar to the segmentation results, we find that the end-to-end model TACKLE$_{\text{class.}}$ circumvents the typical reconstruction but preserves image-level features that are helpful for downstream classification prediction.

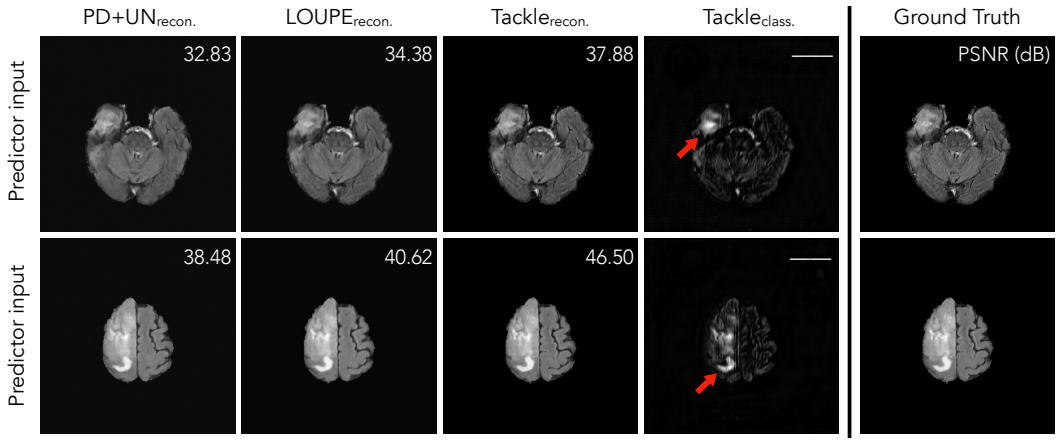


Figure B.3: **Visualization of the input of the predictor network for the brain tumor classification task under 16× acceleration on two samples of the BRATS dataset.** Similar to the segmentation results, as a co-design method, TACKLE$_{\text{class.}}$ circumvents the typical "reconstruction" in terms of point-wise similarity with the ground truth image. Instead, the retriever learns a feature map that highlights the region around the tumor for the downstream prediction.

Appendix C

APPENDIX FOR CHAPTER 6

C.1 Theory

C.1.1 Interpolation of PnP-DM

In this section, we formally introduce the interpolation of PnP-DM. We consider the case where the coupling strength η is constant, i.e., $\eta_k \equiv \eta$ and make the following assumption.

Assumption C.1.1. *There exists a unique t^* such that $\sigma(t^*) = \eta$.*

This assumption is satisfied for common diffusion models. Popular choices of the noise level schedule include $\sigma(t) = t$ or $\sigma(t) = \sqrt{t}$, which are monotonically increasing functions of t. We first present two propositions showing that the two steps in SGS can be implemented by running two SDEs.

Proposition C.1.2 (Brownian bridge for the likelihood step)**.** *For iteration k with iterate $\boldsymbol{x}^{(k)}$, the likelihood step of SGS is equivalent to solving the following SDE from $t = 0$ to $t = 1$:*

$$\mathrm{d}\boldsymbol{x}_t = \eta^2 \nabla \log \phi_t(\boldsymbol{x}_t)\mathrm{d}t + \eta \mathrm{d}\boldsymbol{w}_t \tag{C.1}$$

where $\boldsymbol{x}_0 = \boldsymbol{x}^{(k)}$ and $\phi_t(\boldsymbol{x}) := \int \exp[-f(\boldsymbol{z}; \boldsymbol{y}) - \frac{1}{2\eta^2(1-t)}\|\boldsymbol{x} - \boldsymbol{z}\|_2^2]\mathrm{d}\boldsymbol{z}$.

Proof. This proposition is due to the Brownian bridge construction presented in Lemma 4 of [336]. This SDE satisfies that $p(\boldsymbol{x}_1|\boldsymbol{x}_0) \propto \exp\left(-f(\boldsymbol{x}_1; \boldsymbol{y}) - \frac{1}{2\eta^2}\|\boldsymbol{x}_0 - \boldsymbol{x}_1\|_2^2\right)$. Therefore, solving Equation (C.1) from $t = 0$ to $t = 1$ is equivalent to taking a likelihood step. □

Proposition C.1.3 (EDM reverse diffusion for the prior step)**.** *For iteration k with iterate $\boldsymbol{z}^{(k)}$, the prior step of SGS is equivalent to solving the following SDE from $t = t^*$ to $t = 0$:*

$$\mathrm{d}\boldsymbol{x}_t = \left[u(t)\boldsymbol{x}_t - v(t)^2 \nabla \log p_t\left(\boldsymbol{x}_t\right)\right]\mathrm{d}t + v(t)\mathrm{d}\bar{\boldsymbol{w}}_t \tag{C.2}$$

where $\boldsymbol{x}_{t^*} = s(t^*)\boldsymbol{z}^{(k)}$, $u(t) := \frac{\dot{s}(t)}{s(t)}$, $v(t) := s(t)\sqrt{2\dot{\sigma}(t)\sigma(t)}$, and p_t is the distribution of $s(t)\boldsymbol{x} + s(t)\sigma(t)\boldsymbol{\epsilon}$ with $\boldsymbol{x}$ following the prior distribution $p(\boldsymbol{x}) \propto \exp(-g(\boldsymbol{x}))$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$.

Proof. First note that Equation (C.2) is exactly Equation (6.6) written in terms of $u(t)$ and $v(t)$. We know that the Equation (C.2) is the reverse SDE of the following SDE

$$\mathrm{d}\boldsymbol{x}_t = u(t)\boldsymbol{x}_t\mathrm{d}t + v(t)\mathrm{d}\boldsymbol{w}_t, \tag{C.3}$$

where $\boldsymbol{x}_0 \sim p(\boldsymbol{x})$ and p_t is the marginal distribution of $\boldsymbol{x}_t$. As shown in Section 6.3.2, it holds for Equation (C.3) that

$$p(\boldsymbol{x}_0|\boldsymbol{x}_t) \propto \exp\left(-g(\boldsymbol{x}_0) - \frac{1}{2\sigma(t)^2}\|\boldsymbol{x}_0 - \boldsymbol{x}_t/s(t)\|_2^2\right).$$

As Equation (C.2) is the time-reversed process of Equation (C.3), they share the same path distribution and thus the same conditional distribution $p(\boldsymbol{x}_0|\boldsymbol{x}_t)$. So, if we set $\boldsymbol{x}_{t^*} = s(t^*)\boldsymbol{z}^{(k)}$, we have that

$$p(\boldsymbol{x}_0|\boldsymbol{x}_{t^*}) \propto \exp\left(-g(\boldsymbol{x}_0) - \frac{1}{2\sigma(t^*)^2}\|\boldsymbol{x}_0 - \boldsymbol{z}^{(k)}\|_2^2\right) \propto \exp\left(-g(\boldsymbol{x}_0) - \frac{1}{2\eta^2}\|\boldsymbol{x}_0 - \boldsymbol{z}^{(k)}\|_2^2\right),$$

which is the desired conditional distribution of the prior step. Therefore, solving Equation (C.2) from $t = t^*$ to $t = 0$ is equivalent to taking a prior step. □

Due to Proposition C.1.2 and Proposition C.1.3, the SDEs Equation (C.1) and Equation (C.2) implement the two desired conditional distributions in SGS. In PnP-DM, the prior step involves a network that approximates the score function of the prior distribution, i.e., $\boldsymbol{s}_t \approx \nabla \log p_t$, so the continuous-time process for the actual update is

$$\mathrm{d}\boldsymbol{x}_t = \left[u(t)\boldsymbol{x}_t - v(t)^2\boldsymbol{s}_t\left(\boldsymbol{x}_t\right)\right]\mathrm{d}t + v(t)\mathrm{d}\bar{\boldsymbol{w}}_t. \tag{C.4}$$

We can then interpolate PnP-DM by considering a dynamic that alternates between running Equation (C.1) and Equation (C.4).

Since each likelihood step takes 1 unit of time and each prior step takes t^* units of time, the total time of the interpolating process for K iterations of PnP-DM is $T_K := K(t^* + 1)$. We use τ to denote the time that has elapsed from initializing PnP-DM with $\boldsymbol{x}^{(0)}$. We define $\{\mu_\tau\}$ and $\{\pi_\tau\}$ as the distributions at time τ of the non-stationary process initialized at $\boldsymbol{x}^{(0)} \sim \mu_0^X$ (Figure 6.2 top) and the stationary process initialized at $\boldsymbol{x}^{(0)} \sim \pi^X$ (Figure 6.2 bottom), respectively. Therefore, we have

- $\mu_\tau = \mu_k^X$, $\pi_\tau = \pi^X$ for $\tau = k(t^* + 1)$ with $k = 0, \cdots, K$, and
- $\mu_\tau = \mu_k^Z$, $\pi_\tau = \pi^Z$ for $\tau = k(t^* + 1) + 1$ with $k = 0, \cdots, K - 1$.

C.1.2 A Key Lemma for General Diffusion Processes

Before proving our main result, we present a key lemma for our analysis, which quantifies the time derivative of the KL divergence in terms of the Fisher divergence along a pair of general diffusion processes.

Lemma C.1.4. *Given the following pair of diffusion processes*

$$\mathrm{d}\boldsymbol{x}_t = b(\boldsymbol{x}_t, t)\mathrm{d}t + c(t)\mathrm{d}\boldsymbol{w}_t \tag{C.5}$$

$$\mathrm{d}\widetilde{\boldsymbol{x}}_t = \widetilde{b}(\widetilde{\boldsymbol{x}}_t, t)\mathrm{d}t + c(t)\mathrm{d}\boldsymbol{w}_t \tag{C.6}$$

where $b(\cdot,\cdot) : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$, $\widetilde{b}(\cdot,\cdot) : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$, *and* $c(\cdot) : \mathbb{R} \to \mathbb{R}$. *Let* μ_t *be the distribution of* $\boldsymbol{x}_t$ *initialized with* $\boldsymbol{x}_0 \sim \mu_0$ *for Equation* (C.5)*, and let* $\widetilde{\mu}_t$ *be the distribution of* $\widetilde{\boldsymbol{x}}_t$ *initialized with* $\widetilde{\boldsymbol{x}}_0 \sim \widetilde{\mu}_0$ *for Equation* (C.6)*. Then we have*

$$\partial_t \mathsf{KL}(\mu_t || \widetilde{\mu}_t) \leq -\frac{c(t)^2}{4}\mathsf{FI}\left(\mu_t || \widetilde{\mu}_t\right) + \frac{1}{c(t)^2}\int \left\| \widetilde{b}_t - b_t \right\|_2^2 \mu_t. \tag{C.7}$$

Proof of Lemma C.1.4. Writing $b(\cdot, t)$ as b_t and $\widetilde{b}(\cdot, t)$ as $\widetilde{b}_t$, by the Fokker-Planck equations of Equation (C.5) and Equation (C.6), we have that

$$\partial_t \mu_t = \operatorname{div}\left[\left(\frac{c(t)^2}{2}\nabla \log \mu_t - b_t\right)\mu_t\right] \quad \text{and} \quad \partial_t \widetilde{\mu}_t = \operatorname{div}\left[\left(\frac{c(t)^2}{2}\nabla \log \widetilde{\mu}_t - \widetilde{b}_t\right)\widetilde{\mu}_t\right].$$

Defining $\phi(x) := x \log x$ and $\phi'(x) = \frac{\mathrm{d}}{\mathrm{d}x}\phi(x) = \log x + 1$, we can calculate

$$
\begin{aligned}
&\partial_t \mathsf{KL}(\mu_t||\widetilde{\mu}_t) = \partial_t \int \phi\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\widetilde{\mu}_t \\
&= \int \phi'\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\left(\partial_t\mu_t - \frac{\mu_t}{\widetilde{\mu}_t}\partial_t\widetilde{\mu}_t\right) + \int \phi\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\partial_t\widetilde{\mu}_t \\
&= \int \phi'\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\left(\operatorname{div}\left[\left(\frac{c(t)^2}{2}\nabla\log\mu_t - b_t\right)\mu_t\right] - \frac{\mu_t}{\widetilde{\mu}_t}\operatorname{div}\left[\left(\frac{c(t)^2}{2}\nabla\log\widetilde{\mu}_t - \widetilde{b}_t\right)\widetilde{\mu}_t\right]\right) \\
&\quad + \int \phi\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\operatorname{div}\left[\left(\frac{c(t)^2}{2}\nabla\log\widetilde{\mu}_t - \widetilde{b}_t\right)\widetilde{\mu}_t\right] \\
&= -\int \left\langle \nabla\phi'\left(\frac{\mu_t}{\widetilde{\mu}_t}\right), \frac{c(t)^2}{2}\nabla\log\mu_t - b_t\right\rangle \mu_t + \int \left\langle \nabla\left[\phi'\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\frac{\mu_t}{\widetilde{\mu}_t}\right], \frac{c(t)^2}{2}\nabla\log\widetilde{\mu}_t - \widetilde{b}_t\right\rangle \widetilde{\mu}_t \\
&\quad - \int \left\langle \nabla\phi\left(\frac{\mu_t}{\widetilde{\mu}_t}\right), \frac{c(t)^2}{2}\nabla\log\widetilde{\mu}_t - \widetilde{b}_t\right\rangle \widetilde{\mu}_t \\
&= -\int \left\langle \nabla\phi'\left(\frac{\mu_t}{\widetilde{\mu}_t}\right), \frac{c(t)^2}{2}\nabla\log\left(\frac{\mu_t}{\widetilde{\mu}_t}\right) - b_t + \widetilde{b}_t\right\rangle \mu_t + \int \left\langle \nabla\frac{\mu_t}{\widetilde{\mu}_t}, \frac{c(t)^2}{2}\nabla\log\widetilde{\mu}_t - \widetilde{b}_t\right\rangle \phi'\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\widetilde{\mu}_t \\
&\quad - \int \left\langle \nabla\frac{\mu_t}{\widetilde{\mu}_t}, \frac{c(t)^2}{2}\nabla\log\widetilde{\mu}_t - \widetilde{b}_t\right\rangle \phi'\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\widetilde{\mu}_t \\
&= -\frac{c(t)^2}{2}\int \left\|\nabla\log\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\right\|_2^2 \mu_t - \int \left\langle \nabla\log\left(\frac{\mu_t}{\widetilde{\mu}_t}\right), \widetilde{b}_t - b_t\right\rangle \mu_t \\
&\leq -\frac{c(t)^2}{4}\int \left\|\nabla\log\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\right\|_2^2 \mu_t + \frac{1}{c(t)^2}\int \left\|\widetilde{b}_t - b_t\right\|_2^2 \mu_t \\
&= -\frac{c(t)^2}{4}\int \left\|\nabla\log\left(\frac{\mu_t}{\widetilde{\mu}_t}\right)\right\|_2^2 \mu_t + \frac{1}{c(t)^2}\int \left\|\widetilde{b}_t - b_t\right\|_2^2 \mu_t \\
&= -\frac{c(t)^2}{4}\mathsf{FI}\left(\mu_t||\widetilde{\mu}_t\right) + \frac{1}{c(t)^2}\int \left\|\widetilde{b}_t - b_t\right\|_2^2 \mu_t
\end{aligned}
$$

where we used the fact that $-\frac{1}{2}a^2 - ab \leq -\frac{1}{4}a^2 + b^2, \forall a, b \in \mathbb{R}$ for the inequality. □

C.1.3 Proof of Theorem 6.4.1

Now we are ready to prove Theorem 6.4.1.

Proof. We first consider the likelihood steps over K iterations of PnP-DM. Applying Lemma 2 of [336] to the likelihood steps Equation (C.1) of the non-stationary and stationary processes, we have that

$$\partial_\tau \mathsf{KL}(\pi_\tau||\mu_\tau) = -\frac{\eta^2}{2}\mathsf{FI}(\pi_\tau||\mu_\tau) \leq -\frac{\eta^2}{4}\mathsf{FI}(\pi_\tau||\mu_\tau),$$

for $\tau \in [k(t^*+1), k(t^*+1)+1]$ with $k = 0, ..., K-1$. Integrating both sides over $\tau \in [k(t^*+1), k(t^*+1)+1]$, we get

$$\int_{k(t^*+1)}^{k(t^*+1)+1} \mathsf{FI}\left(\pi_\tau || \mu_\tau\right) \mathrm{d}\tau = \frac{4[\mathsf{KL}(\pi^X || \mu_k^X) - \mathsf{KL}(\pi^Z || \mu_k^Z)]}{\eta^2} \tag{C.8}$$

for $k = 0, ..., K-1$.

Then, applying Lemma C.1.4 to the prior steps Equation (C.4) with

$$\begin{aligned} b(\boldsymbol{x}_t, t) &:= u(t)\boldsymbol{x}_t - v(t)^2 \nabla \log p_t\left(\boldsymbol{x}_t\right) \\ \widetilde{b}(\boldsymbol{x}_t, t) &:= u(t)\boldsymbol{x}_t - v(t)^2 \boldsymbol{s}_t\left(\boldsymbol{x}_t\right) \\ c(t) &:= v(t) \\ \delta &:= \inf_{t \in [0, t^*]} v(t), \end{aligned}$$

we have that

$$\begin{aligned} \partial_\tau \mathsf{KL}(\pi_\tau || \mu_\tau) &\leq -\frac{v(\tau)^2}{4} \mathsf{FI}\left(\pi_\tau || \mu_\tau\right) + \frac{1}{v(\tau)^2} \int \left\| v(\tau)^2 \left(\boldsymbol{s}_\tau - \nabla \log p_\tau\right) \right\|_2^2 \pi_\tau \\ &\leq -\frac{v(\tau)^2}{4} \mathsf{FI}\left(\pi_\tau || \mu_\tau\right) + v(\tau)^2 \int \left\| \boldsymbol{s}_\tau - \nabla \log p_\tau \right\|_2^2 \pi_\tau \\ &\leq -\frac{\delta^2}{4} \mathsf{FI}\left(\pi_\tau || \mu_\tau\right) + v(\tau)^2 \mathbb{E}_{\pi_\tau} \left\| \boldsymbol{s}_\tau - \nabla \log p_\tau \right\|_2^2, \end{aligned}$$

for $\tau \in [k(t^*+1)+1, (k+1)(t^*+1)]$ with $k = 0, ..., K-1$. Integrating both sides over $\tau \in [k(t^*+1)+1, (k+1)(t^*+1)]$, we get

$$\int_{k(t^*+1)+1}^{(k+1)(t^*+1)} \mathsf{FI}\left(\pi_\tau || \mu_\tau\right) \mathrm{d}\tau \leq \frac{4[\mathsf{KL}(\pi^Z || \mu_k^Z) - \mathsf{KL}(\pi^X || \mu_{k+1}^X)]}{\delta^2} + \frac{4\epsilon_{\text{score}}}{\delta^2} \tag{C.9}$$

where

$$\begin{aligned} \epsilon_{\text{score}} &:= \int_{k(t^*+1)+1}^{(k+1)(t^*+1)} v(\tau)^2 \mathbb{E}_{\pi_\tau} \left\| \boldsymbol{s}_\tau - \nabla \log p_\tau \right\|_2^2 \mathrm{d}\tau \\ &= \int_1^{t^*+1} v(\tau)^2 \mathbb{E}_{\pi_\tau} \left\| \boldsymbol{s}_\tau - \nabla \log p_\tau \right\|_2^2 \mathrm{d}\tau. \end{aligned}$$

Finally, combining Equation (C.8) and Equation (C.9) for $k = 0, ..., K-1$, we obtain

$$\begin{aligned} \int_0^{T_K} \mathsf{FI}\left(\pi_\tau || \mu_\tau\right) \mathrm{d}\tau &\leq \frac{4[\mathsf{KL}(\pi^X || \mu_0^X) - \mathsf{KL}(\pi^X || \mu_K^X)]}{\min(\eta, \delta)^2} + \frac{4K\epsilon_{\text{score}}}{\delta^2} \\ &\leq \frac{4\mathsf{KL}(\pi^X || \mu_0^X)}{\min(\eta, \delta)^2} + \frac{4K\epsilon_{\text{score}}}{\delta^2} \end{aligned}$$

where $T_K := K(t^*+1)$. The proof is concluded by dividing T_K on both sides. □

C.1.4 Discussion

To facilitate the discussion, we first present the following proposition.

Proposition C.1.5. *Define a weighting function $\lambda(\tau)$ over $\tau \in [0, T_K]$ such that for $k = 0, ..., K-1$,*

$$\lambda(\tau) = \begin{cases} \eta^2 & \text{if } \tau \in [k(t^*+1), k(t^*+1)+1], \\ v(\tau)^2 & \text{if } \tau \in [k(t^*+1)+1, (k+1)(t^*+1)]. \end{cases}$$

Then, under the same settings of Theorem 6.4.1, we have

$$\frac{1}{T_K}\int_0^{T_K} \lambda(\tau)\mathsf{FI}\left(\pi_\tau||\mu_\tau\right)\mathrm{d}\tau = \frac{4\mathsf{KL}(\pi^X||\mu_0^X)}{K(t^*+1)} + \frac{4\epsilon_{score}}{t^*+1} \tag{C.10}$$

where $\epsilon_{score} := \int_1^{t^*+1} v(\tau)^2 \mathbb{E}_{\pi_\tau}\|\boldsymbol{s}_\tau - \nabla \log p_\tau\|_2^2 \mathrm{d}\tau$.

Proof. With the definition of $\lambda(\tau)$, we can apply Lemma 2 of [336] to the likelihood steps and obtain

$$\int_{k(t^*+1)}^{k(t^*+1)+1} \lambda(\tau)\mathsf{FI}\left(\pi_\tau||\mu_\tau\right)\mathrm{d}\tau = 4[\mathsf{KL}(\pi^X||\mu_k^X) - \mathsf{KL}(\pi^Z||\mu_k^Z)] \tag{C.11}$$

for $k = 0, ..., K-1$. Similarly, we can apply Lemma C.1.4 to the prior steps and obtain

$$\int_{k(t^*+1)+1}^{(k+1)(t^*+1)} \lambda(\tau)\mathsf{FI}\left(\pi_\tau||\mu_\tau\right)\mathrm{d}\tau \le 4[\mathsf{KL}(\pi^Z||\mu_k^Z) - \mathsf{KL}(\pi^X||\mu_{k+1}^X)] + 4\epsilon_{\text{score}} \tag{C.12}$$

where

$$\begin{aligned} \epsilon_{\text{score}} &:= \int_{k(t^*+1)+1}^{(k+1)(t^*+1)} v(\tau)^2 \mathbb{E}_{\pi_\tau}\left\|\boldsymbol{s}_\tau - \nabla \log p_\tau\right\|_2^2 \mathrm{d}\tau \\ &= \int_1^{t^*+1} v(\tau)^2 \mathbb{E}_{\pi_\tau}\left\|\boldsymbol{s}_\tau - \nabla \log p_\tau\right\|_2^2 \mathrm{d}\tau. \end{aligned}$$

Together, for $\tau \in [0, T_K]$. We can then get Equation (C.10) by combining Equation (C.11) and Equation (C.12) for $k = 0, ..., K-1$ and dividing by $T_K := K(t^*+1)$. □

Unlike Theorem 6.4.1, this proposition calculates the weighted average of the Fisher divergence along the two processes with the weighting function $\lambda(\tau)$. The bound in Theorem 6.4.1 on the unweighted average of Fisher divergence can be obtained

by further lower-bounding the left-hand side of Equation (C.10) using the infimum of $\lambda(\tau)$ over $\tau \in [0, T_K]$. Given this observation, we can see the role of δ in Theorem 6.4.1. With a strictly positive δ, the weighting function $\lambda(\tau)$ is always strictly positive, so the (unweighted) average Fisher divergence must converge to 0. This is precisely the case for the VP- and VE-SDE [266]. On the other hand, if $\delta = 0$, the Fisher divergence $\mathsf{FI}(\pi_\tau || \mu_\tau)$ may be increasingly large as $\lambda(\tau)$ gets closer to 0. For iDDPM and EDM, this could happen near $t = 0$ in the reverse diffusion at $v(0) = 0$ for these diffusion processes. Nevertheless, we can instead consider a slightly adjusted diffusion coefficient $\tilde{v}(t) := v(t) + \epsilon$ with $\epsilon > 0$. Using the relation between scores and diffusions $\operatorname{div}(p \nabla \log p) = \Delta p$, we get the following reverse SDE which has the same law as Equation (6.6) at each t:

$$\mathrm{d}\boldsymbol{x}_t = \left[\frac{\dot{s}(t)}{s(t)} \boldsymbol{x}_t + \left(\frac{\epsilon^2}{2} - 2s(t)^2 \dot{\sigma}(t) \sigma(t) \right) \nabla \log p \left(\frac{\boldsymbol{x}_t}{s(t)}; \sigma(t) \right) \right] \mathrm{d}t + \left(s(t) \sqrt{2\dot{\sigma}(t)\sigma(t)} + \epsilon \right) \mathrm{d}\bar{\boldsymbol{w}}_t.$$

In this case, $\tilde{v}(t) = s(t)\sqrt{2\dot{\sigma}(t)\sigma(t)} + \epsilon$ is strictly positive, so the convergence on the unweighted average Fisher divergence is also guaranteed.

C.2 Inverse Problem Setup

Test Data Here is a summary of the test data for all experiments:

- For the synthetic prior experiment, we take images from the CelebA dataset [196], turn them into grayscale, rescale them to $[-1, 1]$, and resize them to 32×32 pixels for efficient computation. We then find the empirical mean and covariance of the images to construct the Gaussian image prior. The ground truth image is randomly drawn from this Gaussian prior.
- For the benchmark experiments, we use the first 100 images (index `00000` to `00099`) in the FFHQ dataset [152]. For all linear inverse problems, the test images are in RGB and normalized to the range $[-1, 1]$. For all nonlinear problems, the test images are in grayscale and normalized to the range $[0, 1]$.
- For the radio interferometry experiments, we generate synthetic sky images using the code[1].
- For the black hole experiments, we use the simulated data used in [269] and the publicly available EHT 2017 data[2] that was used to produce the first image of the M87 black hole.

[1] `https://github.com/liamconnor/polish-torch` (unknown license)

[2] `https://eventhorizontelescope.org/blog/public-data-release-event-horizon-telescope-2017-observations`

Gaussian and Motion Deblur The forward model is defined as

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{B}\boldsymbol{x}_0, \sigma_{\boldsymbol{y}}^2 \boldsymbol{I})$$

where $\boldsymbol{B} \in \mathbb{R}^{n \times n}$ is a circulant matrix that effectively implements a convolution with kernel $\boldsymbol{k}$ under the circular boundary condition. For the Gaussian deblurring problem, we fix the kernel $\boldsymbol{k}$ as a Gaussian kernel with standard deviation 3.0 and size 61×61. For the motion deblurring problem, we randomly generate the kernel $\boldsymbol{k}$ for each test image using the code[3] with intensity of 0.5 and size 61×61. For fair comparison, the blur kernel for each test image is set to the same for all compared methods.

Super-Resolution The forward model is defined as

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{P}_f \boldsymbol{x}_0, \sigma_{\boldsymbol{y}}^2 \boldsymbol{I}),$$

where $\boldsymbol{P}_f \in \mathbb{R}^{\frac{n}{f} \times n}$ is a matrix that implements a block averaging filter to downscale the images by a factor of f. Specifically, we set $f = 4$ and used the SVD implementation from the code[4] of [155].

Coded Diffraction Patterns (CDP) CDP is a measurement model originally proposed in [42]. The target $\boldsymbol{x}$ is illuminated by a coherent source and modulated by a phase mask $\boldsymbol{D}$. The light field then undergoes the far-field Fraunhofer diffraction and is measured by a standard camera. Mathematically, the forward model of CDP is defined as

$$\boldsymbol{y} \sim \mathcal{N}(|\boldsymbol{F}\boldsymbol{D}\boldsymbol{x}_0|, \sigma_{\boldsymbol{y}}^2 \boldsymbol{I}),$$

where $\boldsymbol{F}$ denotes the 2D Fourier transform. We follow [321] to set $\boldsymbol{D}$ as a diagonal matrix with entries drawn randomly from the complex unit circle.

Fourier Phase Retrieval We adopt a similar setting as [65]. In particular, the forward model is defined as

$$\boldsymbol{y} \sim \mathcal{N}(|\boldsymbol{F}\boldsymbol{P}\boldsymbol{x}_0|, \sigma_{\boldsymbol{y}}^2 \boldsymbol{I}),$$

where $\boldsymbol{P}$ denotes the oversampling matrix that effectively pads $\boldsymbol{x}$ in 2D matrix form with zeros. We consider a $4\times$ oversampling ratio for grayscale images of size 256×256, so $\boldsymbol{P}\boldsymbol{x}$ has a size of 512×512.

[3] `https://github.com/LeviBorodenko/motionblur` (unknown license)
[4] `https://github.com/bahjat-kawar/ddrm` (MIT license)

Black Hole Imaging We adopt the same BHI setup as in [269, 273]. Measurements for black hole imaging are obtained using Very Long Baseline Interferometry (VLBI). The cross-correlation of the recorded scalar electric fields at two telescopes, referred to as the (ideal) *visibility*, is related to the source image $\boldsymbol{x}_0$ through a Fourier transform, as given by the van Cittert-Zernike theorem [66, 339]

$$\boldsymbol{k}^t_{\{a,b\}}(\boldsymbol{x}_0) = \int_\eta \int_\delta \exp\left[-i2\pi\left(u^t_{\{a,b\}}\eta + v^t_{\{a,b\}}\delta\right)\right] \boldsymbol{x}_0(\eta,\delta)\mathrm{d}\eta\mathrm{d}\delta \in \mathbb{C}. \tag{C.13}$$

Here, (η, δ) denotes the angular coordinates of the source image, and $(u^t_{\{a,b\}}, v^t_{\{a,b\}})$ is the baseline vector between two telescopes $\{a, b\}$, orthogonal to the source direction. In practice, these measurements can be time-averaged over short intervals during which we assume that the target $\boldsymbol{x}_0$ is static. The relationship between the black hole image and each interferometric measurement, or *visibility*, is given by

$$\boldsymbol{V}^t_{\{a,b\}} = g^t_a g^t_b \cdot \exp\left[-i(\phi^t_a - \phi^t_b)\right] \cdot \boldsymbol{k}^t_{\{a,b\}}(\boldsymbol{x}_0) + \boldsymbol{n}^t_{\{a,b\}} \in \mathbb{C}, \tag{C.14}$$

where a and b denote a pair of telescopes, t represents the time of measurement acquisition, i is the imaginary unit, and $\boldsymbol{k}^t_{\{a,b\}}(\boldsymbol{x}_0)$ is the Fourier component of the target image $\boldsymbol{x}_0$ corresponding to the baseline between telescopes a and b at time t. In practice, there are three main sources of noise in Equation (C.14): gain errors g^t_a and g^t_b at the telescopes, phase errors ϕ^t_a and ϕ^t_b, and baseline-based additive white Gaussian noise $\boldsymbol{n}^t_{\{a,b\}}$. The gain and phase errors stem from atmospheric turbulence and instrument miscalibration and often cannot be ignored. To correct for these two errors, multiple noisy visibilities can be combined into data products that are invariant to these errors, which are called *closure phase* and *log closure amplitude* measurements [23, 50]

$$\begin{aligned}
\boldsymbol{y}^t_{\mathrm{cp},\{a,b,c\}} &:= \angle(\boldsymbol{V}^t_{\{a,b\}}\boldsymbol{V}^t_{\{b,c\}}\boldsymbol{V}^t_{\{a,c\}}) := \mathcal{A}^t_{\mathrm{cp},\{a,b,c\}}(\boldsymbol{x}_0) \in \mathbb{R}, \\
\boldsymbol{y}^t_{\mathrm{logca},\{a,b,c,d\}} &:= \log\left(\frac{\left|\boldsymbol{V}^t_{\{a,b\}}\right|\left|\boldsymbol{V}^t_{\{c,d\}}\right|}{\left|\boldsymbol{V}^t_{\{a,c\}}\right|\left|\boldsymbol{V}^t_{\{b,d\}}\right|}\right) := \mathcal{A}^t_{\mathrm{logca},\{a,b,c,d\}}(\boldsymbol{x}_0) \in \mathbb{R},
\end{aligned}$$

where $\angle$ computes the angle of a complex number. Given a total of M telescopes, there are in total $\frac{(M-1)(M-2)}{2}$ closure phase and $\frac{M(M-3)}{2}$ log closure amplitude measurements at time t, after eliminating repetitive measurements. In our experiments, we use a 9-telescope array ($M = 9$) from the Event Horizon Telescope (EHT) and construct the data likelihood term based on these nonlinear closure quantities. Since closure quantities are nonlinear transformations of the visibilities, the forward

model in black hole imaging becomes non-convex. Additionally, because the closure quantities do not constrain the total flux (i.e., summation of the pixel values) of the underlying black hole image, we add a constraint on the total flux defined as

$$\boldsymbol{y}_{\text{flux}} := \int_{\eta}\int_{\delta} \boldsymbol{x}_0(\eta,\delta)\mathrm{d}\eta\mathrm{d}\delta. \tag{C.15}$$

To aggregate data over time intervals and telescope combinations, the forward model of black hole imaging can be expressed as

$$\boldsymbol{y} := \left[\mathcal{A}_{\text{cp}}(\boldsymbol{x}_0), \mathcal{A}_{\text{logca}}(\boldsymbol{x}_0), \mathcal{A}_{\text{flux}}(\boldsymbol{x}_0)\right] := \left[\boldsymbol{y}_{\text{cp}}, \boldsymbol{y}_{\text{logca}}, \boldsymbol{y}_{\text{flux}}\right], \tag{C.16}$$

where $\boldsymbol{y}_{\text{cp}} = \left[\boldsymbol{y}^t_{\text{cp},\{a,b,c\}}, \forall t \in \mathcal{T}, \{a,b,c\}\right]$ is the set of all closure phase measurements and $\boldsymbol{y}_{\text{cp}} = \left[\boldsymbol{y}^t_{\text{logca},\{a,b,c,d\}}, \forall t \in \mathcal{T}, \{a,b,c,d\}\right]$ is the set of all log closure amplitude measurements over the observation period $\mathcal{T}$ and combinations of telescopes.

Overall, the potential function of the data likelihood is given by the sum of the χ^2 statistics

$$\begin{aligned} f(\boldsymbol{x}_0;\boldsymbol{y}) &= -\log p(\boldsymbol{y} \mid \boldsymbol{x}_0) \\ &= \underbrace{\frac{1}{d_{\text{cp}}\sigma^2_{\text{cp}}}\left\|\mathcal{A}_{\text{cp}}(\boldsymbol{x}_0) - \boldsymbol{y}_{\text{cp}}\right\|^2}_{\chi^2_{\text{cp}}} \\ &\quad + \underbrace{\frac{1}{d_{\text{logca}}\sigma^2_{\text{logca}}}\left\|\mathcal{A}_{\text{logca}}(\boldsymbol{x}_0) - \boldsymbol{y}_{\text{logca}}\right\|^2}_{\chi^2_{\text{logca}}} \\ &\quad + \underbrace{\frac{1}{\sigma^2_{\text{flux}}}\left\|\mathcal{A}_{\text{flux}}(\boldsymbol{x}_0) - \boldsymbol{y}_{\text{flux}}\right\|^2}_{\chi^2_{\text{flux}}} \end{aligned} \tag{C.17}$$

where σ_{cp}, σ_{logca}, and σ_{flux} are the estimated standard deviations of the measured closure phase, log closure amplitude, and flux, respectively. Additionally, d_{cp} and d_{logca} represent the total number of time intervals and telescope combinations for the closure phase and log closure amplitude measurements.

The data mismatch metric reported in Figure 6.7 is defined as the sum of χ^2_{cp} and χ^2_{logca} measurements, which are calculated using the `ehtim.obsdata.Obsdata.chisq`

function of the `ehtim` package[5]. Both χ^2 values should ideally be around 1 for data with a high signal-to-noise ratio (SNR). Therefore, a data mismatch value around 2 to 3 is considered as fitting the measurements well.

Radio Interferometry We consider a deconvolution problem in radio interferometry using the Deep Synoptic Array (DSA), following the setup in [73]. Similar to black hole imaging, each interferometric measurement is given by a pair of antennas and samples a Fourier coefficient of the underlying astronomical image. The spatial frequency sampled by each measurement depends on both the locations of the antennas with respect to each other and the observing wavelength. Given enough antennas, one could densely sample the entire spatial frequency domain. However, in practice, the array consists of a finite number of antenna pairs and operates over a limited range of radio frequencies, resulting in incomplete Fourier coverage. In the image domain, this translates to a convolution with a blur kernel defined by the point spread function (PSF) of the antenna array.

Mathematically, the forward model can be formulated as

$$\boldsymbol{y} = \boldsymbol{B}(\boldsymbol{x}_0 + \boldsymbol{\epsilon}) \tag{C.18}$$

where $\boldsymbol{B} \in \mathbb{R}^{n\times n}$ represents convolution with the PSF under circular boundary conditions, and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma_y^2 \boldsymbol{I})$ models background noise in the true sky image $\boldsymbol{x}_0$ before the signals reach the telescopes. Due to the linearity of $\boldsymbol{B}$, Equation (C.18) can be equivalently written as:

$$\boldsymbol{y} = \boldsymbol{B}\boldsymbol{x}_0 + \boldsymbol{n}$$

where $\boldsymbol{n} \sim \mathcal{N}(\mathbf{0}, \sigma_y^2 \boldsymbol{B}\boldsymbol{B}^T)$. To apply PnP-DM to this problem, we implement the likelihood step by setting the forward model as $\boldsymbol{A} = \boldsymbol{B}$ and noise covariance as $\boldsymbol{\Sigma} = \boldsymbol{B}\boldsymbol{B}^T$. One caveat is that the likelihood step involves $\boldsymbol{\Sigma}^{-1}$, which can be problematic if $\boldsymbol{B}$ is low-rank. To ensure numerical stability, we regularize the inverse computation by adding a small constant of 1e-10 to the diagonal of $\boldsymbol{\Sigma}$. In our experiments, we set $\sigma_y = 0.05$, corresponding to a signal-to-noise ratio of approximately 5, which aligns with the expected noise level in DSA-2000 observations.

[5] `https://github.com/achael/eht-imaging` (GPL-3.0 license)

C.3 Technical Details of PnP-DM

C.3.1 Likelihood Step

Linear Forward Model and Gaussian Noise As discussed in Section 6.3.1, in the case of linear forward models and Gaussian noise, the likelihood step is

$$\pi^{Z|X=\boldsymbol{x}} = \mathcal{N}(\boldsymbol{m}(\boldsymbol{x}), \boldsymbol{\Lambda}^{-1})$$

where $\boldsymbol{\Lambda} := \boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A} + \frac{1}{\eta^2}\boldsymbol{I}$ and $\boldsymbol{m}(\boldsymbol{x}) := \boldsymbol{\Lambda}^{-1}(\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{y} + \frac{1}{\eta^2}\boldsymbol{x})$. The bottleneck here is that both the mean and the covariance involve the matrix inverse $\boldsymbol{\Lambda}^{-1}$, which can be prohibitive to compute directly for high-dimensional problems. Nevertheless, the computational cost can be significantly alleviated when the noise is i.i.d. Gaussian, i.e., $\boldsymbol{\Sigma} = \sigma_{\boldsymbol{y}}^2\boldsymbol{I}$, and $\boldsymbol{A}$ can be efficiently decomposed. For example, if one can efficiently calculate the SVD of the forward model $\boldsymbol{A}$, i.e., $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T$, one can find the Cholesky decomposition of $\boldsymbol{\Lambda}^{-1}$ as

$$\boldsymbol{\Lambda}^{-1} = \boldsymbol{L}\boldsymbol{L}^T \quad \text{where} \quad \boldsymbol{L} := \boldsymbol{V}\left(\frac{1}{\sigma^2}\boldsymbol{S}^2 + \frac{1}{\eta^2}\boldsymbol{I}\right)^{-\frac{1}{2}}.$$

Since $\boldsymbol{S}$ is a diagonal matrix, the second term can be calculated with only $O(n)$ complexity. Then, leveraging the property of multivariate Gaussian distribution, we can sample $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and calculate $\boldsymbol{z} = \boldsymbol{m}(\boldsymbol{x}) + \boldsymbol{L}\boldsymbol{\epsilon}$ as a sample that exactly follows the target Gaussian distribution $\mathcal{N}(\boldsymbol{m}(\boldsymbol{x}), \boldsymbol{\Lambda}^{-1})$. An analogous derivation with the Fourier transform can be done when $\boldsymbol{A}$ is a circulant convolution matrix.

Nonlinear Forward Model We provide the pseudocode of the LMC algorithm for sampling the likelihood step with general differentiable forward models in Algorithm 5.

Algorithm 5 Langevin Monte Carlo for the likelihood step under general $\mathcal{A}$

Require: state $\boldsymbol{x}$, coupling strength $\eta > 0$, likelihood potential $f(\,\cdot\,;\boldsymbol{y})$ with measurements $\boldsymbol{y}$
Ensure: step size $\gamma > 0$, number of iterations $J > 0$
1: $\boldsymbol{u}^{(0)} \leftarrow \boldsymbol{x}$
2: **for** $j = 0, \cdots, J-1$ **do**
3: $\quad \boldsymbol{\epsilon}_j \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
4: $\quad \boldsymbol{u}^{(j+1)} \leftarrow \boldsymbol{u}^{(j)} - \gamma\nabla f(\boldsymbol{u}^{(j)};\boldsymbol{y}) - \frac{\gamma}{\eta^2}(\boldsymbol{u}^{(j)} - \boldsymbol{x}) + \sqrt{2\gamma}\boldsymbol{\epsilon}_j$
5: **end for**
6: **return** $\boldsymbol{u}^{(J)}$

Table C.1 summarizes the hyperparameters we used for solving the nonlinear inverse problems considered in this chapter.

Table C.1: **List of hyperparameters for the likelihood step of PnP-DM.**

Inverse problem	Step size (γ)	Number of iterations (J)
Coded diffraction patterns	1.0e-3	100
Fourier phase retrieval	1.0e-4	100
Black hole imaging	1.0e-5	200

C.3.2 Prior step

The EDM Framework We formally introduce the EDM formulation [151] using our notations. The forward diffusion process is defined as the following linear Itô SDE

$$\mathrm{d}\boldsymbol{x}_t = u(t)\boldsymbol{x}_t\mathrm{d}t + v(t)\mathrm{d}\boldsymbol{w}_t, \tag{C.19}$$

where $u(t) : \mathbb{R} \to \mathbb{R}$, $v(t) : \mathbb{R} \to \mathbb{R}$ are the drift and diffusion coefficients. The generative process is the time-reversed version of Equation (C.19). According to [9], it is another Itô SDE of the form

$$\mathrm{d}\boldsymbol{x}_t = \left[u(t)\boldsymbol{x}_t - v(t)^2\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t)\right]\mathrm{d}t + v(t)\mathrm{d}\bar{\boldsymbol{w}}_t, \tag{C.20}$$

where $p_t(\boldsymbol{x}_t)$ is the marginal distribution of $\boldsymbol{x}_t$. There also exists a reverse probability flow ODE

$$\mathrm{d}\boldsymbol{x}_t = \left[u(t)\boldsymbol{x}_t - \frac{1}{2}v(t)^2\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t)\right]\mathrm{d}t, \tag{C.21}$$

which shares the same marginal distributions as Equation (C.20). Based on Equation (C.19), we have

$$p(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}(s(t)\boldsymbol{x}_0, s(t)^2\sigma(t)^2\boldsymbol{I}),$$

where $s(t) := \exp\left(\int_0^t u(\xi)\mathrm{d}\xi\right)$ and $\sigma(t) := \sqrt{\int_0^t \frac{v(\xi)^2}{s(\xi)^2}\,\mathrm{d}\xi}$. We also have $\boldsymbol{x}_t/s(t) \sim p(\boldsymbol{x};\sigma(t))$ where $p(\boldsymbol{x};\sigma(t))$ is the distribution obtained by adding i.i.d. Gaussian noise of standard deviation $\sigma(t)$ to the prior data. The idea of the EDM formulation is to write the reverse diffusion directly in terms of the scaling and noise level of $\boldsymbol{x}_t$ with respect to $\boldsymbol{x}_0$, which are more important than the drift and diffusion coefficients. With the relations between $u(t)$, $v(t)$, p_t and $s(t)$, $\sigma(t)$, $p(\cdot;\sigma(t))$, we can rewrite Equation (C.20) and Equation (C.21) as

$$\mathrm{d}\boldsymbol{x}_t = \left[\frac{\dot{s}(t)}{s(t)}\boldsymbol{x}_t - 2s(t)^2\dot{\sigma}(t)\sigma(t)\nabla_{\boldsymbol{x}_t}\log p\left(\frac{\boldsymbol{x}_t}{s(t)};\sigma(t)\right)\right]\mathrm{d}t + s(t)\sqrt{2\dot{\sigma}(t)\sigma(t)}\mathrm{d}\bar{\boldsymbol{w}}_t \tag{C.22}$$

and

$$\mathrm{d}\boldsymbol{x}_t = \left[\frac{\dot{s}(t)}{s(t)}\boldsymbol{x}_t - s(t)^2\dot{\sigma}(t)\sigma(t)\nabla_{\boldsymbol{x}_t}\log p\left(\frac{\boldsymbol{x}_t}{s(t)};\sigma(t)\right)\right]\mathrm{d}t. \tag{C.23}$$

Note that Equation (C.22) is precisely the SDE 6.6 we considered for the prior step. Finally, due to the Tweedie's formula [90], we can approximate $\nabla_{\boldsymbol{x}_t}\log p\,(\,\cdot\,;\sigma(t))$ by a denoiser $[D_\theta(\,\cdot\,;\sigma(t))-\,\cdot\,]/\sigma(t)^2$ trained to minimize the ℓ_2 error of a denoising objective. Substituting the score function with the approximation by the denoiser and using the chain rule, we can further rewrite Equation (C.22) and Equation (C.23) as

$$\mathrm{d}\boldsymbol{x}_t = \left[\left(\frac{2\dot{\sigma}(t)}{\sigma(t)} + \frac{\dot{s}(t)}{s(t)}\right)\boldsymbol{x}_t - \frac{2\dot{\sigma}(t)s(t)}{\sigma(t)}D_\theta\left(\frac{\boldsymbol{x}_t}{s(t)};\sigma(t)\right)\right]\mathrm{d}t + s(t)\sqrt{2\dot{\sigma}(t)\sigma(t)}\mathrm{d}\bar{\boldsymbol{w}}_t \tag{C.24}$$

and

$$\mathrm{d}\boldsymbol{x}_t = \left[\left(\frac{\dot{\sigma}(t)}{\sigma(t)} + \frac{\dot{s}(t)}{s(t)}\right)\boldsymbol{x}_t - \frac{\dot{\sigma}(t)s(t)}{\sigma(t)}D_\theta\left(\frac{\boldsymbol{x}_t}{s(t)};\sigma(t)\right)\right]\mathrm{d}t. \tag{C.25}$$

Pseudocode We provide the pseudocode for our prior step in Algorithm 6. Note that the update rule is precisely the Euler discretization of Equation (C.24) and Equation (C.25). The discretization time steps $\{t_i\}_{i=0}^N$, scaling schedule $s(\cdot)$, and noise schedule $\sigma(\cdot)$, are kept the same as in Table 1 of [151]. For all experiments, we set the total number of time steps to 100, i.e., $N = 100$. We note that this does not imply that the number of function evaluations (NFE) of each prior step is 100. Since η is to a small value as the algorithm runs, the number of steps in later iterations of the algorithm is much fewer than 100. The prior step is similar to the image synthesis process in SDEdit [218] that starts from the middle of the reverse diffusion process. We use the `pf-ODE` solver for the CDP problem and the `SDE` solver for all other problems. Part of our code implementation is based on the repository[6].

Model Checkpoint For experiments with FFHQ color images, we use the pre-trained checkpoint from [61] available at the repository[7]. For experiments with synthetic data, FFHQ grayscale images, and black hole images, we train our own models using the same repository. The model network is based on the U-Net

[6] `https://github.com/NVlabs/edm/tree/main` (Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International Public license)

[7] `https://github.com/jychoi118/P2-weighting` (MIT license)

Algorithm 6 EDM for the prior step (Bayesian denoising with noise level η)

Require: noisy image $\boldsymbol{z} \in \mathbb{R}^n$, assumed noise level $\eta > 0$, pre-trained model $D_\theta(\,\cdot\,;\,\cdot\,)$ that approximates $\nabla \log p\,(\boldsymbol{x};\sigma)$ with $(D_\theta(\boldsymbol{x};\sigma) - \boldsymbol{x})/\sigma^2$

Ensure: discretization time steps $\{t_i\}_{i=0}^N$ (monotonically decreasing to $t_N = 0$), scaling schedule $s(\cdot)$, noise schedule $\sigma(\cdot)$, solver (`SDE` or `pf-ODE`)

$i^* \leftarrow \min\{i \in [N] \mid \sigma(t_i) \le \eta\}$ ▷ Find the starting point of the reverse diffusion

$\boldsymbol{v}^{(i^*)} \leftarrow s(t_{i^*})\boldsymbol{z}$ ▷ Initialize at time t_{i^*}

for $i = i^*, \cdots, N-1$ **do**

  $\lambda \leftarrow 2$ **if** solver is `SDE` **else** 1

  $\boldsymbol{d}_i \leftarrow \left(\frac{\lambda\dot\sigma(t_i)}{\sigma(t_i)} + \frac{\dot s(t_i)}{s(t_i)}\right)\boldsymbol{v}_i - \frac{\lambda\dot\sigma(t_i)s(t_i)}{\sigma(t_i)} D_\theta\left(\frac{\boldsymbol{v}_i}{s(t_i)}; \sigma\,(t_i)\right)$

  $\boldsymbol{v}^{(i+1)} \leftarrow \boldsymbol{v}^{(i)} + (t_{i+1} - t_i)\boldsymbol{d}_i$ ▷ Drift

  if $i \neq N-1$ **and** solver is `SDE` **then**

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

    $\boldsymbol{v}^{(i+1)} \leftarrow \boldsymbol{v}^{(i+1)} + s(t)\sqrt{2\dot\sigma(t)\sigma(t)(t_i - t_{i+1})}\boldsymbol{\epsilon}_i$ ▷ Diffusion

  end if

end for

return $\boldsymbol{v}^{(N)}$

architecture in [224] with BigGAN [36] residual blocks, multi-resolution attention, and multi-head attention with fixed channels per head. See the appendix of [61] for architecture details. Specifically, we change the input and output channels to 1 and 2, respectively, to accommodate grayscale inputs, and reduce the number of down-pooling and up-pooling levels in the U-Net for smaller images. We train all models until convergence using an exponential moving average (EMA) rate of 0.9999, 32-bit precision, and the AdamW optimizer [197]. Here is a list of training data we use for each model:

- For the Gaussian prior model, we randomly generate images from the constructed Gaussian prior distribution.
- For the FFHQ grayscale model, we use the images with index `01000` to `69999` in the FFHQ dataset.
- For the black hole model, we use 3,068 simulated black hole images from the GRMHD simulation, which stands for *general relativistic magnetohydrodynamic* simulation [72]. See Figure C.1 for some example training images. We apply data augmentation with random flipping and resizing, so that the flux spin rotation and the ring diameter vary from sample to sample.

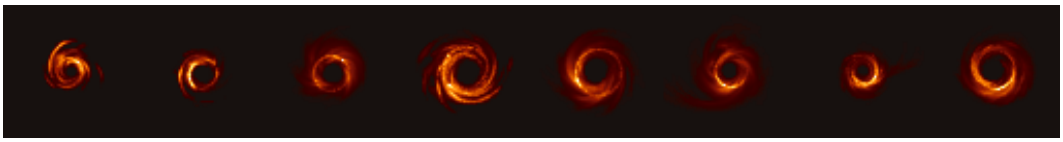


Figure C.1: **Example images from the dataset for training the black hole diffusion model prior.**

Preconditioning Since the checkpoints we use are all trained based on the DDPM (or VP-SDE) formulation [131], we convert them to the denoiser D_θ under the EDM formulation via the VP preconditioning [151]. Specifically, if we denote the pre-trained model as $F_\theta(\,\cdot\,;\,\cdot\,)$, the model we use for Algorithm 6 is

$$D_\theta(\boldsymbol{x};\sigma) := c_{\text{skip}}(\sigma)\boldsymbol{x} + c_{\text{out}}(\sigma)F_\theta\left(c_{\text{in}}(\sigma)\boldsymbol{x}; c_{\text{noise}}(\sigma)\right), \tag{C.26}$$

where $c_{\text{skip}}(\sigma) = 1, c_{\text{out}}(\sigma) = -\sigma, c_{\text{in}}(\sigma) = 1/\sqrt{\sigma^2+1}$, and $c_{\text{noise}}(\sigma) = 999\sigma_{\mathsf{VP}}^{-1}(\sigma)$. Here $\sigma_{\mathsf{VP}}^{-1}(\cdot)$ is the inverse of the VP-SDE noise schedule defined as $\sigma_{\mathsf{VP}}(t) := \sqrt{e^{\frac{1}{2}\beta_{\text{d}}t^2+\beta_{\min}t}-1}$ with $\beta_{\text{d}} = 19.9$ and $\beta_{\min} = 0.1$. This adaptation allows us to make a fair comparison with other DM-based methods using the same pre-trained models. One can also incorporate DMs trained with other formulations into PnP-DM by properly setting the preconditioning parameters.

Connection to `DDS-DDPM` in [326] A concurrent work [326] introduced a rigorous implementation of the prior step, called `DDS-DDPM`, by converting the DDPM [131] (or VP-SDE [266]) sampler into a reverse diffusion based on the VE-SDE [266]. The diffusion process after the conversion can be used to solve Equation (6.5) rigorously by properly choosing the starting point. In fact, our formulation admits `DDS-DDPM` as a special case with the VP preconditioning and reverse diffusion based on the VE-SDE. Here we explicitly show this connection. For the VE-SDE, we have $s_{\mathsf{VE}}(t) = 1$, $\sigma_{\mathsf{VE}}(t) = \sqrt{t}$, $u_{\mathsf{VE}}(t) = 0$, and $v_{\mathsf{VE}}(t) = 1$. So Equation (C.24) becomes

$$\mathrm{d}\boldsymbol{x}_t = \left[\frac{1}{t}\boldsymbol{x}_t - \frac{1}{t}D_\theta\left(\boldsymbol{x}_t;\sqrt{t}\right)\right]\mathrm{d}t + \mathrm{d}\bar{\boldsymbol{w}}_t. \tag{C.27}$$

Applying the VP preconditioning Equation (C.26) to Equation (C.27), we obtain

$$\mathrm{d}\boldsymbol{x}_t = \left[\frac{1}{\sqrt{t}}F_\theta\left(\frac{\boldsymbol{x}_t}{\sqrt{t+1}}; 999\sigma_{\mathsf{VP}}^{-1}(\sqrt{t})\right)\right]\mathrm{d}t + \mathrm{d}\bar{\boldsymbol{w}}_t. \tag{C.28}$$

We can then rescale the time range from $[0,1]$ to $[0,1000]$, discretize Equation (C.28) backward in time over the time steps $\{\tau_t\}$ from [326], and apply the

exponential integrator [348] to the drift term, resulting in the following update rule:

$$\widehat{\boldsymbol{x}}_{t-1} = \widehat{\boldsymbol{x}}_t - 2(\sqrt{\tau_t} - \sqrt{\tau_{t-1}})F_\theta\left(\frac{\widehat{\boldsymbol{x}}_t}{\sqrt{\tau_t + 1}}; \sigma_{\mathsf{VP}}^{-1}(\sqrt{\tau_t})\right) + \sqrt{\tau_t - \tau_{t-1}}\boldsymbol{\epsilon}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Based on the definition $\tau_t := \bar{\alpha}_t^{-1} - 1 = \sigma_{\mathsf{VP}}(t)^2$ in `DDS-DDPM`, we get

$$\widehat{\boldsymbol{x}}_{t-1} = \widehat{\boldsymbol{x}}_t - 2(\sqrt{\tau_t} - \sqrt{\tau_{t-1}})F_\theta\left(\sqrt{\bar{\alpha}_t}\widehat{\boldsymbol{x}}_t; t\right) + \sqrt{\tau_t - \tau_{t-1}}\boldsymbol{\epsilon}. \tag{C.29}$$

This is exactly the update rule of `DDS-DDPM` with $F_\theta(\,\cdot\,; t)$ denoting the noise estimate $\widehat{\epsilon}_t(\cdot)$ of DDPM. One can also verify that the initialization in `DDS-DDPM` is equivalent to ours by checking that $\bar{\alpha}_t \geq \frac{1}{\eta^2+1}$ is equivalent to $\tau_t = \sigma_{\mathsf{VP}}(t)^2 \leq \eta^2$ where $\eta \equiv \eta$ is the assumed noise level in Equation (6.5). As one can see, `DDS-DDPM` is equivalent to our prior step by choosing the VP-preconditioning, VE reverse diffusion, and a particular integration scheme. In fact, our prior step allows for more general definitions of diffusion processes and includes both the ODE and SDE solvers.

C.3.3 Others

Annealing Schedule for η In this chapter, we consider an exponential annealing schedule[8] for the coupling strength η. By specifying a starting level η_0, decay rate α, and a minimum value $\eta_{\min}$, we set

$$\eta_k = \max(\alpha^k \eta_0, \eta_{\min})$$

for $k = 0, \cdots, K-1$. Table C.2 summarizes the annealing hyperparameters that we use for all the inverse problems considered in this chapter.

Table C.2: **List of hyperparameters for the annealing schedule of η in PnP-DM.**

Inverse problem	Starting level (η_0)	Minimum level ($\eta_{\min}$)	Decay rate (α)
Synthetic prior experiments	0.03	0.03	1
Gaussian deblur	10	0.3	0.9
Motion deblur	10	0.3	0.9
Super-resolution	10	0.3	0.9
Coded diffraction patterns	10	0.1	0.9
Fourier phase retrieval	10	0.1	0.9
Radio interferometry	0.1	0.1	1
Black hole imaging	10	0.02	0.93

[8] PnP-DM is compatible with more general schedules, such as linear decay and parabolic decay.

Initialization For the linear inverse problems, including radio interferometry, we use the zero initialization, i.e., $\boldsymbol{x}^{(0)} = \mathbf{0} \in \mathbb{R}^n$. For the CDP and Fourier phase retrieval problems, we use the Gaussian initialization, i.e., $\boldsymbol{x}^{(0)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. For black hole imaging experiments, we use the uniform random initialization between 0 and 1 for each pixel. We find that PnP-DM, as an MCMC algorithm, is insensitive to the initialization. Except for the black hole experiments, where we found the negative values would cause problems, any reasonable initialization would lead to comparable results. This observation corroborates our convergence result, which holds for any initialization μ_0^X.

Number of Iterations We run 500 iterations for the synthetic prior experiments, 200 iterations for the black hole experiments, and 100 iterations for all other experiments. The numbers were chosen so that the algorithm was fully converged.

Sample Collection To collect multiple samples using our method, there are two main approaches: (1) Run a single Markov chain and collect samples after a certain number of iterations, known as the burn-in period, to ensure the chain has converged. (2) Run several independent Markov chains and collect one sample from each chain after convergence. The first approach is more efficient, but the collected samples are not entirely independent and thus may have a small effective sample size. The second approach ensures all samples are fully independent but takes longer to run. In our experiments, we use the first approach for all tests involving 256×256 images to enhance efficiency. Specifically, we set the burn-in period to 40 iterations and collect 20 random samples from the remaining 60 iterations (one every 3 iterations). For other experiments, due to the smaller image sizes, we employ the second approach to obtain fully independent samples.

Compute All experiments were performed on NVIDIA RTX A6000 and A100 GPUs. The runtime per image depends on several factors, such as the choice of GPU, the total number of iterations and the coupling strength schedule $\{\eta_k\}$ (as it takes more network evaluations for larger η for our EDM-based denoiser). In our actual experiments, we ran each image for at least 100 iterations to ensure convergence, which took around 1 minute for a single Markov chain. Here we present a comparison of computational efficiency with the major baselines on a linear super-resolution and a nonlinear coded diffraction patterns problem in Table C.3. The clock time in seconds and the number of function evaluations (NFE) are calculated

for each method to measure its computational efficiency. All hyperparameters are kept the same for each method as those used for Table 6.1 and Table 6.2 in the manuscript. As expected, DM-based approaches (DDRM & DPS) generally yield shorter runtimes due to their lower NFEs. Nevertheless, our PnP-DM method significantly outperforms these methods while achieving comparable runtimes with DPS ($\approx 1.5\times$), despite its larger NFEs ($\approx 3\times$). This is primarily due to two factors: (1) PnP-DM avoids running the full diffusion process by adapting the starting noise level to η_k at each iteration, and (2) the runtime is further reduced by using an annealing schedule of η_k. We also note that the runtime reported for DDRM and DPS below is the time it takes to generate one sample. For the linear inverse problem experiments, where we generate 20 samples for each sampling method, PnP-DM is faster than DPS because we take 20 samples that PnP-DM generated along one Markov chain of batch size 1 (hence same runtime as below, around 50 seconds) but DPS requires running a diffusion process with batch size 20, which is significantly slower (around 330 seconds).

Table C.3: **Comparison of computational efficiency between PnP-DM and other baseline methods.**

Inverse problem	Metric	DDRM	DPS	PnP-SGS	DPnP	PnP-DM (ours)
Super-resolution	Clock time (s)	0.4	39	20	322	55
	NFE	20	1,000	1,030	18,372	3,032
Coded diffraction patterns	Clock time (s)	–	37	54	261	50
	NFE	–	1,000	2,572	14,596	2,482

C.4 Implementation Details of Baseline Methods

PnP-ADMM We set the ADMM penalty parameter as 2 and run for 500 iterations to ensure convergence. We use the pre-trained DnCNN denoiser [346] available at the `deepinv` library[9]. An additional batch dimension is introduced to collect multiple samples.

DPIR We follow the annealing schedule in [345] and run for 40 iterations. We use the pre-trained DRUNet denoiser [347] available at the `deepinv` library. An additional batch dimension is introduced to collect multiple samples.

DDRM We adopt the default hyperparameters: $\eta_B = 1.0$, $\eta = 0.85$, and 20 steps for the DDIM sampler [261]. For the Gaussian deblur problem, we use the

[9] `https://github.com/deepinv/deepinv` (BSD-3-Clause license)

SVD-based forward model implementation based on separable 1D convolution. An additional batch dimension is introduced to collect multiple samples.

DPS We follow the original paper to use a 1000-step DDPM sampler backbone. For the linear inverse problems, we use the step size given in [64], i.e., $\zeta' = 1$. For the nonlinear inverse problems, we optimized the step size ζ' by performing a grid search, which led to $\zeta' = 3$ for CDP and Fourier phase retrieval and $\zeta' = 0.001$ for black hole imaging. For the synthetic prior experiments, we also optimized the step size and obtained $\zeta' = 0.1$ for compressed sensing and $\zeta' = 1$ for Gaussian deblur. An additional batch dimension is introduced to collect multiple samples.

PnP-SGS We performed a grid search for the coupling parameter η and found that $\eta = 0.1$ worked the best for all problems. We follow the practice in [69] to have a burn-in period of 20 iterations during which the reverse diffusion is early-stopped. We run the algorithm for 100 iterations in total and collect 20 samples in the 80 iterations after the burn-in period.

DPnP We implement the `DDS-DDPM` sampler for the prior step. For fair comparison, we use the same annealing schedule for the coupling strength (denoted as η_k in [326]) as PnP-DM. We run the algorithm for the same number of iterations for each inverse problem in the same way of collecting samples as PnP-DM.

HIO We set $\beta = 0.7$ and applied both the non-negative constraint and the finite support constraint. To mitigate the instability of reconstruction depending on initialization, we first repeatedly run the algorithm with 100 different random initializations and choose the reconstruction with the best measurement fit. Then we run another 10,000 iterations with the chosen reconstruction to ensure convergence and report the metrics on the final reconstruction.

C.5 Additional Related Works

Image Reconstruction with Plug-and-Play Priors Plug-and-Play priors (PnP) [291] is an algorithmic framework that leverages off-the-shelf denoisers for solving imaging inverse problems. Recognizing the equivalence between the proximal operator and finding the *maximum a posteriori (MAP)* solution to a denoising problem, PnP substitutes the proximal update in many optimization algorithms, such as ADMM [52, 252] and HQS [345, 347], with generic denoising algorithms, particularly those based on deep learning [217, 345, 347]. The PnP framework enjoys both

convergence guarantees [252, 272] and strong empirical performance [4, 322] due to its compatibility with state-of-the-art learning-based denoising priors. Recent works have also proposed learning-based PnP frameworks that have direction interpretations from an optimization perspective [70, 99]. See [149] for a comprehensive review on the theory and practice of PnP.

Posterior Sampling with MCMC and Learned Priors Learning-based priors have also been considered in the Bayesian context, where one seeks to sample the posterior distribution defined under a learned prior. An important technique is denoising score matching (DSM) [292], which connects image denoising with learning the score function of an image distribution. Based on DSM, prior works have incorporated deep denoising priors into MCMC formulations, particularly focusing on the Langevin Monte Carlo and its variants as they involve the score function of the target distribution [145, 156, 170, 244, 273]. Recently, methods based on SGS have also gained increasing popularity [31, 69, 100, 230, 326]. Unlike PnP methods based on optimization, these sampling methods possess the ability to generate diverse solutions and quantify the uncertainty of the solution space.

Solving Inverse Problems with Diffusion Models The remarkable performance of diffusion models [131, 266] on modeling image distributions makes them desirable choices as image priors for solving inverse problems. One popular approach is to leverage a pre-trained unconditional model and modify the reverse diffusion process during inference to enforce data consistency [34, 64, 65, 155, 189, 260, 262, 263, 265, 306, 361]. Despite the promising performance of these methods, they usually involve approximations and empirically driven designs that are hard to justify theoretically and may lead to inconsistent sample distributions. Another line of work learns task-specific models, which achieves higher accuracy at the cost of re-training models for new problems [17, 189, 253]. Methods based on Particle Filtering and Sequential Monte Carlo are also considered to ensure asymptotic consistency [44, 89, 318]. Diffusion models have also been considered as a prior for variational inference [103, 214] and pluy-and-play image reconstruction [119, 215].

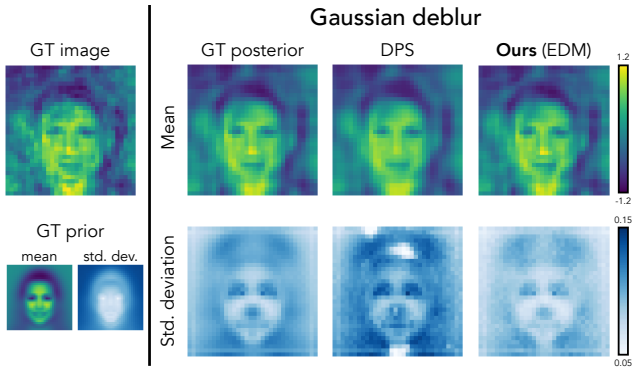


Figure C.2: **Comparison of our method and DPS [64] on estimating the posterior distribution of a Gaussian deblurring problem under a Gaussian prior.** While the mean estimations of the two methods are of roughly the same quality, our approach provides a much more accurate estimation of the posterior per-pixel standard deviation than DPS.

C.6 Additional Experimental Results

C.6.1 Synthetic Prior Experiment

In addition to the compressed sensing experiment presented in the main paper, we show another comparison on a Gaussian deblurring problem in Figure C.2. Here, the linear forward model $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ is a 2D convolution matrix with a Gaussian blur kernel of size 7×7 and standard deviation 3.0. Similar to the compressed sensing experiment, both methods yield accurate reconstructions of the mean. However, in terms of the posterior standard deviation, DPS exhibits a notable difference from the ground truth, whereas our method achieves a significantly more accurate result.

C.6.2 Linear Inverse Problems

We provide visual comparisons for the Gaussian deblur and super-resolution problems in Figure C.3.

Additional visual examples are provided in Figure C.4 (Gaussian deblurring), Figure C.5 (motion deblurring), and Figure C.6 (super-resolution).

C.6.3 Nonlinear inverse problems

We provide visual comparisons for the CDP reconstruction problem in Figure C.7, where we visualize one sample for each method. As shown by the red zoom-in boxes, PnP-DM can recover fine-grained features such as the hair threads that are missing in the reconstructions by the baselines. Additional reconstruction examples are given in Figure C.8 for the CDP reconstruction problem.

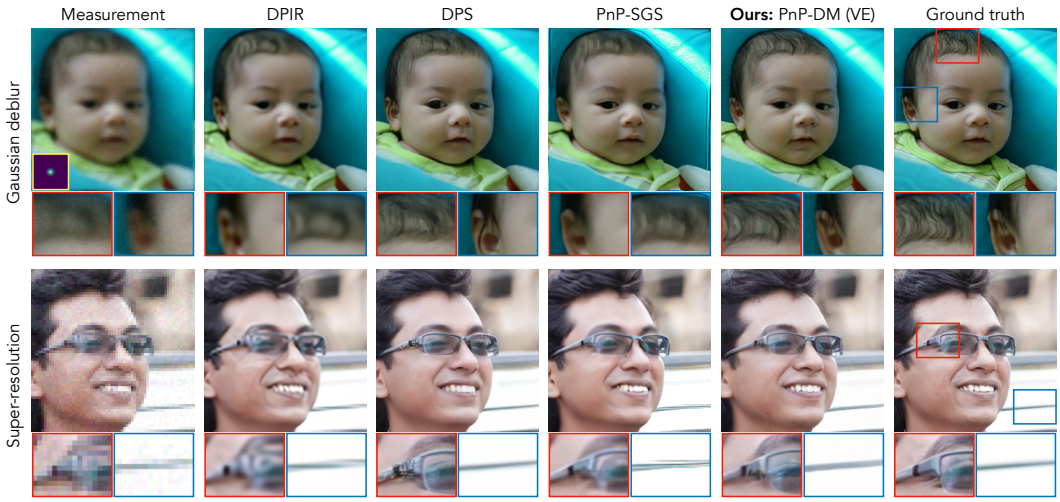


Figure C.3: **Visual comparison between our method and baselines on solving the Gaussian deblurring and super-resolution problems with i.i.d. Gaussian noise** ($\sigma_y = 0.05$)**.** We visualize one sample generated by each algorithm.

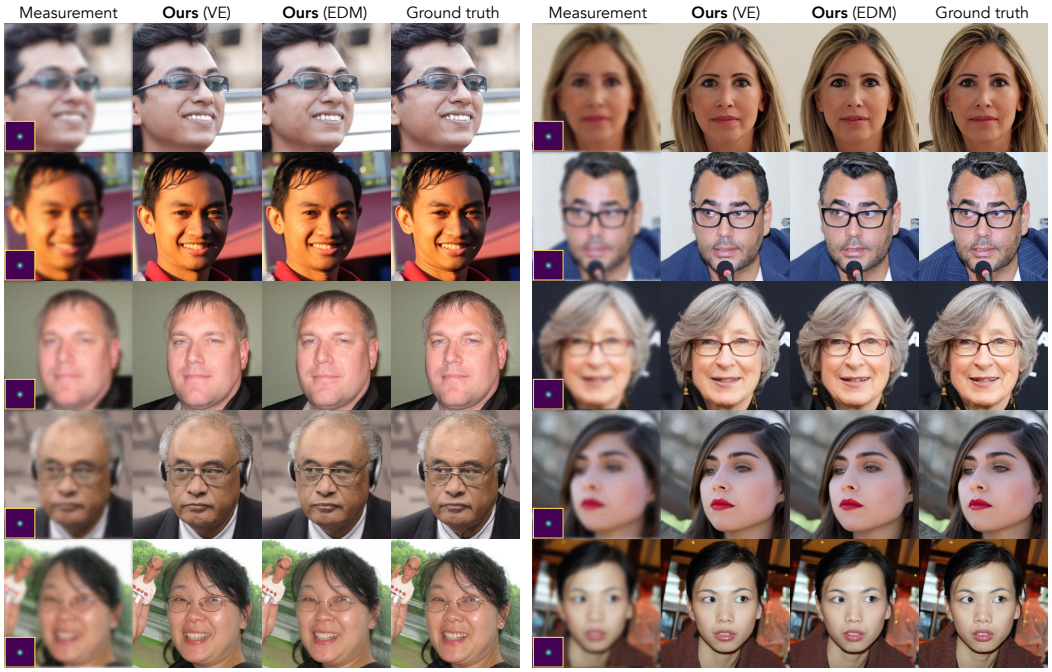


Figure C.4: **Additional visual examples for the Gaussian deblurring problem.**

We then show some additional reconstruction examples with comparison to DPS in Figure C.9. For each method, we visualize the best reconstruction out of four runs for each test image according to the PSNR value. While DPS failed on around half of the test images, our proposed method provided high-fidelity reconstructions on almost all test images. This comparison highlights the better robustness of our method over DPS.

Figure C.5: **Additional visual examples for the motion deblurring problem.**

C.6.4 Black Hole Imaging

Finally, we present visual examples from the black hole imaging experiments. Samples generated by PnP-DM (EDM) and DPS using the simulated data are shown in Figure C.10, with the data mismatch metric labeled at the top right corner of each sample. Consistent with the results in Figure 6.7, DPS can only capture one of the two posterior modes. DPS samples from Mode 2 and Mode 3 significantly deviate from the measurements and lack the expected black hole structure. In contrast, PnP-DM successfully samples both posterior modes and consistently produces samples that fit the measurements well. Additionally, Figure C.11 presents more samples obtained by applying PnP-DM to the real M87 black hole data. The generated samples are not only diverse but also fit the measurements well, with data mismatch values around 2. These samples exhibit a ring diameter consistent with the official EHT reconstruction in Figure 6.8 and share a common bright spot location at the lower half of the ring.

C.6.5 Further Analysis

Sensitivity Analysis on the Annealing Schedule $\{\eta_k\}$ In Figure C.12, we present a sensitivity analysis on the annealing schedule $\{\eta_k\}$. In particular, we show the PSNR curves of $\boldsymbol{x}_k$ with different exponential decay rates α (left) and minimum coupling levels $\eta_{\min}$ (right) for one linear (super-resolution) and one nonlinear

Figure C.6: **Additional visual examples for the 4× super-resolution problem.**

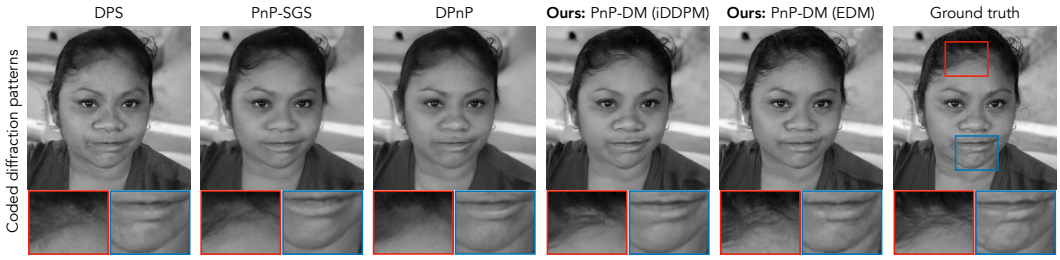


Figure C.7: **Visual comparison between our method and baselines on solving the coded diffraction pattern (CDP) reconstruction problems with i.i.d. Gaussian noise ($\sigma_y = 0.05$).** We visualize one sample generated by each algorithm.

(coded diffraction patterns) problem. We have the following conclusions based on the results. First, different decay rates lead to different rates of convergence, which corroborates our theoretical insights that η plays the same role as the step size. The final level of PSNR is not sensitive to different decay rates, as all curves converge to the same level. Second, as $\eta_{\min}$ decreases, the final PSNR becomes higher. This is as expected because the stationary distribution of the $\boldsymbol{x}_k$, π^X, should converge to the true target posterior, $p(\boldsymbol{x}|\boldsymbol{y})$, as η decreases.

Convergence Curves with Intermediate Visual Examples In Figure C.13, we show some visual examples of intermediate $\boldsymbol{x}_k$ and $\boldsymbol{z}_k$ iterates (left) and convergence plots of PSNR, SSIM, and LPIPS for $\boldsymbol{x}_k$ (right) on the super-resolution problem.

Figure C.8: **Additional visual examples for the Fourier phase retrieval problem.**

As η_k decreases, $\boldsymbol{x}_k$ becomes closer to the ground truth and $\boldsymbol{z}_k$ gets less noisy. Both the visual quality and metric curves stabilize after the minimum coupling strength is achieved. Despite being run for 100 iterations in total, our method generates good images in around 40 iterations, which is around 30 seconds and 1600 NFEs.

Figure C.9: **Additional visual examples for the Fourier phase retrieval problem.**

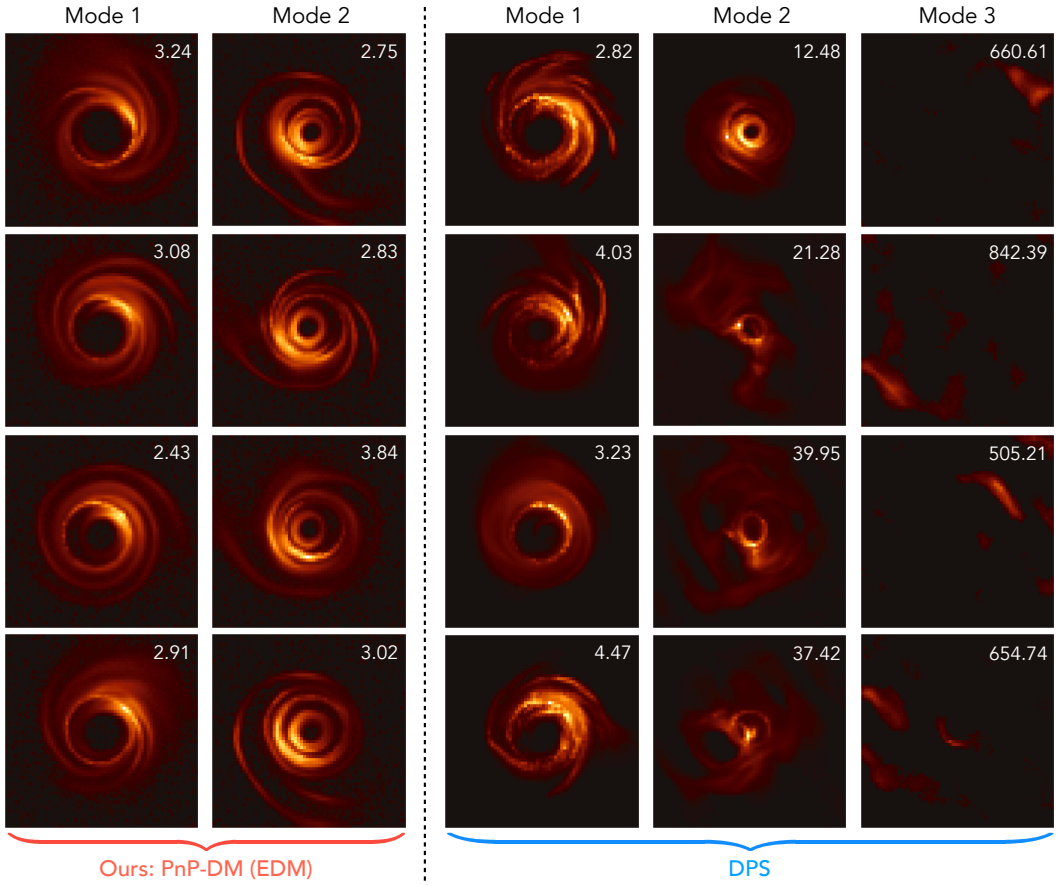


Figure C.10: **Additional visual examples given by PnP-DM and DPS using the simulated black hole data.**

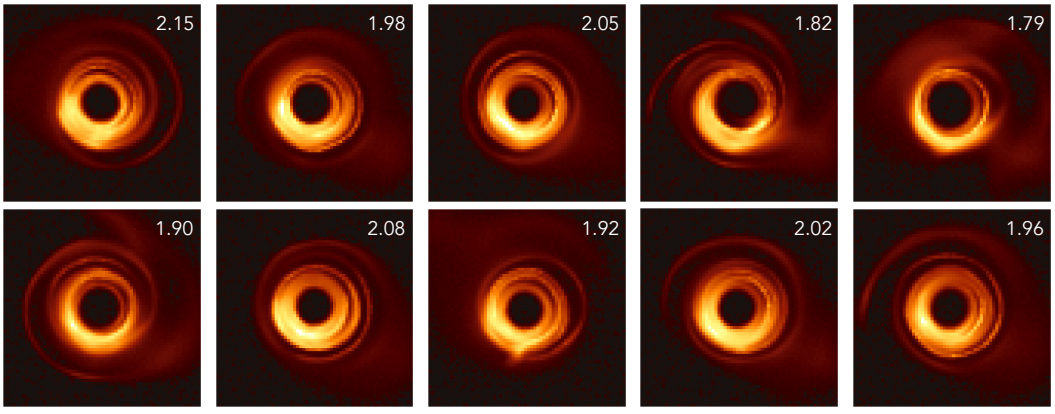


Figure C.11: **Additional visual examples given by PnP-DM using the real M87 black hole data.**

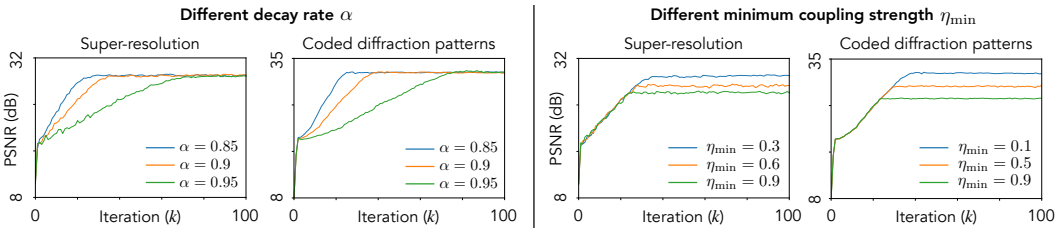


Figure C.12: **Sensitivity analysis on the annealing schedule η_k with different decay rates α (left) and minimum coupling strength $\eta_{\min}$ (right) for a linear (super-resolution) and a nonlinear (coded diffraction patterns) inverse problem.** Recall from Appendix C.3.3 that $\eta_k := \max(\alpha^k \eta_0, \eta_{\min})$, where we set $\eta_0 = 10$ for this experiment.

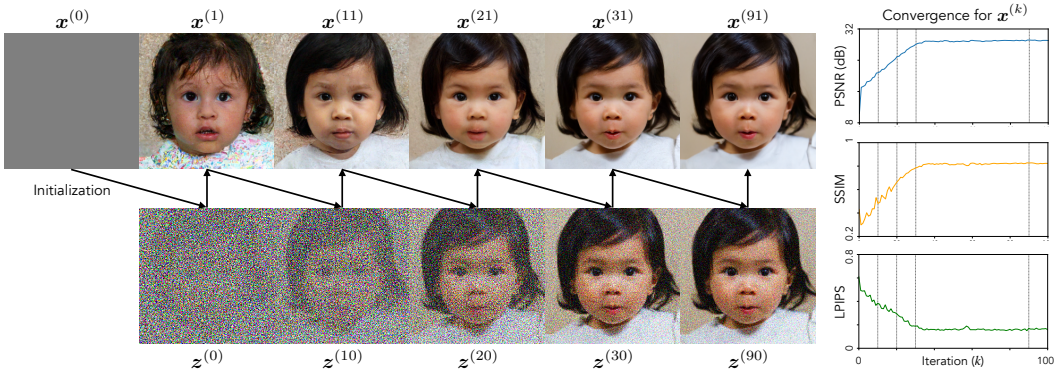


Figure C.13: **Visual examples of intermediate $\boldsymbol{x}^{(k)}$ and $\boldsymbol{z}^{(k)}$ iterates (left) and convergence plots of PSNR, SSIM, and LPIPS for $\boldsymbol{x}^{(k)}$ iterates (right) on the super-resolution problem.** The vertical dashed lines show the iterations at which the $\boldsymbol{x}^{(k)}$ and $\boldsymbol{z}^{(k)}$ iterates are visualized.

Appendix D

APPENDIX FOR CHAPTER 7

D.1 Appendix for Section 7.3

D.1.1 DCDP, PnP-DM, and DAPS as Instantiations of Algorithm 2

D.1.1.1 DCDP

Likelihood Step The likelihood step with LDMs is given in the "Approach I: Enforcing DC in the Latent Space" section from the DCDP paper [181]. Switching this to the notation in this paper, we have

$$\boldsymbol{v}^{(k)} = \arg\min_{\boldsymbol{v}} \frac{1}{2}\|\mathcal{A}(\mathcal{D}(\boldsymbol{v})) - \boldsymbol{y}\|_2^2 + \frac{\mu}{2}\|\boldsymbol{u}^{(k)} - \boldsymbol{v}\|_2^2 \tag{D.1}$$

for some constant μ. Setting $\mu = \sigma_{\boldsymbol{y}}^2/\eta_k^2$ makes this equivalent to

$$\begin{aligned}\boldsymbol{v}^{(k)} &= \arg\min_{\boldsymbol{v}} \frac{1}{2\sigma_{\boldsymbol{y}}^2}\|\mathcal{A}(\mathcal{D}(\boldsymbol{v})) - \boldsymbol{y}\|_2^2 + \frac{1}{2\eta_k^2}\|\boldsymbol{u}^{(k)} - \boldsymbol{v}\|_2^2 \\ &= \arg\max_{\boldsymbol{v}} \exp\left(-\frac{1}{2\sigma_{\boldsymbol{y}}^2}\|\mathcal{A}(\mathcal{D}(\boldsymbol{v})) - \boldsymbol{y}\|_2^2 - \frac{1}{2\eta_k^2}\|\boldsymbol{u}^{(k)} - \boldsymbol{v}\|_2^2\right)/Z. \end{aligned} \tag{D.2}$$

Note that Equation (D.2) is exactly finding the MAP estimation of Equation (7.6) with $\eta_k = \sigma_{\boldsymbol{y}}/\sqrt{\mu}$. This shows the equivalence of sub-step 1.

In sub-step 2, noise is added as the first half of the Diffusion Purification step of DCDP. DCDP runs forward diffusion up to an annealed timestep T_k, which is equivalent to adding η_k noise for the correct choice of η_k.

Prior Step In the second half of the Diffusion Purification step of DCDP, reverse diffusion is run using the VP-SDE formulation. This is the same as reverse diffusion in the EDM framework with $s_t = 1/\sqrt{e^{\frac{1}{2}\beta_d t^2 + \beta_{\min} t}}$ and $\sigma_t = \sqrt{e^{\frac{1}{2}\beta_d t^2 + \beta_{\min} t} - 1}$ [151]. Since this formulation is representable in the EDM framework, it is equivalent to Equation (7.4).

D.1.1.2 PnP-DM

Likelihood Step The likelihood step in PnP-DM in Chapter 6, originally designed for pixel-space diffusion, introduces a hyperparameter η with exponential decay

$\eta_k = \max(\alpha^k \eta_0, \eta_{\min})$, where α, η_0, and $\eta_{\min}$ are preset. Extending this to latent diffusion amounts to replacing $\mathcal{A}$ with $\mathcal{A}(\mathcal{D}(\cdot))$, leading to

$$\boldsymbol{v}^{(k)} \sim \exp\left(-\frac{1}{2\sigma_{\boldsymbol{y}}^2}\|\mathcal{A}(\mathcal{D}(\boldsymbol{v})) - \boldsymbol{y}\|_2^2 - \frac{1}{2\eta_k^2}|\boldsymbol{u}^{(k)} - \boldsymbol{v}\|_2^2\right)/Z. \tag{D.3}$$

This likelihood step matches the distribution in Equation (7.6). No additional noise is introduced in sub-step 2.

Prior Step The prior step of PnP-DM utilizes various diffusion schedulers, such as VP-SDE and EDM-SDE. By adopting the EDM scheduler with $s_t = 1$ and $\sigma_t = t$, the prior step matches the form of Equation (7.4).

D.1.1.3 DAPS

Likelihood Step In DAPS [341], the likelihood and prior steps are coupled, as sampling $\boldsymbol{x}_0 \sim p(\boldsymbol{x}_0 \mid \boldsymbol{x}_t, \boldsymbol{y})$ is required. However, by adopting the Gaussian approximation $p(\boldsymbol{x}_0 \mid \boldsymbol{x}_t) \approx \mathcal{N}(\widehat{\boldsymbol{x}}_0(\boldsymbol{x}_t), \sigma_t^2 \boldsymbol{I})$ (where $\widehat{\boldsymbol{x}}_0(\boldsymbol{x}_t)$ is obtained via a few-step ODE solver), separation becomes possible. This approximation aligns with the prior step formulation in Equation (7.4).

With $\boldsymbol{u}^{(k)} = \widehat{\boldsymbol{x}}_0(\boldsymbol{x}_t)$, the DAPS update rule is:

$$\boldsymbol{v}^{(k)} \sim \exp\left(-\frac{1}{2\beta_{\boldsymbol{y}}^2}\|\mathcal{A}(\mathcal{D}(\boldsymbol{v})) - \boldsymbol{y}\|_2^2 - \frac{1}{2r_t^2}\|\boldsymbol{u}^{(k)} - \boldsymbol{v}\|_2^2\right)/Z, \tag{D.4}$$

where $\beta_{\boldsymbol{y}}$ and r_t are hyperparameters. Setting $\beta_{\boldsymbol{y}} = \sigma_{\boldsymbol{y}}$ and $r_t = \eta_k$ recovers the likelihood step in Equation (7.6).

Prior Step The prior step in DAPS follows from setting $\boldsymbol{u}^{(k)} = \widehat{\boldsymbol{x}}_0(\boldsymbol{x}_t)$ via a few-step ODE solver using the EDM scheduler. Choosing an appropriate η_k makes this sampling procedure equivalent to that described by Equation (7.4).

D.1.2 Hyperparameters

To optimize hyperparameters, we run two Bayesian optimization processes for each task to select the combinations that had the highest PSNR and LPIPS, respectively. These caused different forms of artifacts in the results, so the parameters that yielded the best qualitative outputs are chosen. TReg, PSLD, and PnP-DM are optimized for LPIPS while DCDP is optimized for PSNR. For DAPS, neither of these optimizations leads to results better than previously handpicked parameters, so those are used instead.

Table D.1: **Hyperparameters used by each method on each task.**

Method	Task	Super-resolution (16×)	Box inpainting	Gaussian deblur ($\sigma = 6$)
TReg	Classifier free guidance scale (w)	2.87	4.92	3.31
	λ for conjugate gradient [160]	9.97e-04	2.05e-05	1.12e-04
	Adaptive negation learning rate (η) [160]	1.56e-04	4.17e-04	3.33e-04
PSLD	γ defined in [251]	0.298	0.0187	0.124
	η defined in [251]	0.896	0.102	0.132
DCDP	SGD learning rate	9.95e-02	1.35e-02	3.29e-02
	SGD momentum	0.964	0.872	0.642
	Likelihood optimization steps	49	28	46
PnP-DM	HMC learning rate	4.83e-04	1.99e-05	1.07e-04
	HMC momentum	0.496	0.440	0.404
	Likelihood sampling steps	14	15	26
DAPS	HMC learning rate	1.00e-04	2.00e-06	1.00e-05
	HMC momentum	0.45	0.45	0.45
	Likelihood sampling steps	30	30	30

D.2 Appendix for Section 7.4

D.2.1 Detailed Implementation of STeP

Here, we summarize the proposed framework for solving video inverse problems in Algorithm 7.

Algorithm 7 STeP: a Framework for Solving Video Inverse Problems with **S**patio**Te**mporal **P**rior

Require: discretization time steps $\{t_i\}_{i=1}^N$ where $t_0 = 0$ and $t_N = T$, noise schedule σ_t, likelihood $p(\boldsymbol{y} \mid \cdot)$ with measurements $\boldsymbol{y}$, HMC step size η and damping factor γ, number of HMC updates M, pre-trained latent score function $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z}; \sigma) \approx \nabla_{\boldsymbol{z}} \log p(\boldsymbol{z}; \sigma)$ with image decoder $\mathcal{D}$.

1: $\boldsymbol{z}_{t_N} \sim \mathcal{N}(\mathbf{0}, \sigma_{t_N}^2 \boldsymbol{I})$ ▷ Initialization
2: **for** $i = N, ..., 1$ **do**
3: $\widehat{\boldsymbol{z}}_0 \leftarrow \texttt{Backward}(\boldsymbol{z}_{t_i}; \boldsymbol{s}_{\boldsymbol{\theta}})$ ▷ Solve PF-ODE (D.5) backward from $t = t_i$ to $t = 0$ to enforce prior
4: $\boldsymbol{p} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
5: **for** $j = 1, ..., M$ **do**
6: $\boldsymbol{\epsilon}_j \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
7: $(\widehat{\boldsymbol{z}}_0, \boldsymbol{p}) = \texttt{HamiltonianDynamics}(\widehat{\boldsymbol{z}}_0, \boldsymbol{y}, \boldsymbol{p}, \boldsymbol{\epsilon}_j; \eta, \gamma)$ ▷ HMC updates to enforce data consistency
8: **end for**
9: $\boldsymbol{z}_{t_{i-1}} \sim \mathcal{N}(\widehat{\boldsymbol{z}}_0, \sigma_{t_{i-1}}^2 \boldsymbol{I})$ ▷ Proceed to the next noise level at time $t = t_{i-1}$
10: **end for**
11: **return** $\mathcal{D}(\boldsymbol{z}_{t_0})$ ▷ Return the decoded video

The algorithm's main loop alternates between three key steps: (1) solving the PF-ODE backward from $t = t_i$ to $t = 0$ (line 3), (2) performing multi-step MCMC updates (lines 4–8), and (3) advancing to the next noise level (line 9). We will discuss each step in detail.

Through the lens of Algorithm 2, line 3 can be viewed as the prior step, and lines 4-9 together can be viewed as the likelihood step. The equivalence can be seen by pattern matching with $K \equiv N$, $k \equiv N - i$ and $\eta_k \equiv \sigma_{t_{N-i}}$.

Solving PF-ODE Backward from $t = t_i$ to $t = 0$ The probability flow ordinary differential equation (PF-ODE) [151] of the diffusion model governs the continuous increase or reduction of noise in the image when moving forward or backward in time, given by

$$\mathrm{d}\boldsymbol{z}_t = -\dot{\sigma}_t \sigma_t \nabla_{\boldsymbol{z}_t} \log p(\boldsymbol{z}_t; \sigma_t)\, \mathrm{d}t, \tag{D.5}$$

where $\dot{\sigma}_t$ denotes the time derivative of σ_t, and $\nabla_{\boldsymbol{z}_t} \log p(\boldsymbol{z}_t; \sigma_t)$ represents the time-dependent score function [264, 266]. Our goal is to solve the probability flow ODE (PF-ODE), as defined in Equation (D.5), backward from $t = t_i$ to $t = 0$, given the intermediate state $\boldsymbol{z}_{t_i}$ and the pre-trained latent score function $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z}; \sigma) \approx \nabla_{\boldsymbol{z}} \log p(\boldsymbol{z}; \sigma)$. Any ODE solver, such as Euler's method or the fourth-order Runge-Kutta method (RK4) [39], can be used to solve this problem. Following previous conventions [341], we adopt a few-step Euler method for solving it efficiently.

Multi-Step MCMC Updates Any MCMC samplers can be used, such as Langevin Dynamic Monte Carlo (LMC) and Hamiltonian Monte Carlo (HMC). For example, the LMC update with step size η is

$$\boldsymbol{z}_0^+ = \boldsymbol{z}_0 + \eta \nabla_{\boldsymbol{z}_0} \log p(\boldsymbol{y} \mid \mathcal{D}(\boldsymbol{z}_0)) + \eta \nabla_{\boldsymbol{z}_0} \log p(\boldsymbol{z}_0 \mid \boldsymbol{z}_t) + \sqrt{2\eta}\boldsymbol{\epsilon}. \tag{D.6}$$

Note that the first gradient term can be computed with a differentiable forward model $\mathcal{A}$ and the neural network decoder $\mathcal{D}$. The second gradient term, on the other hand, can be calculated by

$$\nabla_{\boldsymbol{z}_0} \log p(\boldsymbol{z}_0 \mid \boldsymbol{z}_t) = \nabla_{\boldsymbol{z}_0} \log p(\boldsymbol{z}_t \mid \boldsymbol{z}_0) + \nabla_{\boldsymbol{z}_0} \log p(\boldsymbol{z}_0) \approx \nabla_{\boldsymbol{z}_0} \log p(\boldsymbol{z}_t \mid \boldsymbol{z}_0) + \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z}_0, t_{\min}).$$

This approximation holds for $t_{\min} \approx 0$, assuming that $\boldsymbol{z}_0$ lies close to the clean latent manifold [264]. To improve both convergence speed and approximation accuracy, the MCMC samplers are initialized with the solutions obtained from the previous PF-ODE step, leveraging its outputs as a warm start.

Note that during MCMC updates, the decoder $\mathcal{D}$ needs to be evaluated multiple times in the backward pass. To accelerate this process, we adopt Hamiltonian Monte Carlo (HMC), which typically requires fewer steps for convergence, thereby speeding up the algorithm. For each multi-step MCMC update, we introduce an additional

Table D.2: **Summary of the hyperparameters of Algorithm 7 (STeP) for black hole video reconstruction and dynamic MRI.** The HMC-related parameters are tuned on 3 leave-out validation videos. The run time and memory are tested using one NVIDIA A100 GPU.

	Black hole video reconstruction	Dynamic MRI
PF-ODE related		
number of steps N_{ode}	20	20
scheduler σ_t	t	t
HMC related		
number of steps M	60	53
scaling factor $1-\gamma\eta$	0.00	0.83
step size square η^2	1.2e-5	1.2e-3
observation noise level $\sigma_{\boldsymbol{y}}$	0.02	0.01
Annealing Schedule related		
number of steps N	25	20
final time T	100	100
discretization time $\{t_i\}, i = 1, \cdots, N$	$\left(\frac{N-i}{N} \cdot T^{\frac{1}{7}}\right)^7$	$\left(\frac{N-i}{N} \cdot T^{\frac{1}{7}}\right)^7$
Inference related		
decoder NFE N_{dec}	1500	1060
diffusion model NFE N_{dm}	500	400
time (s) per sample	645	332
memory (GB)	48	21

momentum variable $\boldsymbol{p}$, initialized as $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. The `HamiltonianDynamics`$(\widehat{z}_0, \boldsymbol{y}, \boldsymbol{p}, \boldsymbol{\epsilon}; \eta, \gamma)$ update with step size η and damping factor γ is given by:

$$\boldsymbol{p}^+ = (1-\gamma\eta)\cdot\boldsymbol{p} + \eta\nabla_{z_0}\log p(\boldsymbol{y} \mid \mathcal{D}(z_0)) + \eta\nabla_{z_0}\log p(z_0 \mid z_t) + \sqrt{2\gamma\eta}\boldsymbol{\epsilon},$$
$$z_0^+ = z_0 + \eta\boldsymbol{p}^+.$$

Proceeding to Next Noise Level According to Proposition 1 in [341], one can obtain a sample $z_{t_{i-1}} \sim p(z_{t_{i-1}} \mid \boldsymbol{y})$ by simply adding Gaussian noise from a sample $\hat{z}_0 \sim p(z_0 \mid z_{t_i}, \boldsymbol{y})$, given $z_{t_i} \sim p(z_{t_i} \mid \boldsymbol{y})$ from last step. Thus, we solve the target posterior sampling by gradually sampling from the time-marginal posterior of the diffusion trajectory. The full hyperparameters and running cost STeP is summarized in Table D.2. The HMC-related parameters are searched on a leave-out validation dataset consisting of three videos that are different from the testing videos.

D.2.2 Experimental Details

D.2.2.1 Black Hole Video Reconstruction

We introduce the black hole video reconstruction problem in more detail. The goal is to reconstruct a video $\boldsymbol{x}_0 \in \mathbb{R}^{n_f \times n_h \times n_w}$ of a rapidly moving black hole. Each

measurement, or *visibility*, is given by correlating the measurements from a pair of telescopes to sample a particular spatial Fourier frequency of the source with very long baseline interferometry (VLBI) [66, 339]. In VLBI, the cross-correlation of the recorded scalar electric fields at two telescopes, known as the ideal *visibility*, is related to the ideal source video $\boldsymbol{x}_0$ through a 2D Fourier transform, as given by the van Cittert-Zernike theorem [66, 339]. Specifically, the ideal visibility of the j-th frame of the target video is

$$\boldsymbol{k}_{\{a,b\}}^{[j]}(\boldsymbol{x}_0) := \int_{\rho}\int_{\delta} \exp\left[-i2\pi\left(u_{\{a,b\}}^{[j]}\rho + v_{\{a,b\}}^{[j]}\delta\right)\right] \boldsymbol{x}_0^{[j]}(\rho,\delta)\mathrm{d}\rho\mathrm{d}\delta \in \mathbb{C}, \tag{D.7}$$

where (ρ, δ) denotes the angular coordinates of the source video frame, and $\left(u_{\{a,b\}}^{[j]}, v_{\{a,b\}}^{[j]}\right)$ is the dimensionless baseline vector between two telescopes $\{a, b\}$, orthogonal to the source direction.

Due to atmospheric turbulence and instrumental calibration errors, the observed visibility is corrupted by gain error, phase error, and additive Gaussian thermal noise [91, 273]:

$$\boldsymbol{V}_{\{a,b\}}^{[j]} := g_a^{[j]} g_b^{[j]} \exp\left[-i\left(\phi_a^{[j]} - \phi_b^{[j]}\right)\right] \boldsymbol{k}_{\{a,b\}}^{[j]}(\boldsymbol{x}_0) + \boldsymbol{n}_{\{a,b\}}^{[j]} \in \mathbb{C}, \tag{D.8}$$

where gain errors are denoted by $g_a^{[j]}, g_b^{[j]}$, phase errors are denoted by $\phi_a^{[j]}, \phi_b^{[j]}$, and thermal noise is denoted by $\boldsymbol{n}_{\{a,b\}}^{[j]}$. While the phase of the observed visibility cannot be directly used due to phase errors, the product of three visibilities among any combination of three telescopes, known as the *bispectrum*, can be computed to retain useful information. Specifically, the phase of the bispectrum, termed the *closure phase*, effectively cancels out the phase errors in the observed visibilities. Similarly, a strategy can be employed to cancel out amplitude gain errors and extract information from the visibility amplitude [23]. Formally, these quantities are defined as

$$\boldsymbol{y}_{\mathrm{cp},\{a,b,c\}}^{[j]} := \angle(\boldsymbol{V}_{\{a,b\}}^{[j]} \boldsymbol{V}_{\{b,c\}}^{[j]} \boldsymbol{V}_{\{a,c\}}^{[j]}) := \mathcal{A}_{\mathrm{cp},\{a,b,c\}}^{[j]}(\boldsymbol{x}_0) \in \mathbb{R}, \tag{D.9}$$

$$\boldsymbol{y}_{\mathrm{logca},\{a,b,c,d\}}^{[j]} := \log\left(\frac{\left|\boldsymbol{V}_{\{a,b\}}^{[j]}\right|\left|\boldsymbol{V}_{\{c,d\}}^{[j]}\right|}{\left|\boldsymbol{V}_{\{a,c\}}^{[j]}\right|\left|\boldsymbol{V}_{\{b,d\}}^{[j]}\right|}\right) := \mathcal{A}_{\mathrm{logca},\{a,b,c,d\}}^{[j]}(\boldsymbol{x}_0) \in \mathbb{R}. \tag{D.10}$$

Here, $\angle(\cdot)$ denotes the complex angle, and $|\cdot|$ computes the complex amplitude. For a total of M telescopes, the number of closure phase measurements $\boldsymbol{y}_{\mathrm{cp},\{a,b,c\}}^{[j]}$ at is $\frac{(M-1)(M-2)}{2}$, and the number of log closure amplitude measurements $\boldsymbol{y}_{\mathrm{logca},\{a,b,c,d\}}^{[j]}$ is

$\frac{M(M-3)}{2}$, after accounting for redundancy. Let $d_{\text{cp}}^{[j]}$ and $d_{\text{logca}}^{[j]}$ to indicate the dimension of $\boldsymbol{y}_{\text{cp}}^{[j]}$ and $\boldsymbol{y}_{\text{logca}}^{[j]}$. Since closure quantities are nonlinear transformations of the visibilities, the black hole video reconstruction problem is non-convex. Additionally, because the closure quantities do not constrain the total flux (i.e., summation of the pixel values) of the underlying black hole video, we add a constraint on the total flux for each frame, defined as

$$\boldsymbol{y}_{\text{flux}}^{[j]} := \int_{\rho}\int_{\delta} \boldsymbol{x}_0^{[j]}(\rho,\delta)\mathrm{d}\rho\mathrm{d}\delta. \tag{D.11}$$

To aggregate data over different measurement times and telescope combinations, the forward model of black hole video reconstruction for the j-th frame can be expressed as

$$\boldsymbol{y}^{[j]} := \left[\mathcal{A}_{\text{cp}}^{[j]}(\boldsymbol{x}_0), \mathcal{A}_{\text{logca}}^{[j]}(\boldsymbol{x}_0), \mathcal{A}_{\text{flux}}^{[j]}(\boldsymbol{x}_0)\right] := \left[\boldsymbol{y}_{\text{cp}}^{[j]}, \boldsymbol{y}_{\text{logca}}^{[j]}, \boldsymbol{y}_{\text{flux}}^{[j]}\right], \tag{D.12}$$

where $\boldsymbol{y}_{\text{cp}}^{[j]} = \left[\boldsymbol{y}_{\text{cp},\{a,b,c\}}^{[j]}, \forall\{a,b,c\}\right]$ is the set of all closure phase measurements and $\boldsymbol{y}_{\text{cp}}^{[j]} = \left[\boldsymbol{y}_{\text{logca},\{a,b,c,d\}}^{[j]}, \forall\{a,b,c,d\}\right]$ is the set of all log closure amplitude measurements over all combinations of telescopes for the j-th frame. The overall data consistency is an aggregation over all frames and typically expressed using the χ^2 statistics

$$\begin{aligned}
-\log p(\boldsymbol{y} \mid \boldsymbol{x}_0) \propto & \underbrace{\sum_{j=1}^{n_f} \frac{1}{n_f d_{\text{cp}}^{[j]} \sigma_{\text{cp}}^2} \left\| \mathcal{A}_{\text{cp}}^{[j]}(\boldsymbol{x}_0) - \boldsymbol{y}_{\text{cp}}^{[j]} \right\|^2}_{\chi^2_{\text{cp}}} \\
& + \underbrace{\sum_{j=1}^{n_f} \frac{1}{n_f d_{\text{logca}}^{[j]} \sigma_{\text{logca}}^2} \left\| \mathcal{A}_{\text{logca}}^{[j]}(\boldsymbol{x}_0) - \boldsymbol{y}_{\text{logca}}^{[j]} \right\|^2}_{\chi^2_{\text{logca}}} \\
& + \beta \underbrace{\sum_{j=1}^{n_f} \frac{1}{n_f \sigma_{\text{flux}}^2} \left\| \mathcal{A}_{\text{flux}}^{[j]}(\boldsymbol{x}_0) - \boldsymbol{y}_{\text{flux}}^{[j]} \right\|^2}_{\chi^2_{\text{flux}}}
\end{aligned} \tag{D.13}$$

where σ_{cp}, σ_{logca}, and σ_{flux} are the estimated standard deviations of the measured closure phase, log closure amplitude, and flux, respectively, and β is a hyperparameter that controls the strength of the flux regularization, which is empirically

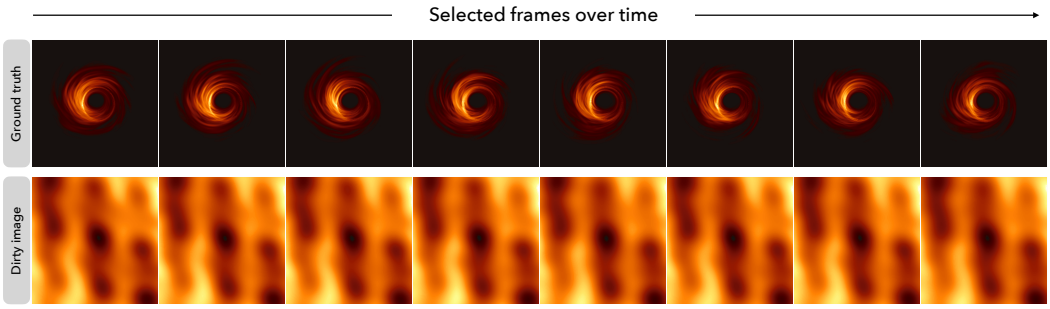


Figure D.1: **The dirty images from the ideal visibilities.** We use the standard implementation in `EHT` library to get dirty images for each selected frame.

determined. To evaluate the data fitting, we introduce a unified $\tilde{\chi}^2$ statistics

$$\tilde{\chi}^2 = \chi^2 \cdot \mathbb{1}\{\chi^2 \geq 1\} + \frac{1}{\chi^2} \cdot \mathbb{1}\{\chi^2 < 1\}. \tag{D.14}$$

The $\tilde{\chi}^2$ is no less than 1 and closer to 1, indicating better measurement data fit. In our experiments we use an average $(\tilde{\chi}^2_{\text{cp}} + \tilde{\chi}^2_{\text{logca}})/2$ for the data misfit metrics for evaluation.

Our experiments are based on the simulation of observing the Sagittarius A* black hole with the EHT 2017 array of eight radio telescopes over an observation period of around 100 minutes. We refer the readers to Figure 5 of [178] for a visualization of the measurement patterns in Fourier space over time. To show the difficulty of this black hole video reconstruction problem, we visualize the dirty video frames obtained by applying the inverse Fourier transform to the ideal visibilities, assuming no measurement errors, in Figure D.1. One can see that substantial spatiotemporal information is lost during the measurement process, so obtaining high-quality reconstructions relies on the effectiveness of incorporating prior information in the reconstruction process.

D.2.2.2 Dynamic MRI

MRI is an important imaging technique for clinical diagnosis and biomedical research, where the objective is to recover a video $\boldsymbol{x}_0 \in \mathbb{C}^{n_f \times n_h \times n_w}$ of the heart from the subsampled Fourier space (a.k.a k-space) measurements $\boldsymbol{y}$. Despite its many advantages, MRI is known to be slow because of the physical limitations of the data acquisition in k-space. This leads to low patient throughput and sensitivity to patients' motion [298]. To accelerate the scan speed, instead of fully sampling k-space, the compressed subsampling MRI (CS-MRI) technique subsamples k-space

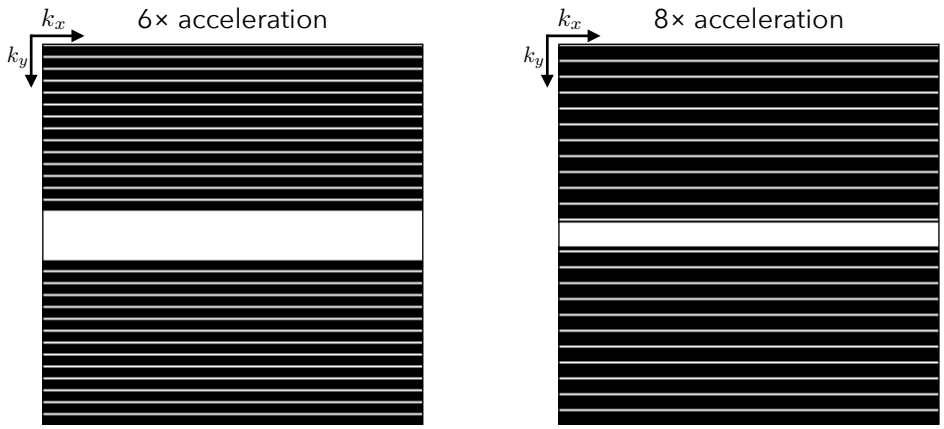


Figure D.2: **Subsampling masks of 6× (left) and 8× (right) accelerations used in dynamic MRI experiments.** The white areas in the center indicate the auto-calibration (ACS) signals. The horizontal and vertical directions are the frequency (k_x) and phase (k_x) encoding directions, respectively. The same mask is applied to the sampling of each individual frame of all videos.

with masks $\{\boldsymbol{m}^{[j]}\}_{j=1}^{n_f}$. Mathematically, this can be formulated as

$$\boldsymbol{y}^{[j]} = \boldsymbol{m}^{[j]} \odot \left(\boldsymbol{F}\boldsymbol{x}_0^{[j]}\right) + \boldsymbol{n}^{[j]} \in \mathbb{C}^n \quad \text{for } j = 1, ..., n_f, \tag{D.15}$$

where $\boldsymbol{m}^{[j]} \in \{0,1\}^{n_h \times n_w}$ is the subsampling mask for the j-th frame, $\odot$ denotes element-wise multiplication, and $\boldsymbol{n}^{[j]}$ is the measurement noise. In our experiments, we used subsampling masks with an equi-spaced pattern (similar to those visualized in [298]) of both 6× acceleration with 24 auto-calibration signal (ACS) lines (Table D.4) and 8× acceleration with 12 ACS lines (Table 7.2). For dynamic MRI, we use the Gaussian likelihood function

$$-\log p(\boldsymbol{y} \mid \boldsymbol{x}_0) \propto \frac{1}{n_f} \sum_{j=1}^{n_f} \left\| \mathcal{A}^{[j]}(\boldsymbol{x}_0) - \boldsymbol{y}^{[j]} \right\|_2^2, \tag{D.16}$$

where $\mathcal{A}^{[j]}(\boldsymbol{x}_0) := \boldsymbol{m}^{[j]} \odot \left(\boldsymbol{F}\boldsymbol{x}_0^{[j]}\right)$. Figure D.2 visualizes the subsampling masks used in our experiments, where k_x, k_y indicate the frequency encoding and phase encoding directions, respectively. The same mask is applied to the sampling of each frame of all videos.

D.2.2.3 Baseline Implementations

We provide details on implementations of our baseline methods, following the same grouping in Section 7.4.3.1.

Group 1: Simple Heuristics We implement *batch independent sampling* (BIS) and *batch consistent sampling* (BCS) following [168, 169]. BIS is implemented to reconstruct each frame independently, whereas BCS promotes temporal consistency by using identical ("batch-consistent") noise across the temporal dimension. To ensure a fair comparison, we adapt Algorithm 7 by initializing the noise variables z_{t_N} (line 1) and $z_{t_{i-1}}$ (line 9) with batch-consistent noise.

Group 2: Noise Warping We follow [53, 82] in performing noise warping based on optical flow estimated from measurements. As described in Section 7.4.3.1, we derive optical flow using ground-truth black hole videos and inverse Fourier-transformed dynamic MRI videos with a pre-trained model [278]. Since we utilize a latent diffusion model, we further downsample the optical flow via interpolation to match the latent noise dimension. We adapt Algorithm 7 accordingly by replacing the original *i.i.d.* noise with the warped noise while keeping other components unchanged. We implement *Bilinear*, *Bicubic*, and *Nearest* warping strategies using their corresponding interpolation methods, and follow the implementation from `https://github.com/yitongdeng-projects/infinite_resolution_integral_noise_warping_code` (MIT License) for $\int$-noise. Lastly, since [77] does not provide publicly available code, we implement GP-Warp following Equation (2) from their paper.

D.2.3 Training Details for Spatiotemporal Diffusion Prior

In this section, we show the details of getting a video diffusion prior on black hole video reconstruction and dynamic MRI, and we summarize the training hyperparameters in Table D.3. We define D_{image} and D_{video} as the image and video datasets, containing N_{image} and N_{video} data points, respectively. The image dataset D_{image} includes all individual frames from the video dataset D_{video}, along with additional large-scale image data to enhance generalization. For data augmentation, we apply random horizontal/vertical flipping and random zoom-in-and-out to improve robustness and diversity in training.

We first train the compression functions, the encoder $\mathcal{E}$ and decoder $\mathcal{D}$, on an image dataset. The training objective consists of an ℓ_1 reconstruction loss combined with a KL divergence term scaled by a factor β_{KL}. The loss function for training is as defined in Equation (D.17). The Adam optimizer is used as the default optimizer throughout the paper. The loss function for training the variational autoencoder

(VAE) is given by

$$\mathcal{L}_{\text{VAE}} = \mathbb{E}_{q_\phi(z_0|\boldsymbol{x}_0),\boldsymbol{x}_0\sim D_{\text{image}}}\left[\|\mathcal{D}(z_0) - \boldsymbol{x}_0\|_1\right] + \beta_{\text{KL}}\mathsf{KL}\big(q_\phi(z_0|\boldsymbol{x}_0)\|p(z_0)\big) \quad \text{(D.17)}$$

where $p(z_0)$ is the standard Gaussian $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$ and $q_\phi(z_0|\boldsymbol{x}_0)$ is the isotropic Gaussian distribution over z_0 where the mean and standard deviation is given by $\mathcal{E}(\boldsymbol{x}_0)$. Next, we train the image diffusion U-Net $\boldsymbol{s}_{\boldsymbol{\theta}}$ using the standard score-matching loss

$$\mathcal{L}_{\text{IDM}} = \mathbb{E}_{z_0\sim q_\phi(z_0|\boldsymbol{x}_0),x_0\sim D_{\text{image}},\epsilon\sim\mathcal{N}(\mathbf{0},\boldsymbol{I}),t\sim\mathcal{U}(0,1)}\left[\sigma_t^2\left\|\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) - \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t|\boldsymbol{x}_0)\right\|^2\right], \quad \text{(D.18)}$$

following [131, 266]. After pre-training, the image diffusion U-Net $\boldsymbol{s}_{\boldsymbol{\theta}}$ is then converted to a spatiotemporal U-Net by adding zero-initialized temporal modules to 2D spatial modules and fine-tuned jointly with video and image datasets. We use the same Equation (D.18) without changing $\boldsymbol{x}_0$ to video or pseudo-video input and use the encoder to process each frame independently.

Table D.3: **Summary of the training of spatiotemporal diffusion prior.** We provide and group the hyperparameters according to each component in the model. The model is trained with 1 NVIDIA A100-SCM4-80GB GPU.

Hyper-parameters	Black hole video reconstruction	Dynamic MRI
Dataset Related		
frames n_f	64	12
resolution $n_h \times n_w$	256×256	192×192
N_{image}	50000	39888
N_{video}	648	3324
VAE Training Related		
latent channels	1	2
block channels	[64, 128, 256, 256]	[256, 512, 512]
down sampling factor	8	4
batch size	16	16
epochs	25	10
β_{KL}	0.06	0.03
IDM Training Related		
block channels	[128, 256, 512, 512]	[128, 256, 512, 512]
batch size	16	16
epochs	200	50
Joint Fine-tuning Related		
p_{joint}	0.8	0.8
epochs	500	300
Other Info		
VAE parameters	14.8M	57.5M
diffusion model parameters	131.7M	131.7M
VAE training time	4.5h	8.9h
image diffusion model training time	5.5h	3.8h
joint fine-tuning time	13.7h	22.8h

Table D.4: **Additional results on dynamic MRI with 6× acceleration.** Following the same setup as in Table 7.2, we report the quantitative results on 10 test videos.

Tasks	Methods	PSNR (↑)	SSIM (↑)	LPIPS (↓)	d-PSNR (↑)	d-SSIM (↑)	FVD (↓)	Data Misfit (↓)
MRI (6×)	BIS [168]	39.47 (0.59)	0.958 (0.007)	0.086 (0.011)	43.26 (1.23)	0.962 (0.005)	113.17	11.071 (0.740)
	BCS [168]	40.69 (0.57)	0.959 (0.006)	0.081 (0.012)	44.73 (1.31)	0.974 (0.004)	110.01	11.085 (0.720)
	Bilinear [53]	40.85 (0.57)	0.960 (0.006)	0.080 (0.012)	44.84 (1.28)	0.975 (0.004)	114.37	11.038 (0.735)
	Bicubic [53]	40.71 (0.67)	0.959 (0.007)	0.079 (0.012)	44.74 (1.38)	0.974 (0.005)	106.82	11.068 (0.755)
	Nearest [53]	40.37 (0.56)	0.960 (0.007)	0.080 (0.012)	44.81 (1.41)	0.974 (0.005)	110.91	11.050 (0.739)
	$\int$-noise [53, 82]	40.09 (0.50)	0.960 (0.006)	0.082 (0.012)	44.77 (1.34)	0.974 (0.005)	111.92	11.059 (0.731)
	GP-Warp [77]	39.50 (0.48)	0.959 (0.007)	0.080 (0.012)	44.53 (1.33)	0.973 (0.005)	105.70	11.070 (0.727)
	STEP (video only)	40.76 (0.43)	0.967 (0.005)	0.077 (0.012)	46.38 (1.82)	0.981 (0.005)	101.83	**10.788** (0.713)
	STEP (image-video joint)	**41.39** (0.52)	**0.969** (0.005)	**0.076** (0.012)	**46.61** (1.72)	**0.982** (0.004)	**98.15**	10.808 (0.723)

After pre-training, the image diffusion U-Net s_θ is transformed into a spatiotemporal U-Net by integrating zero-initialized temporal modules into the existing 2D spatial modules. The model is then fine-tuned jointly using both video and image datasets. We use the same loss as in Equation (D.18), by changing $\boldsymbol{x}_0$ to a video or a pseudo-video input. Each frame is independently processed using the encoder $\mathcal{E}$, ensuring that spatial representations remain aligned while temporal consistency is learned through the added temporal modules.

D.2.4 Limitations

Though STEP is a general framework for solving scientific VIPs with spatiotemporal diffusion prior, the sampling cost of STEP is relatively high due to the requirement of backpropagation through decoder $\mathcal{D}$ in MCMC updates in Algorithm 7. So we have to strike a balance between the capability of the decoder and its computational cost.

D.2.5 More Results and Visualization

Data Misfit Values for Samples in Figure 7.6 We report the data misfit values in Table D.5.

Table D.5: **The data misfit values for samples shown in Figure 7.6.** We report the data misfit metrics for the three obtained modes, which demonstrate that all modes fit the measurement data equally well.

Metrics	Mode 1	Mode 2	Mode 3
χ^2_{cp}	1.045	0.987	1.084
χ^2_{logca}	1.007	1.001	1.202
Data Misfit	1.026	1.007	1.143

Dynamic MRI with Higher Acceleration To access the capability of using a spatiotemporal prior for solving more challenging inverse problems, we increase

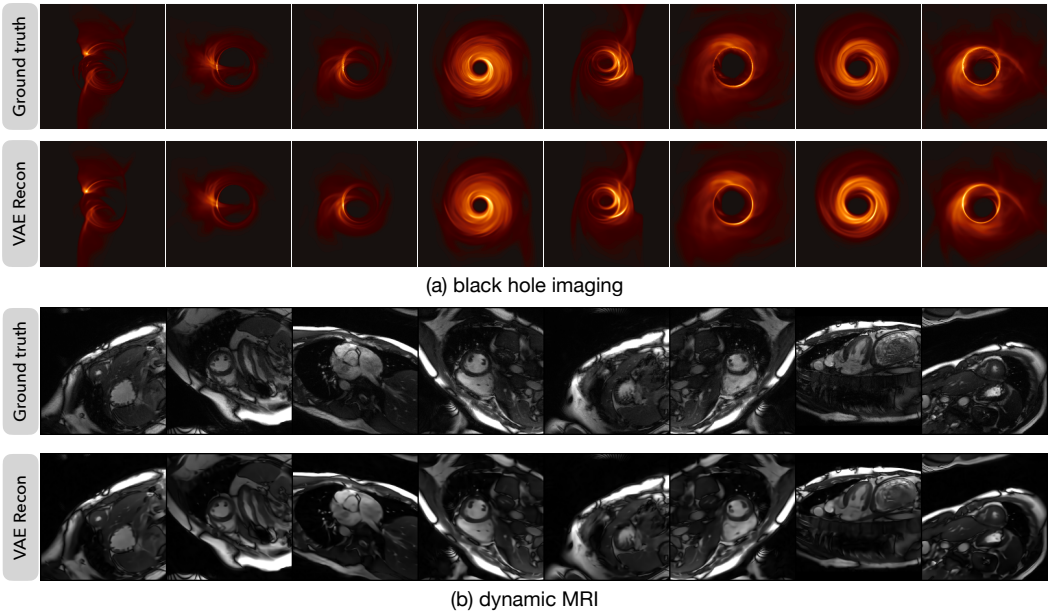


Figure D.3: **Visualization of VAE reconstructions**. The VAE reconstructions are computed by first encoding the ground truth videos and then decoding them.

the acceleration times in Dynamic MRI, which makes the observation more sparse. The results are summarized in Table D.4.

More Visualizations Here, we show the VAE reconstruction results in Figure D.3, unconditional samples in Figure D.4 and additional posterior samples in Figure D.5.

D.3 Appendix for Section 7.5

D.3.1 Theory

D.3.1.1 Notation

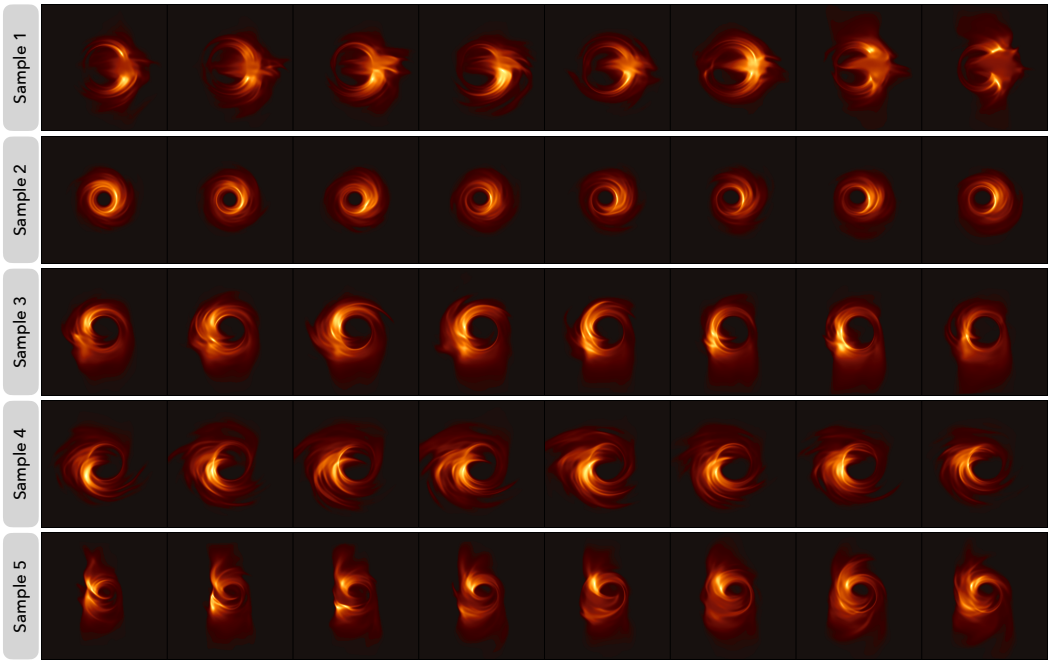
Recall that for the prior step, we consider the following SDE that corresponds to Equation (6.6) with $s(t) = 1$

$$\mathrm{d}\boldsymbol{x}_t = \left[-(2\dot{\sigma}(t)\sigma(t) + \beta(t))\nabla \log p_t\left(\frac{\boldsymbol{x}_t}{s(t)}; \sigma(t)\right)\right]\mathrm{d}t + \left(\sqrt{2\dot{\sigma}(t)\sigma(t)} + \sqrt{2\beta(t)}\right)\mathrm{d}\bar{\boldsymbol{w}}_t. \tag{D.19}$$

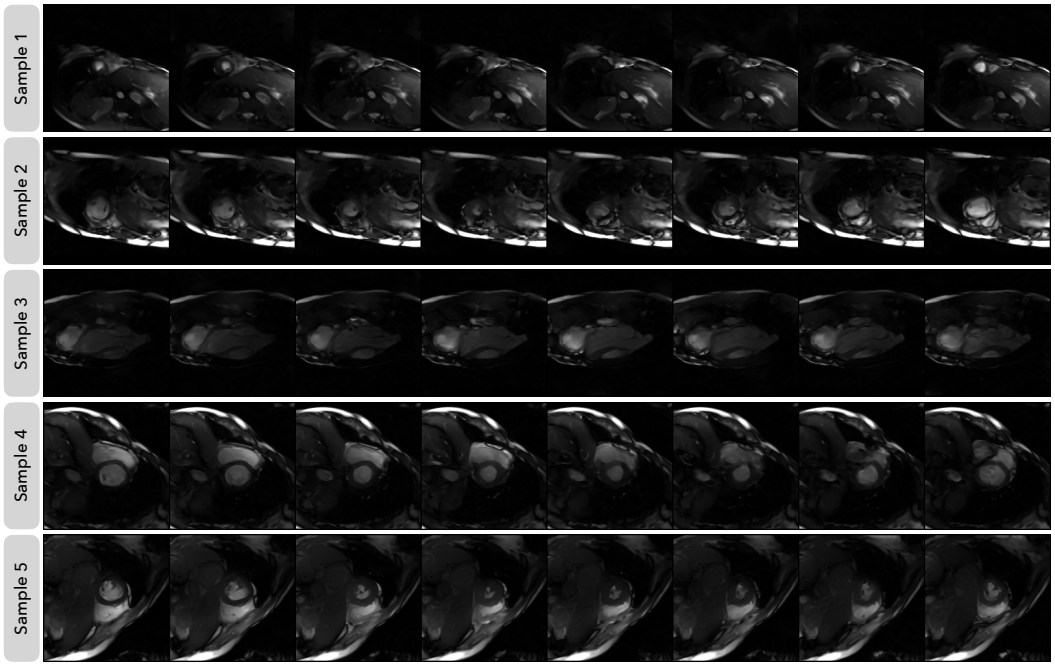
We denote the drift coefficient and the diffusion coefficient as $h(t)$ and $\delta(t)$, respectively:

$$h(t) := -(2\dot{\sigma}(t)\sigma(t) + \beta(t)) \tag{D.20}$$

$$\delta(t) := \sqrt{2\dot{\sigma}(t)\sigma(t)} + \sqrt{2\beta(t)}. \tag{D.21}$$

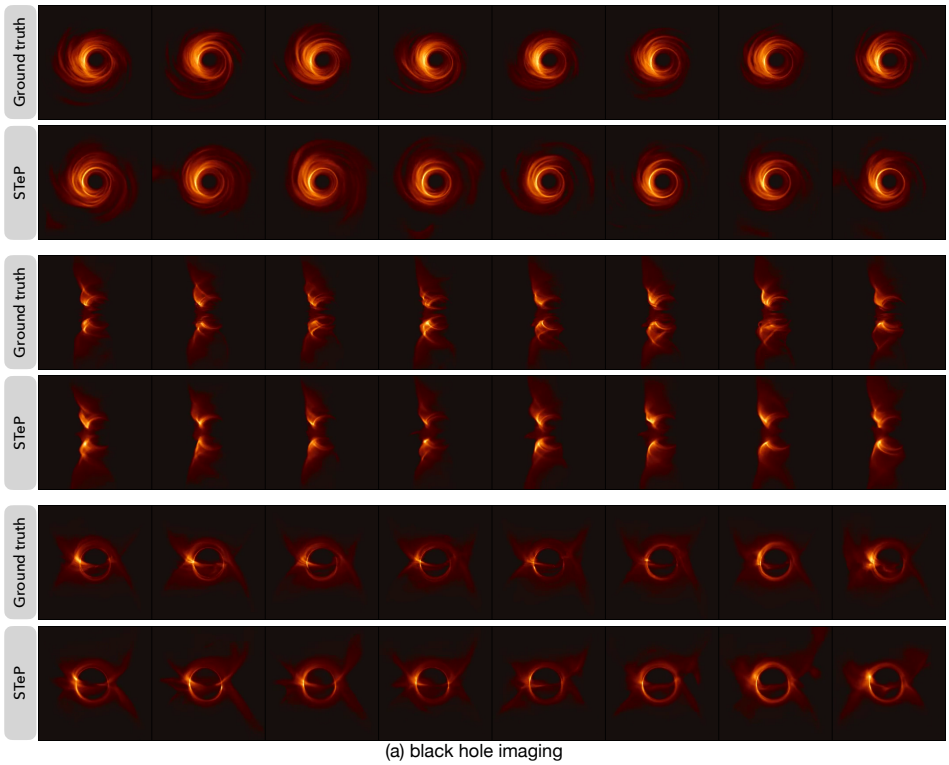


(a) black hole imaging

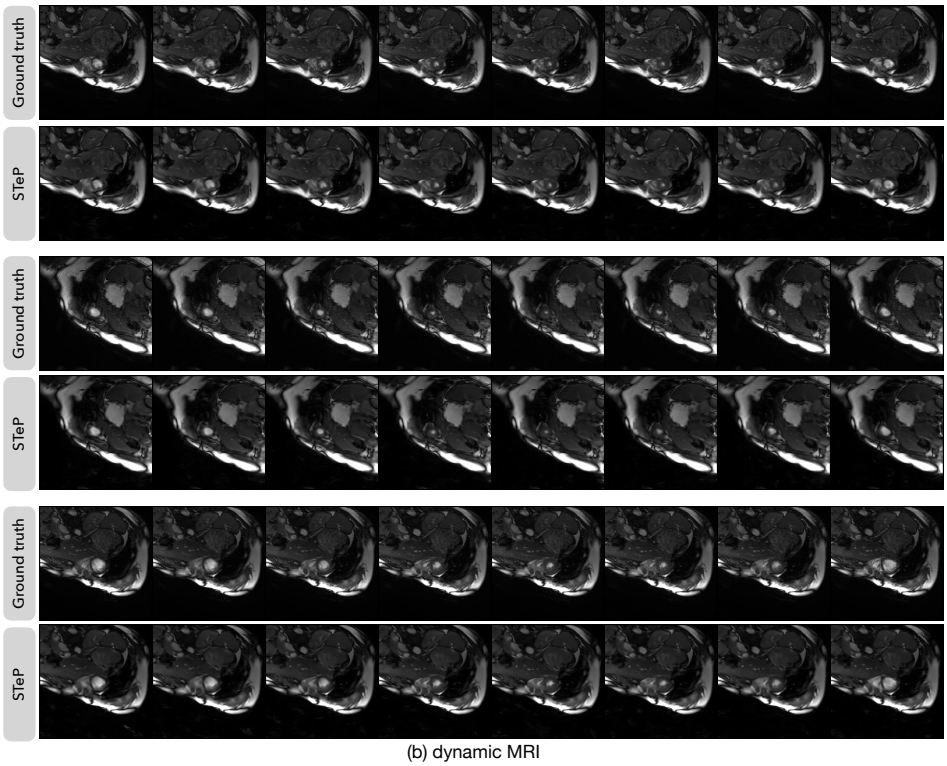


(b) dynamic MRI

Figure D.4: **Visualization of video diffusion model unconditional samples**. The videos are sampled by solving the PF-ODE with 100 Euler steps.



(a) black hole imaging



(b) dynamic MRI

Figure D.5: **Visualization of STeP posterior samples**. The videos are sampled using the Algorithm 7.

Recall from Section 6.4 that the Kullback–Leibler (KL) divergence between two distributions μ and π is defined by

$$\mathsf{KL}(\mu||\pi) = \int \mu \log \frac{\mu}{\pi} = \mathbb{E}_\mu \left[\log \frac{\mu}{\pi}\right].$$

The Fisher divergence between two distributions μ and π is defined by

$$\mathsf{FI}(\mu||\pi) = \int \mu \left\|\nabla \log \frac{\mu}{\pi}\right\|_2^2 = \mathbb{E}_\mu \left\|\nabla \log \frac{\mu}{\pi}\right\|_2^2.$$

For $\boldsymbol{x} \in \mathbb{R}^n$, we define the weighted norm induced by a positive semi-definite matrix $\boldsymbol{M} \in \mathbb{R}^{n\times n}$ as

$$\|\boldsymbol{x}\|_{\boldsymbol{M}}^2 = \boldsymbol{x}^T \boldsymbol{M} \boldsymbol{x}, \tag{D.22}$$

and the divergence of a matrix $T(\boldsymbol{x}) \in \mathbb{R}^{n\times n}$ as the vector field

$$(\nabla_{\boldsymbol{x}} \cdot T)_i = \sum_{j=1}^{n} \frac{\partial T_{ij}}{\boldsymbol{x}_j}. \tag{D.23}$$

D.3.1.2 Assumptions and Lemmas

Assumption D.3.1. *The average score approximation error of the diffusion model $\boldsymbol{s}_t := \boldsymbol{s}_{\boldsymbol{\theta}}(\,\cdot\,;\, t)$ is bounded,*

$$\epsilon_{\text{score}} = \sup_{k=0,\ldots,K-1} \left\{ \frac{1}{t^*} \int_{T_k+t^\dagger}^{T_{k+1}} \frac{h(t)^2}{\delta(t)^2} \mathbb{E}_{\mu_t} \|\boldsymbol{s}_t - \nabla_{\boldsymbol{x}_t} \log p_{\sigma(t)}\|_2^2 \mathrm{d}t \right\} < +\infty, \tag{D.24}$$

where $h(t)$ is defined in Equation (D.20) *and $\delta(t)$ is defined in Equation* (D.21).

Assumption D.3.2. *The average derivative approximation error of the linear surrogate model $\boldsymbol{A}_t$ is bounded,*

$$\epsilon_{\text{model}} = \sup_{k=0,\ldots,K-1} \left\{ \frac{1}{t^\ddagger} \int_{T_k}^{T_k+t^\dagger} \mathbb{E}_{\mu_t} \left\| \nabla f(\boldsymbol{z}_t; \boldsymbol{y}) - \frac{1}{\sigma_{\boldsymbol{y}}^2} \boldsymbol{A}_t^T \left(\mathcal{A}(\boldsymbol{z}_t^{(j)}) - \boldsymbol{y} \right) \right\|_{C_t}^2 \mathrm{d}t \right\} < +\infty, \tag{D.25}$$

where $\|\cdot\|_{C_t}$ is the weighted norm defined in Equation (D.22).

Assumption D.3.3. *The Radon–Nikodym derivative $\frac{\mathrm{d}\mu_t}{\mathrm{d}\widetilde{\mu}_t}$ is constant along the null space of C_t almost surely, where C_t is the covariance matrix of $\widetilde{\mu}_t$.*

$$\frac{\mathrm{d}\mu_t}{\mathrm{d}\widetilde{\mu}_t}(\boldsymbol{x}) = \frac{\mathrm{d}\mu_t}{\mathrm{d}\widetilde{\mu}_t}(\boldsymbol{x} + \boldsymbol{v}), \forall \boldsymbol{v} \in \mathrm{Ker}(C_t).$$

Remark D.3.4. Assumption D.3.1 coincides with the usual bounded score-matching error condition that underpins convergence results [57, 175, 320]. Assumption D.3.2 characterizes the ℓ_2 accuracy of the linear proxy. Assumption D.3.3 is the weakest assumption to bound the weighted Fisher divergence in our analysis. Two common sufficient (but not necessary) scenarios are: (1) C_t is full-rank and (2) μ_t is absolutely continuous with respect to $\widetilde{\mu}_t$, and both log densities are continuously differentiable.

Lemma D.3.5 (Stationary distribution of the likelihood step)**.** *Assume the particle distribution is not a Dirac measure, the dynamics of Equation* (7.11) *admits* $\pi^{Z|X=\boldsymbol{x}^{(j)}}(z) \propto \exp(-f(z;\boldsymbol{y}) - \frac{1}{2\eta^2}\|z-\boldsymbol{x}^{(j)}\|_2^2)$ *as a stationary distribution. Further, if the covariance matrix is positive definite, the stationary distribution is unique.*

Proof. This result has been proved in various forms in the literature [111, 206]. Here, we provide a simple proof of our use case for ease of understanding. Suppose $\mu_t(z)$ is the probability density of z at time t. For the ease of notation, we ignore the particle index j in z_t. Let $\Phi(z) := f(z;\boldsymbol{y}) + \frac{1}{2\eta^2}\|z-\boldsymbol{x}^{(j)}\|_2^2$. The corresponding Fokker-Planck equation for Equation (7.11) reads

$$\partial_t \mu_t = \nabla \cdot (\mu_t C_t \nabla \Phi(z)) + \nabla \cdot (C_t \nabla \mu_t)\,,$$

which can be rewritten as

$$\partial_t \mu_t = \nabla \cdot (\mu_t C_t (\nabla \Phi(z) + \nabla \log \mu_t))\,. \tag{D.26}$$

Let μ_∞ denote the stationary distribution of Equation (D.26). We have

$$0 = \nabla \cdot (\mu_t C_t (\nabla \Phi(z) + \nabla \log \mu_\infty))\,.$$

If the particle distribution is not Dirac, $C_t \neq 0$ due to Lemma 2.1 in [111]. Therefore,

$$\nabla \Phi(z) + \nabla \log \mu_\infty = c,$$

where c is a constant. Integrating both sides gives

$$\mu_\infty(z) \propto \exp(-\Phi(z)) = \exp\left(-f(z;\boldsymbol{y}) - \frac{1}{2\eta^2}\|z-\boldsymbol{x}^{(j)}\|_2^2\right),$$

showing that $\pi^{Z|X=\boldsymbol{x}^{(j)}}(z)$ is a stationary distribution of the dynamics of Equation (7.11). Further, if C_t is positive definite, it ensures the irreducibility and strong Feller property, and the stationary distribution is unique [206, 247]. □

Lemma D.3.6. *Given the following pair of stochastic processes*

$$\mathrm{d}\boldsymbol{x}_t = b(\boldsymbol{x}_t, t)\mathrm{d}t + H(t)\mathrm{d}\boldsymbol{w}_t, \tag{D.27}$$

$$\mathrm{d}\widetilde{\boldsymbol{x}}_t = \widetilde{b}(\widetilde{\boldsymbol{x}}_t, t)\mathrm{d}t + H(t)\mathrm{d}\boldsymbol{w}_t, \tag{D.28}$$

where $b, \widetilde{b} : \mathbb{R}^n \times \mathbb{R}^+ \to \mathbb{R}^n$ are the drift terms, $H : \mathbb{R}^+ \to \mathbb{R}^n \times \mathbb{R}^n$ is the diffusion term, $\boldsymbol{w}_t$ is the standard Wiener process. Let μ_t (respectively $\widetilde{\mu}_t$) be the law of $\boldsymbol{x}_t$ (respectively $\widetilde{\boldsymbol{x}}_t$), $C(t) := H(t)H(t)^T$, and λ_t^* be the smallest non-zero eigenvalue of $C(t)$. Assuming that $b_t - \widetilde{b}_t \in \mathrm{Range}(C(t))$ and Assumption D.3.3 holds, we have

$$\partial_t \mathsf{KL}(\mu_t || \widetilde{\mu}_t) \leq -\frac{\lambda_t^*}{4}\mathsf{FI}(\mu_t || \widetilde{\mu}_t) + \mathbb{E}_{\mu_t}\left\| b_t - \widetilde{b}_t \right\|^2_{C(t)^\dagger}, \tag{D.29}$$

where $C(t)^\dagger$ is the pseudo-inverse of $C(t)$.

Proof. Since the diffusion terms only depend on t and $C(t) = H(t)H(t)^T$, the Fokker-Planck equations of Equation (D.27) and Equation (D.28) read

$$\partial_t \mu_t = \nabla \cdot \left[\left(\frac{1}{2} C(t) \nabla \log \mu_t - b_t \right) \mu_t \right], \tag{D.30}$$

$$\partial_t \widetilde{\mu}_t = \nabla \cdot \left[\left(\frac{1}{2} C(t) \nabla \log \widetilde{\mu}_t - \widetilde{b}_t \right) \widetilde{\mu}_t \right]. \tag{D.31}$$

Let $r_t := \frac{\mu_t}{\widetilde{\mu}_t}$ and $\phi(r_t) := r_t \log r_t$ (so $\phi'(r_t) = \frac{\mathrm{d}}{\mathrm{d}r_t}\phi(r_t) = \log r_t + 1$). Differentiating the KL divergence gives

$$\begin{aligned}
\partial_t \mathsf{KL}(\mu_t || \widetilde{\mu}_t) &= \partial_t \int \phi(r_t)\widetilde{\mu}_t \\
&= \int \left[\phi(r_t)\partial_t \widetilde{\mu}_t + \phi'(r_t)\partial_t \widetilde{\mu}_t \right] \\
&= \int \left[\phi(r_t)\partial_t \widetilde{\mu}_t + \phi'(r_t)\partial_t \mu_t - \phi'(r_t) r_t \partial_t \widetilde{\mu}_t \right] \\
&= \int \left[(\log r_t + 1)\partial_t \mu_t - r_t \partial_t \widetilde{\mu}_t \right],
\end{aligned} \tag{D.32}$$

where the last step uses the fact that $\phi(r_t) - r_t\phi'(r_t) = -r_t$. Plugging Equation (D.30) and Equation (D.31) into Equation (D.32) and applying integration by parts further,

we have

$$
\begin{aligned}
&\partial_t \mathsf{KL}(\mu_t || \widetilde{\mu}_t) \\
&= \int (\log r_t + 1) \nabla \cdot \left[\left(\frac{1}{2} C(t) \nabla \log \mu_t - b_t \right) \mu_t \right] - \int r_t \nabla \cdot \left[\left(\frac{1}{2} C(t) \nabla \log \widetilde{\mu}_t - \widetilde{b}_t \right) \widetilde{\mu}_t \right] \\
&= - \int \left\langle \nabla \log r_t, \frac{1}{2} C(t) \nabla \log \mu_t - b_t \right\rangle \mu_t + \int \left\langle \nabla r_t, \frac{1}{2} C(t) \nabla \log \widetilde{\mu}_t - \widetilde{b}_t \right\rangle \widetilde{\mu}_t \\
&= - \int \left\langle \nabla \log r_t, \frac{1}{2} C(t) \nabla \log \mu_t - b_t \right\rangle \mu_t + \int \left\langle \nabla \log r_t, \frac{1}{2} C(t) \nabla \log \widetilde{\mu}_t - \widetilde{b}_t \right\rangle \mu_t \\
&= - \int \left\langle \nabla \log r_t, \frac{1}{2} C(t) \left(\nabla \log \mu_t - \nabla \log \widetilde{\mu}_t \right) \right\rangle \mu_t + \int \left\langle \nabla \log r_t, b_t - \widetilde{b}_t \right\rangle \mu_t \\
&= -\frac{1}{2} \int \left\langle \nabla \log r_t, C(t) \nabla \log r_t \right\rangle \mu_t + \int \left\langle \nabla \log r_t, b_t - \widetilde{b}_t \right\rangle \mu_t . \qquad \text{(D.33)}
\end{aligned}
$$

The weighted Young's inequality states that, for any $u, v \in \mathbb{R}^n$, when $v \in \mathrm{Range}(C)$, we have

$$
\langle u, v \rangle \leq \frac{1}{4} \langle u, Cu \rangle + \left\langle v, C^\dagger v \right\rangle ,
$$

where $C^\dagger$ is the pseudo-inverse. By Assumption D.3.3, Equation (D.33) can be bounded as follows

$$
\begin{aligned}
&- \frac{1}{2} \int \left\langle \nabla \log r_t, C(t) \nabla \log r_t \right\rangle \mu_t + \int \left\langle \nabla \log r_t, b_t - \widetilde{b}_t \right\rangle \mu_t \\
&\leq -\frac{1}{4} \int \left\langle \nabla \log r_t, C(t) \nabla \log r_t \right\rangle \mu_t + \int \left\langle b_t - \widetilde{b}_t, C(t)^\dagger (b_t - \widetilde{b}_t) \right\rangle \mu_t \\
&\leq -\frac{\lambda_t^*}{4} \mathsf{FI}(\mu_t || \widetilde{\mu}_t) + \mathbb{E}_{\mu_t} \left\| b_t - \widetilde{b}_t \right\|^2_{C(t)^\dagger} \qquad \text{(D.34)}
\end{aligned}
$$

where λ_t^* is the smallest non-zero eigenvalue of $C(t)$. □

This is a generalization of Lemma C.1.4 to the general matrix-valued diffusion term. Intuitively, the condition that $b_t - \widetilde{b}_t$ belongs to the range of $C(t)$ means that the two drift terms may only differ along the directions that are actually driven by noise. In the context of our proof below, this is always satisfied because the drift terms are either preconditioned with $C(t)$ or $C(t)$ is full-rank.

D.3.1.3 Proof of Theorem 7.5.1

Proof. For $\tau \in [T_k, T_k+t^\dagger]$, $k = 0, ..., K-1$, we apply Lemma D.3.6 to the likelihood step with

$$
\begin{aligned}
b(z_t, t) &:= -C_t \nabla f(z_t; \boldsymbol{y}) - \frac{1}{\eta^2} C_t (z_t - \boldsymbol{x}^{(j)}) \\
\widetilde{b}(z_t, t) &:= -C_t \frac{1}{\sigma_{\boldsymbol{y}}^2} \boldsymbol{A}_t^T (\mathcal{A}(z_t) - \boldsymbol{y}) - \frac{1}{\eta^2} C_t (z_t - \boldsymbol{x}^{(j)}) \\
H(t) &= \sqrt{C_t},
\end{aligned}
$$

where $\boldsymbol{A}_t = \mathbb{E}_{\widetilde{\mu}_t}[(\mathcal{A}(z_t) - \mathbb{E}_{q_t}\mathcal{A}(z_t))z_t^T]C_t^{-1}$ as defined in Equation (7.13). Note that the condition $b - \widetilde{b} \in \text{Range}(C_t)$ is satisfied as both drift terms are preconditioned with C_t. Thus, by Assumption D.3.3, we have

$$
\begin{aligned}
\partial_\tau \mathsf{KL}(\mu_\tau || \widetilde{\mu}_\tau) &\leq -\frac{\lambda_\tau^*}{4} \mathsf{FI}(\mu_\tau || \widetilde{\mu}_\tau) + \mathbb{E}_{\mu_\tau} \left\langle b_\tau - \widetilde{b}_\tau, C_\tau^\dagger (b_\tau - \widetilde{b}_\tau) \right\rangle \\
&\leq -\frac{\lambda^*}{4} \mathsf{FI}(\mu_\tau || \widetilde{\mu}_\tau) + \mathbb{E}_{\mu_\tau} \left\| \nabla f(z_\tau; \boldsymbol{y}) - \frac{1}{\sigma_{\boldsymbol{y}}^2} \boldsymbol{A}_\tau^T (\mathcal{A}(z_\tau) - \boldsymbol{y}) \right\|_{C_\tau}^2
\end{aligned}
$$

where λ_τ^* is the smallest non-zero eigenvalue of C_τ and $\lambda^* := \inf_{\tau \in [T_k, T_k+t^\dagger], k=0,...,K-1} \lambda_\tau^*$. By Assumption D.3.2 , integrating both sides over $[T_k, T_k + t^\dagger]$ gives

$$
\begin{aligned}
&\mathsf{KL}(\mu_{T_k+t^\dagger} || \widetilde{\mu}_{T_k+t^\dagger}) - \mathsf{KL}(\mu_{T_k} || \widetilde{\mu}_{T_k}) \\
&\leq -\frac{\lambda^*}{4} \int_{T_k}^{T_k+t^\dagger} \mathsf{FI}(\mu_\tau || \widetilde{\mu}_\tau) \mathrm{d}\tau + \int_{T_k}^{T_k+t^\dagger} \mathbb{E}_{\mu_\tau} \left\| \nabla f(z_\tau; \boldsymbol{y}) - \frac{1}{\sigma_{\boldsymbol{y}}^2} \boldsymbol{A}_\tau^T (\mathcal{A}(z_\tau) - \boldsymbol{y}) \right\|_{C_\tau}^2 \mathrm{d}\tau \\
&\leq -\frac{\lambda^*}{4} \int_{T_k}^{T_k+t^\dagger} \mathsf{FI}(\mu_\tau || \widetilde{\mu}_\tau) \mathrm{d}\tau + t^\dagger \epsilon_{\text{model}}, \qquad \text{(D.35)}
\end{aligned}
$$

where ϵ_{model} is defined in Equation (D.25). For $\tau \in [T_k + t^\dagger, T_{k+1}]$, $k = 0, ..., K-1$, we apply Lemma D.3.6 to the prior step (D.19) with

$$
\begin{aligned}
b(\boldsymbol{x}_t, t) &:= h(t) \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t; \sigma(t)) \\
\widetilde{b}(z_t, t) &:= h(t) \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t; t) \\
H(t) &:= \delta(t) \boldsymbol{I},
\end{aligned}
$$

where $h(t)$ is the drift coefficient defined in Equation (D.20), $\delta(t)$ is the diffusion coefficient defined in Equation (D.21), $\boldsymbol{s}_{\boldsymbol{\theta}}$ is the pre-trained diffusion model with score approximation error ϵ_{score}. Note that $H(t)H(t)^T$ is full-rank so that the

condition of Lemma D.3.6 is satisfied. Therefore, we have

$$\begin{aligned}\partial_\tau \mathsf{KL}(\mu_\tau||\widetilde{\mu}_\tau) &\leq -\frac{\delta(\tau)^2}{4}\mathsf{FI}(\mu_\tau||\widetilde{\mu}_\tau) + \frac{h(\tau)^2}{\delta(\tau)^2}\mathbb{E}_{\mu_\tau}\|\nabla_{\boldsymbol{x}_\tau}\log p_{\sigma(\tau)} - \boldsymbol{s}_\tau\|_2^2 \\ &\leq -\frac{\delta}{4}\mathsf{FI}(\mu_\tau||\widetilde{\mu}_\tau) + \frac{h(\tau)^2}{\delta(\tau)^2}\mathbb{E}_{\mu_\tau}\|\nabla_{\boldsymbol{x}_\tau}\log p_{\sigma(\tau)} - \boldsymbol{s}_\tau\|_2^2,\end{aligned}$$

where $\delta := \inf_{\tau\in[0,t^*]}\delta(\tau)^2$. Integrating both sides over $[T_k + t^\dagger, T_{k+1}]$ and applying Assumption D.3.1 gives

$$\begin{aligned}&\mathsf{KL}(\mu_{T_{k+1}}||\widetilde{\mu}_{T_{k+1}}) - \mathsf{KL}(\mu_{T_k+t^\dagger}||\widetilde{\mu}_{T_k+t^\dagger}) \\ &\leq -\frac{\delta}{4}\int_{T_k+t^\dagger}^{T_{k+1}}\mathsf{FI}(\mu_\tau||\widetilde{\mu}_\tau)\mathrm{d}\tau + \int_{T_k+t^\dagger}^{T_{k+1}}\frac{h(\tau)^2}{\delta(\tau)^2}\mathbb{E}_{\mu_\tau}\|\log p_{\sigma(\tau)} - \boldsymbol{s}_\tau\|_2^2\mathrm{d}\tau \\ &\leq -\frac{\delta}{4}\int_{T_k+t^\dagger}^{T_{k+1}}\mathsf{FI}(\mu_\tau||\widetilde{\mu}_\tau)\mathrm{d}\tau + t^*\epsilon_{\text{score}}, \qquad \text{(D.36)}\end{aligned}$$

where ϵ_{score} is defined in Equation (D.24). Summing up both sides of Equation (D.35) and Equation (D.36) for $k = 0, ..., K-1$ gives

$$\mathsf{KL}(\mu_{T_K}||\widetilde{\mu}_{T_K}) - \mathsf{KL}(\mu_0||\widetilde{\mu}_0) \leq -\frac{\min(\lambda^*,\delta)}{4}\int_0^{T_K}\mathsf{FI}(\mu_\tau||\widetilde{\mu}_\tau)\mathrm{d}\tau + K(t^\dagger\epsilon_{\text{model}} + t^*\epsilon_{\text{score}}).$$

Rearranging the terms gives

$$\begin{aligned}&\frac{1}{T_K}\int_0^{T_K}\mathsf{FI}(\mu_\tau||\widetilde{\mu}_\tau)\mathrm{d}\tau \\ &\leq \frac{4}{T_K\min(\lambda^*,\delta)}\left(\mathsf{KL}(\mu_0||\widetilde{\mu}_0) - \mathsf{KL}(\mu_{T_K}||\widetilde{\mu}_{T_K})\right) + \frac{4}{\min(\lambda^*,\delta)(t^\dagger+t^*)}(\epsilon_{\text{model}} + \epsilon_{\text{score}}) \\ &\leq \frac{4}{\min(\lambda^*,\delta)}\left[\frac{\mathsf{KL}(\mu_0||\widetilde{\mu}_0)}{K(t^\dagger+t^*)} + \frac{t^\dagger\epsilon_{\text{model}} + t^*\epsilon_{\text{score}}}{t^\dagger+t^*}\right] \\ &= \frac{c\mathsf{KL}(\mu_0||\widetilde{\mu}_0)}{K} + ct^\dagger\epsilon_{\text{model}} + ct^*\epsilon_{\text{score}}\end{aligned}$$

where $c := \frac{4}{\min(\lambda^*,\delta)(t^\dagger+t^*)}$. The proof is concluded by the fact that $\mu_0 = \pi^X$. □

Lemma D.3.7. *Let $\pi(z;\boldsymbol{x}^{(j)})$ denote the invariant measure associated with the potential $\Phi(z;\boldsymbol{x}^{(j)})$ where $\nabla_z\Phi(z;\boldsymbol{x}^{(j)}) = \left[\frac{1}{\sigma_{\boldsymbol{y}}^2}\boldsymbol{A}_t^T(\mathcal{A}(z) - \boldsymbol{y}) + \frac{1}{\eta^2}(z - \boldsymbol{x}^{(j)})\right]$. Then $\pi(z;\boldsymbol{x}^{(j)})$ is an invariant measure of the finite-particle system in Equation* (7.15) *as well as its large particle limit in Equation* (7.14).

Proof. In the large particle limit, the covariance C_t does not depend on any specific particle but depends on the particle distribution only. Therefore, the Fokker-Planck

equation of Equation (7.14) reads:

$$\begin{aligned}\partial_t p_t &= \nabla \cdot \left(p_t C_t \nabla \Phi(z_t^{(j)}; \boldsymbol{x}^{(j)}) \right) + C_t \nabla^2 p_t \\ &= \nabla \cdot \left(p_t C_t \left(\nabla \Phi(z_t^{(j)}; \boldsymbol{x}^{(j)}) + \nabla \log p_t \right) \right)\end{aligned}$$

where p_t is the probability density at time t. We can see that $\pi(z; \boldsymbol{x}^{(j)})$ is an invariant measure by setting both sides to zero. In the finite-particle system, the covariance $C_t = \frac{1}{J} \sum_{j=1}^{J} (z_t^{(j)} - \bar{z}_t)(z_t^{(j)} - \bar{z}_t)^T$, which depends on the current state $z_t^{(j)}$. Therefore, the Fokker-Plank equation of the finite-particle dynamics in Equation (7.15) is

$$\begin{aligned}\partial_t p_t =& \nabla \cdot \left[p_t \left(C_t \nabla \Phi(z_t^{(j)}; \boldsymbol{x}^{(j)}) - \frac{n+1}{J} (z_t^{(j)} - \bar{z}_t) \right) \right] + \nabla \cdot (\nabla \cdot (p_t C_t)) \\ =& \nabla \cdot \left[p_t \left(C_t \nabla \Phi(z_t^{(j)}; \boldsymbol{x}^{(j)}) - \frac{n+1}{J} (z_t^{(j)} - \bar{z}_t) \right) \right] + \nabla \cdot (C_t \nabla p_t + p_t \nabla \cdot C_t) \\ =& \nabla \cdot \left[p_t C_t \left(\nabla \Phi(z_t^{(j)}; \boldsymbol{x}^{(j)}) + \nabla \log p_t \right) \right] - \nabla \cdot \left(\frac{n+1}{J} (z_t^{(j)} - \bar{z}_t) \right) + \nabla \cdot (p_t \nabla \cdot C_t) \\ =& \nabla \cdot \left[p_t C_t \left(\nabla \Phi(z_t^{(j)}; \boldsymbol{x}^{(j)}) + \nabla \log p_t \right) \right] - \nabla \cdot \left(\frac{n+1}{J} (z_t^{(j)} - \bar{z}_t) \right) \\ &+ \nabla \cdot \left(p_t \nabla_{z_t^{(j)}} \cdot \frac{1}{J} \sum_{i=1}^{J} (z_t^{(i)} - \bar{z}_t)(z_t^{(i)} - \bar{z}_t)^T \right) \\ =& \nabla \cdot \left[p_t C_t \left(\nabla \Phi(z_t^{(j)}; \boldsymbol{x}^{(j)}) + \nabla \log p_t \right) \right] - \nabla \cdot \left(p_t \frac{n+1}{J} (z_t^{(j)} - \bar{z}_t) \right) \\ &+ \nabla \cdot \left(p_t \frac{1}{J} (n+1)(z_t^{(j)} - \bar{z}_t) \right) \\ =& \nabla \cdot \left[p_t C_t \left(\nabla \Phi(z_t^{(j)}; \boldsymbol{x}^{(j)}) + \nabla \log p_t \right) \right]\end{aligned}$$

where the divergence of a matrix is defined in Equation (D.23), and we use the following properties:

$$\begin{aligned}\nabla_{z_t^{(j)}} \cdot \left(z_t^{(j)} z_t^{(j)^T} \right) &= (n+1) z_t^{(j)}, \\ \nabla_{z_t^{(j)}} \cdot \left(z_t^{(j)} z_t^{(i)^T} \right) &= z_t^{(i)}, \\ \nabla_{z_t^{(j)}} \cdot \left(z_t^{(i)} z_t^{(j)^T} \right) &= n z_t^{(i)}, \\ \nabla_{z_t^{(j)}} \cdot \left(\bar{z}_t \bar{z}_t^T \right) &= \frac{n+1}{J} \bar{z}_t,\end{aligned}$$

where $i \neq j$. By taking both sides to zero, we have that $\pi(z; \boldsymbol{x}^{(j)})$ is an invariant measure of Equation (7.15) as well. □

This proof is largely adapted from [112, 226], which apply to more general scenarios. We tailor the proof to our case for ease of understanding.

Algorithm 8 `LikelihoodStep` of Algorithm 3

Require: initial ensemble $\boldsymbol{X} = \{\boldsymbol{x}^{(j)}\}_{j=1}^{J}$, forward model $\mathcal{A}$, observation $\boldsymbol{y}$, effective observation noise $\widetilde{\sigma}_{\boldsymbol{y}}$, coupling strength η, number of discretization steps N, step size scale γ.

$\boldsymbol{Z}_0 = \{z_0^{(j)}\}_{j=1}^{J} \leftarrow \boldsymbol{X}$
for $i = 0, ..., N-1$ **do**
  $\boldsymbol{d}_1^{(j)} \leftarrow -\frac{1}{J}\sum_{j'=1}^{J} \frac{1}{\widetilde{\sigma}_{\boldsymbol{y}}^2}\langle \mathcal{A}(z_i^{(j')}) - \bar{\mathcal{A}}, \mathcal{A}(z_i^{(j)}) - \boldsymbol{y}\rangle (z_i^{(j')} - \bar{z}_i) \quad$ for $j = 1, ..., J$
  $C_i \leftarrow \mathrm{cov}\left(z_i^{(1)}, ..., z_i^{(J)}\right)$
  $\boldsymbol{d}_2^{(j)} \leftarrow -\frac{1}{\eta^2} C_i(\boldsymbol{x}^{(j)} - z_i^{(j)}) + \frac{n+1}{J}(z_i^{(j)} - \bar{z}_i) \quad$ for $j = 1, ..., J$
  $\sqrt{C_i} \leftarrow \frac{1}{\sqrt{J}}\left[z_i^{(1)} - \bar{z}_i, ..., z_i^{(J)} - \bar{z}_i\right] \in \mathbb{R}^{n \times J}$
  $\boldsymbol{\epsilon}_{ij} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_J) \quad$ for $j = 1, ..., J$
  $\eta \leftarrow \gamma / \|\boldsymbol{d}_1 + \boldsymbol{d}_2\|_2^2$
  $z_{i+1}^{(j)} \leftarrow z_i^{(j)} + \eta(\boldsymbol{d}_1^{(j)} + \boldsymbol{d}_2^{(j)}) + \sqrt{2\eta}\sqrt{C_i}\boldsymbol{\epsilon}_{ij} \quad$ for $j = 1, ..., J$
end for
return $\boldsymbol{Z}_N = \{z_N^{(j)}\}_{j=1}^{J}$

Algorithm 9 `PriorStep` of Algorithm 3

Require: initial ensemble $\boldsymbol{Z} = \{z^{(j)}\}_{j=1}^{J}$, diffusion model $\boldsymbol{s}_{\boldsymbol{\theta}}$, assumed noise level $\eta > 0$, number of discretization steps N, noise schedule $\sigma(t) = t$, discretization time steps $\{t_i\}_{i=0}^{N}$ (monotonically decreasing to $t_N = 0$), solver (`SDE` or `ODE`)

$i^* \leftarrow \min\{i \in [N] \mid \sigma(t_i) \le \eta\}$
$\boldsymbol{X}_{i^*} = \{\boldsymbol{x}_{i^*}^{(j)}\}_{j=1}^{J} \leftarrow \boldsymbol{Z}$
for $i = i^*, ..., N-1$ **do**
  $\lambda \leftarrow 2$ **if** solver is `SDE` **else** 1
  $\boldsymbol{d}_i^{(j)} \leftarrow -\lambda t_i \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i^{(j)}; \sigma(t_i)) \quad$ for $j = 1, ..., J$
  $\boldsymbol{x}_{i+1}^{(j)} \leftarrow \boldsymbol{x}_i^{(j)} + (t_{i+1} - t_i)\boldsymbol{d}_i^{(j)} \quad$ for $j = 1, ..., J$
  if $i \neq N-1$ **and** solver is `SDE` **then**
    $\boldsymbol{\epsilon}_{ij} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \quad$ for $j = 1, ..., J$
    $\boldsymbol{x}_{i+1}^{(j)} \leftarrow \boldsymbol{x}_{i+1}^{(j)} + \sqrt{2t_i(t_i - t_{i+1})}\boldsymbol{\epsilon}_{ij} \quad$ for $j = 1, ..., J$
  end if
end for
return $\boldsymbol{X}_N = \{\boldsymbol{x}_N^{(j)}\}_{j=1}^{J}$

D.3.2 Background

D.3.2.1 Traditional Methods for Derivative-Free Posterior Estimation

Traditional methods for derivative-free posterior estimation include Markov chain Monte Carlo (MCMC) samplers [75, 113, 115] and Sequential Monte Carlo (SMC) approaches [78]. These methods offer theoretical convergence guarantees but face significant scalability challenges in real-world applications, especially in high dimensions. On the other hand, approximate Bayesian methods [46, 111, 138] offer better efficiency, but often struggle to capture complex posteriors. Additionally, these methods require access to the prior density (up to a normalizing constant). However, in practice, prior knowledge is often implicit in data (e.g., simulation archives and historical measurements) and difficult to directly model the density in high dimensions.

D.3.2.2 Diffusion-Based Derivative-Free Posterior Estimation

Many recent derivative-free algorithms [141, 277, 356] leverage diffusion models (DMs) as plug-and-play priors for solving high-dimensional inverse problems with complex prior distributions. DMs can flexibly capture complex prior distributions, but require optimization or sampling for posterior inference, mainly due to modeling the score function rather than the density. Optimization-based approaches typically introduce approximations that can lead to mis-calibrated posterior samples even in simple linear-Gaussian settings (see Section 5.1 of [297]). Sampling-based approaches are often asymptotically correct sampling [44, 89, 283, 318], but are typically strictly restricted to linear problems and do not generalize to nonlinear settings.

D.3.2.3 Ensemble Kalman Methodology

The Ensemble Kalman methodology was first introduced by [96] in the context of filtering problems and has gained popularity in applications such as reservoir modeling [227] and weather forecasting [137] due to its derivative-free nature and effectiveness in practical settings. In the context of inverse problems, [143] revisits this idea to propose Ensemble Kalman Inversion (EKI), spawning a variety of extensions: momentum-augmented updates for neural network training [166], Tikhonov and other regularization schemes for improved stability and efficiency [48, 139, 144]. More recently, [161, 356] have explored this idea to create derivative-free

diffusion guidance from an optimization perspective to guide the generation of a diffusion model for different applications.

D.3.3 Implementation Details of BLADE

In this sub-section, we provide details on the practical implementation of `Blade`. The choices on key hyperparameters are summarized in Table D.6. All the experiments are conducted on single NVIDIA GH200 GPU.

D.3.3.1 Likelihood Step

Initialization As Theorem 7.5.1 indicates, the initialization of `Blade` is quite flexible, provided the initial distribution maintains a finite KL divergence from the target distribution. We used samples from the prior distribution as initializations for the Navier-Stokes equation experiments, indicated by "DM" in Table D.6.

Discretization We discretize the SDE in Equation (7.15) using the standard Euler method with an adaptive step size defined as

$$\text{Step size} = \frac{\gamma}{\|\text{drift}\|_2^2},$$

where γ is the hyperparameter that controls the scale of the step size, drift is the drift term of the SDE in Equation (7.15). This adaptive step size is effective across all our experiments. Further design of adaptive step sizes could potentially reduce discretization error with fewer steps.

Resample During the likelihood step, we employ a resampling strategy to ensure that the particles are at the correct noise level η. Resampling is a commonly used method that has been shown to help improve the performance of algorithms such as DAPS [341], DiffPIR [361], and ReSample [260]. Specifically, we define the following resampling strategy:

$$z^{(j)}_{\text{resample}} = z^{(j)} + \eta' \boldsymbol{\epsilon},$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$, $\eta' = \max(0, \eta - \frac{\text{tr}(C_t)}{n})$, and n is the dimension of the variable z. Intuitively, we approximate the current noise level in z and add a corresponding amount of noise to bring $z^{(j)}_{\text{resample}}$ to noise level η. A key distinction from the prior work is that our η' is estimated from the ensemble while the existing methods need to tune it as part of the hyperparameters. In our main experiments, we apply a resampling strategy since it introduces minimal additional computation cost and yields slightly better results.

Table D.6: **Hyperparameters of `Blade` for the Navier-Stokes experiments in Table 7.3.**

σ_y	γ	$\widetilde{\sigma}_y$	$\eta_{\max}$	$\eta_{\min}$	Anneal schedule	# anneal steps	# iterations	Initialization
0	20	0.03	4.8	0.08	linear	25	50	DM
1	30	0.17	4.8	0.08	linear	25	50	DM
2	30	0.30	4.8	0.08	linear	25	50	DM

Effective Observation Noise In practice, we observe that weighting the likelihood with a smaller σ_y yields better performance. We denote this adjusted value as the effective observation noise, $\widetilde{\sigma}_y$. Using an effective noise smaller than σ_y potentially compensates for the smoothing effect introduced by statistical linearization. In practice, we treat $\widetilde{\sigma}_y$ as a hyperparameter and tune it so that the spread-skill ratio is close to 1.

D.3.3.2 Prior Step

The prior step is implemented as a denoising diffusion process, with its pseudocode detailed in Algorithm 9. We set $\sigma(t) = t$ for simplicity and employ the Euler ODE sampler for faster sampling. We discretize the denoising diffusion process with the standard Euler method. Following [151], we use the following step size:

$$t_i = \left(t_{\max}^{1/7} + \frac{i}{N-1} \left(t_{\min}^{1/7} - t_{\max}^{1/7} \right) \right)^7 \quad \text{for } i = 0, ..., N-1.$$

D.3.3.3 Annealing Schedule

We use the linear annealing schedule for the coupling strength η. Given the number of iterations K, the maximum value $\eta_{\max}$, and minimum value $\eta_{\min}$, the linear decay schedule reduces η_k as

$$\eta_k = \eta_{\max} + \frac{k}{K-1}(\eta_{\min} - \eta_{\max}) \quad \text{for } k = 0, ..., K-1.$$

D.3.4 Implementation Details of Baselines

For baseline methods that do not require additional training on paired data, such as DPG [277], SCG [141], EnKG [356], EKI [143], EKS [111] (with diffusion prior initialization), we follow the implementation provided in InverseBench (Chapter 8). For methods that do require training on paired data, specifically the end-to-end U-Net and conditional diffusion model (CDM), we first generate a collection of measurement-target pairs by simulating measurements from the prior training

dataset available in InverseBench. To evaluate their in-distribution performance, we retrain the U-Net and CDM for each noise level, which takes around 7-10 hours on a single NVIDIA GH200 GPU.

The end-to-end U-Net architecture is adapted from the U-Net used in our diffusion model by removing the time conditioning branch. The measurements are upsampled to the same resolution before being fed into the U-Net. However, it is important to note that observations are not always spatially aligned with the unknown signal in a general setting. Consequently, end-to-end neural networks typically require additional design considerations for different types of observations.

The CDM is also adapted from the U-Net architecture of the prior diffusion model. This involved replacing the self-attention module with cross-attention and incorporating a CNN-based observation encoder, following the conditioning mechanism used in Rombach et al. [249].

D.3.5 Navier-Stokes Equation

D.3.5.1 Problem Setup

We follow the experimental setup in InverseBench (Chapter 8), which we include here for completeness. We consider the 2D Navier-Stokes equation for a viscous, incompressible fluid in vorticity form on a torus,

$$\begin{aligned} \partial_t \omega(\boldsymbol{x}, t) + \boldsymbol{v}(\boldsymbol{x}, t) \cdot \nabla_{\boldsymbol{x}} \omega(\boldsymbol{x}, t) &= \nu \nabla_{\boldsymbol{x}}^2 \omega(\boldsymbol{x}, t) + f(\boldsymbol{x}), && \boldsymbol{x} \in (0, 2\pi)^2, t \in (0, T] \\ \nabla_{\boldsymbol{x}} \cdot \boldsymbol{v}(\boldsymbol{x}, t) &= 0, && \boldsymbol{x} \in (0, 2\pi)^2, t \in [0, T] \\ \omega(\boldsymbol{x}, 0) &= \omega_0(\boldsymbol{x}), && \boldsymbol{x} \in (0, 2\pi)^2 \end{aligned} \tag{D.37}$$

where $\boldsymbol{v} \in C\left([0, T]; H^r_{\text{per}}((0, 2\pi)^2; \mathbb{R}^2)\right)$ for any $r > 0$ is the velocity field, $\omega = \nabla_{\boldsymbol{x}} \times \boldsymbol{v}$ is the vorticity, $\omega_0 \in L^2_{\text{per}}\left((0, 2\pi)^2; \mathbb{R}\right)$ is the initial vorticity, $\nu \in \mathbb{R}_+$ is the viscosity coefficient, and $f \in L^2_{\text{per}}\left((0, 2\pi)^2; \mathbb{R}\right)$ is the forcing function. The solution operator $\mathcal{G}$ is defined as the operator mapping the vorticity from the initial vorticity to the vorticity at time T, i.e., $\mathcal{G} : \omega_0 \to \omega_T$. Numerically, it is realized as a pseudo-spectral solver [127]. This Navier-Stokes equation is a standard benchmark problem widely used in the literature [143, 184, 276]. The forward model is given by

$$\boldsymbol{y} = \boldsymbol{P}\boldsymbol{G}(\boldsymbol{w}_0) + \boldsymbol{n}, \tag{D.38}$$

where $\boldsymbol{P}$ is the sampling operator and $\boldsymbol{G}(\cdot)$ is the discretization of $\mathcal{G}$.

D.3.5.2 Evaluation Metrics

We adopt the following three standard metrics to evaluate the results from different perspectives.

Relative ℓ_2 Error Suppose $\boldsymbol{x}_0$ is the ground truth function and $\widehat{\boldsymbol{x}}$ is the predicted function. The relative ℓ_2 error measures the norm of the error relative to the norm of the ground truth, defined as $\frac{\|\widehat{\boldsymbol{x}}-\boldsymbol{x}_0\|_2}{\|\boldsymbol{x}_0\|_2}$.

Continuous Ranked Probability Score (CRPS) The CRPS [118] is a standard probabilistic metric to assess the quality of the entire predicted distribution for inverse problems, which is defined as

$$\text{CRPS} = \mathbb{E}|\widehat{\boldsymbol{x}} - \boldsymbol{x}_0| - \frac{1}{2}\mathbb{E}|\widehat{\boldsymbol{x}} - \widehat{\boldsymbol{x}}'|,$$

where $\widehat{\boldsymbol{x}}, \widehat{\boldsymbol{x}}'$ are independent random predictions. Intuitively, it measures the distance between a predicted distribution and the single ground truth $\boldsymbol{x}_0$ that actually occurred. It is minimized when the ensemble prediction is drawn from the same distribution as the ground truth, i.e., $\boldsymbol{x}^{(j)} \sim p(\boldsymbol{x}_0)$ for all j. We consider the multi-dimensional version of CRPS defined in [241]. For an ensemble prediction $\{\boldsymbol{x}^{(j)}\}_{j=1}^{J}$ where $\boldsymbol{x}^{(j)} \in \mathbb{R}^n$, the CRPS for the single ground truth $\boldsymbol{x}_0$ is given by

$$\text{CRPS} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{J}\sum_{j=1}^{J}|\boldsymbol{x}^{(j)}(i) - \boldsymbol{x}_0(i)| - \frac{1}{2J(J-1)}\sum_{j=1}^{J}\sum_{k=1}^{J}|\boldsymbol{x}^{(j)}(i) - \boldsymbol{x}^{(k)}(i)|\right), \tag{D.39}$$

which can be implemented in $O(nJ\log J)$ complexity using the equivalent form introduced in [337]. In our experiments, we report the CRPS averaged over all test cases.

Spread-Skill Ratio (SSR) The spread-skill ratio (SSR) is a simple yet powerful metric for quantifying how well an ensemble prediction's stated uncertainty (spread) matches its actual error (skill) [106]. Intuitively, if the ensemble distribution truly captures the variability of the ground truth, then ensemble members should be statistically indistinguishable from observed outcomes. Formally, let $\{\boldsymbol{x}_i^*\}_{i=1}^{N}$ denote a set of observed ground truths. Suppose, for each observed ground truth $\boldsymbol{x}_i^*$, we have an ensemble prediction $\{\boldsymbol{x}_{i,j}\}_{j=1}^{J}$. Let $\bar{\boldsymbol{x}}_i := \frac{1}{J}\sum_j \boldsymbol{x}_{i,j}$. The unbiased estimator of SSR can be written as

$$\text{SSR} = \sqrt{\frac{\text{spread}^2}{\text{skill}^2}}, \tag{D.40}$$

where

$$\text{spread}^2 = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{J-1}\sum_{j=1}^{J}\|\boldsymbol{x}_{i,j}-\bar{\boldsymbol{x}}_i\|_2^2,$$

$$\text{skill}^2 = \frac{1}{N}\sum_{i=1}^{N}\|\frac{1}{J}\sum_{j}\boldsymbol{x}_{i,j}-\boldsymbol{x}_i^*\|_2^2 + \frac{1}{J(J-1)}\text{spread}^2.$$

A value of SSR = 1 indicates the perfect calibration. Small SSR means that the ensemble prediction is over-confident, while large SSR indicates that the ensemble prediction is over-cautious.

D.4 Appendix for Section 7.6

D.4.1 Theory

Notations We consider a linear stochastic process whose forward Kolmogorov equation can be written as $\partial_t\pi_t = \boldsymbol{Q}_t\pi_t$ with boundary condition $\pi_{t=0} = \pi_0$, where $\pi_t \in \mathcal{P}(\mathcal{X})$ is a probability measure on $\mathcal{X}$. For the simplicity of notation, we let $\boldsymbol{Q}_t$ to generate *reverse continuous-time Markov chain*, which means $\boldsymbol{Q}_t^{[i,j]} = \boldsymbol{s}(\boldsymbol{x}_j;\, t)_{\boldsymbol{x}_i}\boldsymbol{Q}_{\text{uniform}}^{[j,i]}$ where $\boldsymbol{s}(\boldsymbol{x}_j;\, t)_{\boldsymbol{x}_i}$ is the concrete score. Time t flows forward as the sampling algorithm progresses, which means that for unconditional generation π_0 is the data distribution and π_T is the uniform distribution.

For two probability mass functions μ and π, we define the KL divergence of μ with respect to π as

$$\mathsf{KL}(\mu\|\pi) := \mathbb{E}_{\boldsymbol{x}\sim\mu}\left[\log\frac{\mu}{\pi}(\boldsymbol{x})\right]. \tag{D.41}$$

We define the *Fisher divergence (or relative Fisher information)* of μ with respect to π as

$$\mathsf{FI}_{\boldsymbol{Q}}(\mu\|\pi) := \sum_{\boldsymbol{x}_i,\boldsymbol{x}_j\in\mathcal{X}}\pi(\boldsymbol{x}_i)\boldsymbol{Q}^{[j,i]}\left(f(\boldsymbol{x}_j) - f(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)\log\frac{f(\boldsymbol{x}_j)}{f(\boldsymbol{x}_i)}\right), \tag{D.42}$$

where $f := \mu/\pi$. Note that when $\boldsymbol{Q}$ is irreducible, the Fisher divergence $\mathsf{FI}_{\boldsymbol{Q}}(\mu\|\pi) \geq 0$, and $\mathsf{FI}_{\boldsymbol{Q}}(\mu\|\pi) = 0$ if and only if $\mu = \pi$. It is also important to see that this definition of KL divergence for discrete spaces is the same as its continuous version in Section 6.4 and Appendix D.3.1.1. However, the definition of Fisher divergence is significantly different because the gradient is not well-defined in discrete spaces.

In continuous state spaces, the Fisher divergence has been used to derive general first-order guarantees for non-log-concave sampling [15]. This line of analysis has been adapted to provide theoretical insights for posterior sampling methods

using diffusion models [273, 320]. The formula in Equation (D.42) represents a discrete state space analogue of Fisher divergence. We refer interested readers to [27, 87, 130] for more discussions on this topic and the relation between KL and Fisher divergence in the discrete state space. Our analysis in this paper adapts these techniques and provides general first-order guarantees for posterior sampling with discrete diffusion models.

Time Interpolation of SGDD SGDD alternates between likelihood steps and prior steps. Let $t^* > 0$ such that $\sigma(t^*) = \eta$. We define $\{\pi_\tau\}$ as the distributions at time τ of the stationary process, and $\{\mu_\tau\}$ as the distributions of the non-stationary process.

- In time intervals $\tau \in [k(t^*+1)+1, (k+1)(t^*+1)]$. The stationary distribution is initialized with $\pi_{k(t^*+1)+1}(\boldsymbol{x}) = \pi_\eta^Z(\boldsymbol{x})$. We run a prior step on μ_τ with the learned concrete score function for H steps, while π_τ evolves in continuous time with the true concrete score function.
- In time intervals $\tau \in [k(t^*+1), k(t^*+1)+1]$, we run a Metropolis-Hastings sampling algorithm on both π_τ and μ_τ.

Assumptions Our analysis relies on the following assumptions:

(i) Concrete score is well estimated: $\left\| \frac{s_\theta(\cdot\,;t) - s(\cdot\,;t)}{s(\cdot\,;t)} \right\|_\infty \leq \epsilon < 1$.

(ii) Smoothness of score function in t: $\| \frac{s(\cdot\,;t+\Delta t) - s(\cdot\,;t)}{s(\cdot\,;t)} \|_\infty \leq L \cdot \Delta t$.

(iii) Strong irreducibility: $\boldsymbol{Q}_t^{[i,j]} > 0$ for $i \neq j$.

(iv) Bounded probability ratio: $\sup_t \left\| \log \frac{\mu_t(\boldsymbol{x})}{\pi_t(\boldsymbol{x})} \right\|_\infty \leq B$.

(v) The entry-wise absolute of the reverse-time transition matrix is bounded: $\sup_t \| |\boldsymbol{Q}_t| \|_1 \leq M$.

D.4.1.1 Lemmas

Lemma D.4.1 (Data processing inequality of Metropolis-Hastings)**.** *Running Metropolis-Hastings algorithm on two distributions π_τ and μ_τ does not increase their KL divergence, i.e.,*

$$\mathsf{KL}(\mu_{k(t^*+1)+1} \| \pi_{k(t^*+1)+1}) \leq \mathsf{KL}(\mu_{k(t^*+1)} \| \pi_{k(t^*+1)}). \tag{D.43}$$

Lemma D.4.2 (Free-energy-rate-functional-relative-Fisher-information (FIR) inequality (from Theorem 6.2.3. in [129]))**.** *Consider two continuous time Markov chains: $\partial_t \pi_t = \boldsymbol{Q}_t \pi_t$ and $\partial_t \mu_t = \widetilde{\boldsymbol{Q}}_t \mu_t$. Suppose Assumption (iv) holds, then there exists a constant $c > 0$, such that*

$$\partial_t \mathsf{KL}(\mu_t \| \pi_t) \leq -\frac{1}{2} \mathsf{FI}_{\boldsymbol{Q}_t}(\mu_t \| \pi_t) + \frac{2}{c} \mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \partial_t \mu_t), \tag{D.44}$$

where $\mathcal{L}_{\boldsymbol{Q}}(\mu_t, \partial_t \mu_t) \geq 0$ is the Lagrangian defined as

$$\mathcal{L}_{\boldsymbol{Q}}(\mu_t, \partial_t \mu_t) = \sup_{\varphi \in C_b(\mathcal{X})} \left[\langle \varphi, \partial_t \mu_t \rangle - \sum_{\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}} \mu_t(\boldsymbol{x}_i) \boldsymbol{Q}^{[j,i]} \exp(\varphi(\boldsymbol{x}_j) - \varphi(\boldsymbol{x}_i)) \right]. \tag{D.45}$$

Proof Sketch. This lemma follows from Theorem 6.2.3. in [129]. For completeness, we provide a sketch of the proof here.

By a direct calculation of derivatives, we have

$$\begin{aligned}
\partial_t \mathsf{KL}(\mu_t \| \pi_t) &= \partial_t \sum_{\boldsymbol{x}_i} \mu_t(\boldsymbol{x}_i) \log \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} = \sum_{\boldsymbol{x}_i} \left(\partial_t \mu_t(\boldsymbol{x}_i) \log \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} - \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} \partial_t \pi_t(\boldsymbol{x}_i) \right) \\
&= \sum_{\boldsymbol{x}_i, \boldsymbol{x}_j} \left(\widetilde{\boldsymbol{Q}}_t^{[i,j]} \mu_t(\boldsymbol{x}_j) \log \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} - \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} \boldsymbol{Q}_t^{[i,j]} \pi_t(\boldsymbol{x}_j) \right) \\
&= \sum_{\boldsymbol{x}_i, \boldsymbol{x}_j} \boldsymbol{Q}_t^{[i,j]} \pi_t(\boldsymbol{x}_j) \left(\frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} \log \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} - \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} \right) \\
&\quad - \sum_{\boldsymbol{x}_i, \boldsymbol{x}_j} \left(\boldsymbol{Q}_t^{[i,j]} - \widetilde{\boldsymbol{Q}}_t^{[i,j]} \right) \mu_t(\boldsymbol{x}_j) \log \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)}.
\end{aligned}$$

Using the fact that $\sum_{\boldsymbol{x}_i} \boldsymbol{Q}_t^{[i,j]} = 0$ and the definition in Equation (D.42), we have the equality

$$\sum_{\boldsymbol{x}_i, \boldsymbol{x}_j} \boldsymbol{Q}_t^{[i,j]} \pi_t(\boldsymbol{x}_j) \left(\frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} \log \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} - \frac{\mu_t(\boldsymbol{x}_i)}{\pi_t(\boldsymbol{x}_i)} \right) = -\mathsf{FI}_{\boldsymbol{Q}_t}(\mu_t \| \pi_t).$$

Using the relation $\partial_t \mu_t = \widetilde{\boldsymbol{Q}}_t \mu_t$ then leads to

$$\partial_t \mathsf{KL}(\mu_t \| \pi_t) = -\mathsf{FI}_{\boldsymbol{Q}_t}(\mu_t \| \pi_t) - \underbrace{\left(\log \frac{\mu_t}{\pi_t} \right)^T (\boldsymbol{Q}_t - \partial_t) \mu_t}_{\textit{error term}}. \tag{D.46}$$

This formula is similar to Lemma 1 in [333].

We define

$$\begin{aligned}\mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \partial_t\mu_t) &= \sup_{\varphi\in C_b(\mathcal{X})} [\langle\varphi, \partial_t\mu_t\rangle - (e^{-\varphi}\mu_t)^T\boldsymbol{Q}_t^T e^{\varphi}] \\ &= \sup_{\varphi\in C_b(\mathcal{X})} [\langle\varphi, \partial_t\mu_t - \boldsymbol{Q}_t\mu_t\rangle - \underbrace{\mu_t^T(e^{-\varphi}\boldsymbol{Q}_t^T e^{\varphi} - \boldsymbol{Q}_t^T\varphi)}_{\text{denoted as } \widetilde{\mathcal{H}}(\mu_t,\varphi)}]. \end{aligned} \tag{D.47}$$

By the variational characterization of the Lagrangian, for any continuous, bounded φ,

$$\langle\varphi, \partial_t\mu_t - \boldsymbol{Q}_t\mu_t\rangle \le \mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \partial_t\mu_t) + \widetilde{\mathcal{H}}(\mu_t, \varphi). \tag{D.48}$$

If we choose $\varphi = \log\frac{\mu_t}{\pi_t}$, the inequality above gives a bound to the error term in Equation (D.46). Moreover, it is easy to verify that

$$\widetilde{\mathcal{H}}\left(\mu_t, \log\frac{\mu_t}{\pi_t}\right) = \mathsf{FI}(\mu_t\|\pi_t). \tag{D.49}$$

In fact, when the space $\mathcal{X}$ is continuous, choosing $\varphi(\boldsymbol{x}) = \log\frac{\mu_t}{\pi_t}(\boldsymbol{x})$ exactly recovers Lemma A.4 in [320]. However, as pointed out in [129], it is necessary to consider a rescaled $\varphi = \lambda\log\frac{\mu_t}{\pi_t}$ with $\lambda \in (0, 1)$ to derive a bound in finite space $\mathcal{X}$.

Fortunately, according to Lemma 6.2.2. in [129], for any $\varphi \in C_b(\mathcal{X})$ with $\|\varphi\|_\infty \le B$, there exists a positive constant $c = c(B) \in (0, 1)$, such that

$$\widetilde{\mathcal{H}}(\mu_t, \lambda\varphi) \le \frac{\lambda^2}{c}\widetilde{\mathcal{H}}(\mu_t, \varphi). \tag{D.50}$$

Plugging in $\varphi = \lambda\log\frac{\mu_t}{\pi_t}$ in Equation (D.48), we have

$$\left\langle\lambda\log\frac{\mu_t}{\pi_t}, \partial_t\mu_t - \boldsymbol{Q}_t\mu_t\right\rangle \le \mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \partial_t\mu_t) + \widetilde{\mathcal{H}}\left(\mu_t, \lambda\log\frac{\mu_t}{\pi_t}\right) \tag{D.51}$$

$$\le \mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \partial_t\mu_t) + \frac{\lambda^2}{c}\widetilde{\mathcal{H}}\left(\mu_t, \log\frac{\mu_t}{\pi_t}\right). \tag{D.52}$$

Combining this with Equation (D.46), we get

$$\begin{aligned}\partial_t\mathsf{KL}(\mu_t\|\pi_t) &\le -\mathsf{FI}_{\boldsymbol{Q}_t}(\mu_t\|\pi_t) + \frac{1}{\lambda}\mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \partial_t\mu_t) + \frac{\lambda}{c}\widetilde{\mathcal{H}}\left(\mu_t, \log\frac{\mu_t}{\pi_t}\right) \\ &= -(1 - \frac{\lambda}{c})\mathsf{FI}_{\boldsymbol{Q}_t}(\mu_t\|\pi_t) + \frac{1}{\lambda}\mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \partial_t\mu_t). \end{aligned} \tag{D.53}$$

Finally, choosing $\lambda = \frac{c}{2} \in (0, 1)$ recovers the statement of Lemma D.4.2.

□

Remark D.4.3. The Lagrangian $\mathcal{L}_{\boldsymbol{Q}}(\mu_t, \partial_t \mu_t) = 0$ if and only if $\partial_t \mu_t = \boldsymbol{Q}\mu_t$.

Lemma D.4.4. *When the learned score function $\boldsymbol{s}_{\boldsymbol{\theta}}$ satisfies Assumption (i),*

$$\mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \widetilde{\boldsymbol{Q}}_t \mu_t) \leq M\epsilon. \tag{D.54}$$

Proof. By definition,

$$\begin{aligned}
&\mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \widetilde{\boldsymbol{Q}}_t \mu_t) \\
&= \sup_{\varphi \in C_b(\mathcal{X})} \sum_{\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}} \left[\mu_t(\boldsymbol{x}_i) \widetilde{\boldsymbol{Q}}_t^{[j,i]} \varphi(\boldsymbol{x}_j) - \mu_t(\boldsymbol{x}_i) \boldsymbol{Q}_t^{[j,i]} e^{\varphi(\boldsymbol{x}_j) - \varphi(\boldsymbol{x}_i)} \right] \\
&= \sup_{\varphi \in C_b(\mathcal{X})} \sum_{\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}} \left[\mu_t(\boldsymbol{x}_i) \widetilde{\boldsymbol{Q}}_t^{[j,i]} (\varphi(\boldsymbol{x}_j) - \varphi(\boldsymbol{x}_i)) - \mu_t(\boldsymbol{x}_i) \boldsymbol{Q}_t^{[j,i]} e^{\varphi(\boldsymbol{x}_j) - \varphi(\boldsymbol{x}_i)} \right] \\
&\qquad\qquad (\mathbf{1}^T \widetilde{\boldsymbol{Q}}_t \boldsymbol{u} = 0 \text{ for any } \boldsymbol{u}) \\
&= \sup_{\varphi \in C_b(\mathcal{X})} \sum_{\boldsymbol{x}_i \neq \boldsymbol{x}_j} \left[\mu_t(\boldsymbol{x}_i) \boldsymbol{Q}_t^{[j,i]} \left(-\frac{\widetilde{\boldsymbol{Q}}_t^{[j,i]}}{\boldsymbol{Q}_t^{[j,i]}} z_{ij} - e^{-z_{ij}} \right) \right] + \sum_{\boldsymbol{x}_i} \mu_t(\boldsymbol{x}_i) (\widetilde{\boldsymbol{Q}}_t^{[i,i]} - \boldsymbol{Q}_t^{[i,i]}) \\
&\qquad\qquad (z_{ij} := \varphi(\boldsymbol{x}_i) - \varphi(\boldsymbol{x}_j)) \\
&\leq \sum_{\boldsymbol{x}_i \neq \boldsymbol{x}_j} \left[\mu_t(\boldsymbol{x}_i) \boldsymbol{Q}_t^{[j,i]} \sup_{z_{ij} \in \mathbb{R}} \left(-\frac{\widetilde{\boldsymbol{Q}}_t^{[j,i]}}{\boldsymbol{Q}_t^{[j,i]}} z_{ij} - e^{-z_{ij}} \right) \right] + \epsilon \sum_{\boldsymbol{x}_i} \mu_t(\boldsymbol{x}_i) \left| \boldsymbol{Q}_t^{[i,i]} \right|
\end{aligned} \tag{D.55}$$

where the inequality is due to swapping supremum and summation, and that $\mu_t(\boldsymbol{x}_i) \boldsymbol{Q}_t^{[j,i]} \geq 0$ for $i \neq j$.

Consider the function $g(z) = -uz - e^{-z}$. When $u \geq 0$, this function is maximized when $z = -\log u$. So, $-uz - e^{-z} \leq (u-1)\log u$. Setting $u = \widetilde{\boldsymbol{Q}}_t^{[j,i]} / \boldsymbol{Q}_t^{[j,i]} = \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i; t)_{\boldsymbol{x}_j} / \boldsymbol{s}(\boldsymbol{x}_i; t)_{\boldsymbol{x}_j}$, we can invoke Assumption (i) to obtain

$$\sup_{z_{ij} \in \mathbb{R}} \left(-\frac{\widetilde{\boldsymbol{Q}}_t^{[j,i]}}{\boldsymbol{Q}_t^{[j,i]}} z_{ij} - e^{-z_{ij}} \right) \leq (u-1)\log u \leq |u-1| = \left| \frac{\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i; t)_{\boldsymbol{x}_j} - \boldsymbol{s}(\boldsymbol{x}_i; t)_{\boldsymbol{x}_j}}{\boldsymbol{s}(\boldsymbol{x}_i; t)_{\boldsymbol{x}_j}} \right| \leq \epsilon \tag{D.56}$$

for $\left| \frac{\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i; t)_{\boldsymbol{x}_j} - \boldsymbol{s}(\boldsymbol{x}_i; t)_{\boldsymbol{x}_j}}{\boldsymbol{s}(\boldsymbol{x}_i; t)_{\boldsymbol{x}_j}} \right| \leq 1$. Combining with Equation (D.55), we have

$$\mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \widetilde{\boldsymbol{Q}}_t \mu_t) \leq \epsilon \mathbf{1}^T |\boldsymbol{Q}_t| \mu_t \leq \epsilon \| |\boldsymbol{Q}_t| \|_1 \| \mu_t \|_1 \leq \epsilon M. \tag{D.57}$$

□

Remark D.4.5. The Lagrangian $\mathcal{L}_{\boldsymbol{Q}}(\mu_t, \partial_t \mu_t)$ can potentially be bounded by the score entropy defined in Definition 3.1 of [198]. To see the this, we calculate

$$
\begin{aligned}
&\mathcal{L}_{\boldsymbol{Q}_t}(\mu_t, \widetilde{\boldsymbol{Q}}_t \mu_t) \\
&= \sup_{\varphi \in C_b(\mathcal{X})} \sum_{\boldsymbol{x}_i \neq \boldsymbol{x}_j} \left[\mu_t(\boldsymbol{x}_i) \boldsymbol{Q}_t^{[j,i]} \left(-\frac{\widetilde{\boldsymbol{Q}}_t^{[j,i]}}{\boldsymbol{Q}_t^{[j,i]}} z_{ij} - e^{-z_{ij}} \right) \right] + \sum_{\boldsymbol{x}_i} \mu_t(\boldsymbol{x}_i) (\widetilde{\boldsymbol{Q}}_t^{[i,i]} - \boldsymbol{Q}_t^{[i,i]}) \\
&\overset{(1)}{\leq} \sum_{\boldsymbol{x}_i \neq \boldsymbol{x}_j} \left[\mu_t(\boldsymbol{x}_i) \boldsymbol{Q}_t^{[j,i]} \sup_{z_{ij} \in \mathbb{R}} \left(-\frac{\widetilde{\boldsymbol{Q}}_t^{[j,i]}}{\boldsymbol{Q}_t^{[j,i]}} z_{ij} - e^{-z_{ij}} \right) \right] + \sum_{\boldsymbol{x}_i} \mu_t(\boldsymbol{x}_i) (\widetilde{\boldsymbol{Q}}_t^{[i,i]} - \boldsymbol{Q}_t^{[i,i]}) \\
&\overset{(2)}{\leq} \sum_{\boldsymbol{x}_i \neq \boldsymbol{x}_j} \left[\mu_t(\boldsymbol{x}_i) \boldsymbol{Q}_t^{[j,i]} \frac{\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}}{\boldsymbol{s}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}} \log \frac{\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}}{\boldsymbol{s}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}} \right] - \sum_{\boldsymbol{x}_i} \mu_t(\boldsymbol{x}_i) \sum_{j \neq i} (\widetilde{\boldsymbol{Q}}_t^{[j,i]} - \boldsymbol{Q}_t^{[j,i]}) \\
&\overset{(3)}{=} \sum_{\boldsymbol{x}_i \neq \boldsymbol{x}_j} \left[\mu_t(\boldsymbol{x}_i) \boldsymbol{Q}_t^{\mathrm{fw}[i,j]} \left(\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j} \log \frac{\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}}{\boldsymbol{s}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}} - \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j} + \boldsymbol{s}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j} \right) \right] \\
&\overset{(4)}{\leq} \mathbb{E}_{\boldsymbol{x}_i \sim \mu_t} \sum_{\boldsymbol{x}_i \neq \boldsymbol{x}_j} \left[K \left(\frac{\boldsymbol{s}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}}{\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}} \right) \boldsymbol{Q}_t^{\mathrm{fw}[i,j]} \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j} \right]
\end{aligned}
$$

where (1) is due to swapping supremum and summation, (2) is due to Equation (D.56) and the property of transition-rate matrices, (3) is due to the definitions of $\boldsymbol{Q}_t$ and $\widetilde{\boldsymbol{Q}}_t$, and (4) is due to the definition of $K(x) := x - \log x - 1$. Note that the last line is the score entropy of $\boldsymbol{s}$ with respect to $\boldsymbol{s}_{\boldsymbol{\theta}}$ defined as

$$
\mathsf{SE}_{\boldsymbol{Q}_t^{\mathrm{fw}}}(\boldsymbol{s}(\,\cdot\,;\, t) || \boldsymbol{s}_{\boldsymbol{\theta}}(\,\cdot\,;\, t)) := \sum_{\boldsymbol{x}_i \neq \boldsymbol{x}_j} \left[K \left(\frac{\boldsymbol{s}_{\theta}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}}{\boldsymbol{s}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j}} \right) \mathbf{Q}_t^{\mathrm{fw}[i,j]} \boldsymbol{s}(\boldsymbol{x}_i;\, t)_{\boldsymbol{x}_j} \right].
$$

Therefore, by swapping μ_t with π_t and $\boldsymbol{Q}_t$ with $\widetilde{\boldsymbol{Q}}_t$, we have

$$
\mathcal{L}_{\widetilde{\boldsymbol{Q}}_t}(\pi_t, \partial_t \pi_t) = \mathcal{L}_{\widetilde{\boldsymbol{Q}}_t}(\pi_t, \boldsymbol{Q}_t \pi_t) \leq \mathbb{E}_{\boldsymbol{x}_i \sim \pi_t} \mathsf{SE}_{\boldsymbol{Q}_t^{\mathrm{fw}}}(\boldsymbol{s}_{\boldsymbol{\theta}}(\,\cdot\,;\, t) || \boldsymbol{s}(\,\cdot\,;\, t)) \tag{D.58}
$$

where $\mathsf{SE}_{\boldsymbol{Q}_t^{\mathrm{fw}}}(\boldsymbol{s}_{\boldsymbol{\theta}}(\,\cdot\,;\, t) || \boldsymbol{s}(\,\cdot\,;\, t))$ is the score entropy of $\boldsymbol{s}_{\boldsymbol{\theta}}$ with respect to $\boldsymbol{s}$, which is the loss function for training the score entropy estimator $\boldsymbol{s}_{\boldsymbol{\theta}}$. This implies that if we swap μ_t with π_t and $\boldsymbol{Q}_t$ with $\widetilde{\boldsymbol{Q}}_t$, we can potentially substitute Assumption (i) and Assumption (v) with a more interpretable assumption on the score entropy error and obtain a convergence result that explicitly incorporates this error, which would resemble that of discrete DMs for unconditional generation [245]. Nevertheless, we underscore that the right-hand side of Equation (D.58) is an integration over the stationary process π_t, which may be different from the path on which the DM was trained. This is in fact expected and highlights the difference between analyzing *the generative process* of discrete DMs versus analyzing *the process of using discrete DMs for solving inverse problems*. The ideal process in the generative process of

discrete DMs is the same as the process for training (only time going in different directions). In contrast, when using discrete DMs as priors for solving inverse problems, they conduct inference on arbitrary distributions that may be different from the training distributions, which is a generalization gap that requires stronger condition on the DMs for convergence. For the purpose of this chapter, we proceed with Lemma D.4.4 to simply the subsequent discretization analysis.

D.4.1.2 Proof of Theorem 7.6.1

Proof. We denote the time index in this proof by τ to align with the time interpolation of K iterations of SGDD introduced earlier. Consider discretizing a prior step $[0, t^*]$ uniformly into H intervals, $[h\delta, (h+1)\delta]$, for $h = 0, \ldots, H-1$, where $\delta = t^*/H$. In the h-th interval, we have

$$\partial_\tau \mu_\tau = \widetilde{\boldsymbol{Q}}_{h\delta} \mu_\tau. \tag{D.59}$$

Applying Lemma D.4.2 to the h-th interval for μ_τ and π_τ, we get

$$\partial_\tau \mathsf{KL}(\mu_\tau \| \pi_\tau) \leq -\frac{1}{2} \mathsf{FI}_{\boldsymbol{Q}_\tau}(\mu_\tau \| \pi_\tau) + \frac{2}{c} \mathcal{L}_{\boldsymbol{Q}_\tau}(\mu_\tau, \widetilde{\boldsymbol{Q}}_{h\delta} \mu_\tau). \tag{D.60}$$

Denoting the starting time of the k-th prior step as $T_k = k(t^*+1)+1$, we then integrate both sides by τ and obtain

$$\begin{aligned}
\int_{T_k+h\delta}^{T_k+(h+1)\delta} \mathsf{FI}_{\boldsymbol{Q}\tau}(\mu_\tau \| \pi_\tau) \mathrm{d}\tau &\leq \int_{T_k+h\delta}^{T_k+(h+1)\delta} \left[-2\partial_\tau \mathsf{KL}(\mu_\tau \| \pi_\tau) + \frac{4}{c} \mathcal{L}_{\boldsymbol{Q}_\tau}(\mu_\tau, \widetilde{\boldsymbol{Q}}_{T_k+h\delta} \mu_\tau) \right] \mathrm{d}\tau \\
&= 2 \left[\mathsf{KL}(\mu_{T_k+h\delta} \| \pi_{T_k+h\delta}) - \mathsf{KL}(\mu_{T_k+(h+1)\delta} \| \pi_{T_k+(h+1)\delta}) \right] \\
&\quad + \frac{4}{c} \int_{T_k+h\delta}^{T_k+(h+1)\delta} \mathcal{L}_{\boldsymbol{Q}_\tau}(\mu_\tau, \widetilde{\boldsymbol{Q}}_{T_k+h\delta} \mu_\tau) \mathrm{d}\tau.
\end{aligned} \tag{D.61}$$

Taking a summation over h gives us

$$\begin{aligned}
\int_{T_k}^{T_k+t^*} \mathsf{FI}_{\boldsymbol{Q}_\tau}(\mu_\tau \| \pi_\tau) \mathrm{d}\tau &\leq 2 \left[\mathsf{KL}(\mu_{T_k} \| \pi_{T_k}) - \mathsf{KL}(\mu_{T_k+t^*} \| \pi_{T_k+t^*}) \right] \\
&\quad + \sum_{h=0}^{H-1} \int_{T_k+h\delta}^{T_k+(h+1)\delta} \frac{4}{c} \mathcal{L}_{\boldsymbol{Q}_\tau}(\mu_\tau, \widetilde{\boldsymbol{Q}}_{T_k+h\delta} \mu_\tau) \mathrm{d}\tau.
\end{aligned} \tag{D.62}$$

Therefore, by Lemma D.4.4, we have

$$\int_{T_k+h\delta}^{T_k+(h+1)\delta} \frac{4}{c} \mathcal{L}_{\boldsymbol{Q}_\tau}(\mu_\tau, \widetilde{\boldsymbol{Q}}_{T_k+h\delta} \mu_\tau) \mathrm{d}\tau \leq \int_0^\delta \frac{4}{c} M(\epsilon + Ls) \mathrm{d}s = \frac{4M}{c} \left(\frac{\epsilon t^*}{H} + \frac{L t^{*2}}{2H^2} \right). \tag{D.63}$$

where we used the fact that $\left\| \frac{s_\theta(\cdot;t+\Delta t)-s(\cdot;t)}{s(\cdot;t)} \right\|_\infty \leq \epsilon + L \cdot \Delta t$ due to both Assumption (i) and Assumption (ii). It follows that

$$\int_{T_k}^{T_k+t^*} \mathsf{FI}_{\boldsymbol{Q}_\tau}(\mu_\tau \| \pi_\tau) \mathrm{d}\tau \leq 2 \left[\mathsf{KL}(\mu_{T_k} \| \pi_{T_k}) - \mathsf{KL}(\mu_{T_k+t^*} \| \pi_{T_k+t^*}) \right] + \frac{4M}{c} \left(\epsilon t^* + \frac{L t^{*2}}{2H} \right).$$

Finally, taking summation over $k = 0, \ldots, K-1$ and applying Lemma D.4.1 to each likelihood step, we have

$$\sum_{k=0}^{K-1} \int_{T_k}^{T_k+t^*} \mathsf{FI}_{\boldsymbol{Q}_\tau}(\mu_\tau \| \pi_\tau) \mathrm{d}\tau \leq 2\mathsf{KL}(\mu_0 \| \pi_0) + \frac{4M}{c} \left(\epsilon K t^* + \frac{L K t^{*2}}{2H} \right).$$

Dividing by Kt^* on both sides gives us

$$\frac{1}{K} \sum_{k=0}^{K-1} \frac{1}{t^*} \int_{T_k}^{T_k+t^*} \mathsf{FI}_{\boldsymbol{Q}_\tau}(\mu_\tau \| \pi_\tau) \mathrm{d}\tau \leq \frac{2\mathsf{KL}(\mu_0 \| \pi_0)}{K t^*} + \frac{4M\epsilon}{c} + \frac{2MLt^*}{cH}. \tag{D.64}$$

□

D.4.1.3 Potential Function of Split Gibbs Samplers

As defined in Equation (7.21), the Split Gibbs Sampler draws samples from the augmented distribution

$$\pi(\boldsymbol{x}, \boldsymbol{z}; \eta) \propto \exp(-f(\boldsymbol{z}; \boldsymbol{y}) - g(\boldsymbol{x}) - D(\boldsymbol{x}, \boldsymbol{z}; \eta)).$$

The key requirement of split Gibbs samplers is that the potential function $D(\boldsymbol{x}, \boldsymbol{z}; \eta)$ satisfies

$$\lim_{\eta \to 0^+} D(\boldsymbol{x}, \boldsymbol{z}; \eta) = \infty, \forall \boldsymbol{x} \neq \boldsymbol{z}. \tag{D.65}$$

Or more precisely,

$$\lim_{\eta \to 0^+} \frac{\exp(-D(\boldsymbol{x}, \boldsymbol{z}; \eta))}{\int \exp(-D(\boldsymbol{x}, \boldsymbol{z}; \eta)) \mathrm{d}\boldsymbol{z}} = \delta_{\boldsymbol{x}}(\boldsymbol{z}). \tag{D.66}$$

Therefore,

$$\begin{aligned} \lim_{\eta \to 0^+} \pi^X(\boldsymbol{x}; \eta) &\propto \lim_{\eta \to 0^+} \int p(\boldsymbol{x}) p(\boldsymbol{y} \mid \boldsymbol{z}) \exp(-D(\boldsymbol{x}, \boldsymbol{z}; \eta)) \mathrm{d}\boldsymbol{z} \\ &= \int p(\boldsymbol{x}) p(\boldsymbol{y} \mid \boldsymbol{z}) \delta(\boldsymbol{x} - \boldsymbol{z}) \mathrm{d}\boldsymbol{z} = p(\boldsymbol{x}) p(\boldsymbol{y} \mid \boldsymbol{x}). \end{aligned} \tag{D.67}$$

Also, the similar derivation holds for $\pi^Z(\boldsymbol{z}; \eta)$. This result has also been shown in [296]. Combining this with Theorem 7.6.1, SGDD is guaranteed to sample from the true posterior distribution when η goes to zero.

D.4.2 Experimental Details

D.4.2.1 Pre-Trained Models

We learn prior distributions for each dataset using SEDD [198] discrete diffusion models. We use the SEDD small architecture with around 90M parameters for all experiments, and the models are trained with AdamW [197] with batch size 32 and a learning rate of 3×10^{-4}.

D.4.2.2 Baseline Methods

DPS DPS [64] is designed to solve general inverse problems with a pre-trained (continuous) diffusion model. It performs posterior sampling from $p(\boldsymbol{x} \mid \boldsymbol{y})$ by modifying the reverse SDE

$$\mathrm{d}\boldsymbol{x}_t = -2\dot{\sigma}_t\sigma_t\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t \mid \boldsymbol{y})\mathrm{d}t + \sqrt{2\dot{\sigma}_t\sigma_t}\mathrm{d}t \tag{D.68}$$

$$= -2\dot{\sigma}_t\sigma_t(\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{x}_t;\sigma_t)) + \nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{y} \mid \boldsymbol{x}_t))\mathrm{d}t + \sqrt{2\dot{\sigma}_t\sigma_t}\mathrm{d}t. \tag{D.69}$$

To estimate the intractable guidance term $\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{y} \mid \boldsymbol{x}_t)\mathrm{d}t$, DPS proposes to approximate it with $p(\boldsymbol{y} \mid \boldsymbol{x}_t) \approx p(\boldsymbol{y} \mid \mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t])$. The guidance term is thus approximated by

$$\nabla_{\boldsymbol{x}_t}\log p(\boldsymbol{y}|\boldsymbol{x}_t) \approx -\nabla_{\boldsymbol{x}_t}\frac{\|\mathcal{A}(\widehat{\boldsymbol{x}}_0(\boldsymbol{x}_t;\sigma_t)) - \boldsymbol{y}\|_2^2}{2\sigma_{\boldsymbol{y}}^2}, \tag{D.70}$$

where $\widehat{\boldsymbol{x}}_0(\boldsymbol{x}_t;\sigma_t)$ is a one-step approximation of $\mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t]$ using the pre-trained diffusion model, and the measurement is assumed to be $\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}) + \boldsymbol{n}$ with $\boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, \sigma_{\boldsymbol{y}}^2\boldsymbol{I})$. However, DPS is not directly applicable to inverse problems in discrete-state spaces since propagating gradients through $\mathcal{A}$ and D_θ in Equation (D.70) is impossible. Therefore, we consider the counterpart of DPS in discrete spaces. We modify the continuous-time Markov chain of the discrete diffusion model by

$$\partial_t p_{T-t} = \boldsymbol{Q}_{T-t}^{\boldsymbol{y}} p_{T-t}, \tag{D.71}$$

in which

$$\boldsymbol{Q}_{T-t}^{\boldsymbol{y}\ [i,j]} = \frac{p_{T-t}(\boldsymbol{x}_i|\boldsymbol{y})}{p_{T-t}(\boldsymbol{x}_j|\boldsymbol{y})} = \frac{p_{T-t}(\boldsymbol{x}_i)}{p_{T-t}(\boldsymbol{x}_j)}\frac{p_{T-t}(\boldsymbol{y}|\boldsymbol{x}_i)}{p_{T-t}(\boldsymbol{y}|\boldsymbol{x}_j)} = \boldsymbol{Q}_{T-t}^{[i,j]} \cdot \frac{p_{T-t}(\boldsymbol{y}|\boldsymbol{x}_i)}{p_{T-t}(\boldsymbol{y}|\boldsymbol{x}_j)}. \tag{D.72}$$

Similar ideas are applied to classifier guidance for discrete diffusion models [225], where the matrix $\boldsymbol{R}_t^{\boldsymbol{y}} = \left[\frac{p_{T-t}(\boldsymbol{y}|\boldsymbol{x}_i)}{p_{T-t}(\boldsymbol{y}|\boldsymbol{x}_j)}\right]_{i,j}$ is called a guidance rate matrix. We compute

$\boldsymbol{R}_t^{\boldsymbol{y}}$ at $\boldsymbol{x}_j$-column by enumerating every neighboring $\boldsymbol{x}_i$ and calculating $\frac{p_t(\boldsymbol{y}|\boldsymbol{x}_i)}{p_t(\boldsymbol{y}|\boldsymbol{x}_j)}$ for each $\boldsymbol{x}_i$. The discrete version of DPS can be summarized by

$$\partial_t p_{T-t} = \boldsymbol{Q}_t \boldsymbol{R}_t^{\boldsymbol{y}} p_{T-t}. \tag{D.73}$$

However, the discrete version of DPS is very time-consuming, especially when the vocabulary size is large, since it enumerates $(N-1) \times n$ number of neighboring $\boldsymbol{x}$ when computing $\boldsymbol{R}_t^{\boldsymbol{y}}$. We find it slow for the discrete image reconstruction problems on binary MNIST where $N = 2$.

SVDD SVDD [182] aims to sample from the distribution $p_\beta(\boldsymbol{x}_0) \propto p(\boldsymbol{x}_0) \exp(\beta r(\boldsymbol{x}_0))$, which is equivalent to the regularized MDP problem:

$$p^\beta(\boldsymbol{x}_0) = \arg\max_\pi \mathbb{E}_{\boldsymbol{x}_0 \sim \pi} r(\boldsymbol{x}_0) - \mathsf{KL}(\pi \| p)/\beta. \tag{D.74}$$

They calculate the soft value function as

$$v_t(\boldsymbol{x}_t) = \log \mathbb{E}_{\boldsymbol{x}_0 \sim p(\boldsymbol{x}_0|\boldsymbol{x}_t)} [\exp(\beta r(\boldsymbol{x}_0))]/\beta \tag{D.75}$$

and propose to sample from the optimal policy

$$p_t^\star(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}) \propto p_t(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}) \exp(\beta v_t(\boldsymbol{x}_t)). \tag{D.76}$$

In time step t, SVDD samples a batch of M particles from the unconditional distribution $p_t(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1})$, and conduct importance sampling according to $\exp(\beta v_t(\boldsymbol{x}_t))$.

Although [182] is initially designed for guided diffusion generation, it also applies to solving inverse problems by carefully choosing reward functions. We consider $r(\boldsymbol{x}) = -\|\mathcal{A}(\boldsymbol{x}) - \boldsymbol{y}\|_0/\sigma_{\boldsymbol{y}}$ and $\beta = 1$, so that it samples from the posterior distribution $p(\boldsymbol{x} \mid \boldsymbol{y})$. As recommended in [182], we choose $\beta = \infty$ ($\alpha = 0$ in their notation) in practice, so the importance sampling reduces to finding the particle with the maximal value in each iteration. We use SVDD-PM, a training-free method provided by [182] in our experiments. It approximates the value function by $v_t(\boldsymbol{x}_t) = r((\boldsymbol{x}_t; \sigma_t))$, where $(\boldsymbol{x}_t; \sigma_t)$ is an approximation of $\mathbb{E}[\boldsymbol{x}_0|\boldsymbol{x}_t]$. In practice, we find that approximating $(\boldsymbol{x}_t; \sigma_t)$ by Monte Carlo sampling with a few-step Euler sampler achieves slightly better results. We use 3 Monte Carlo samples to estimate $v_t(\boldsymbol{x}_t)$ in our experiments.

SMC Sequential Monte Carlo (SMC) methods evolve multiple particles to approximate a series of distributions, eventually converging to the target distribution. Specifically, in our experiments, we implement the SMC method to sample from

$p(\boldsymbol{x}_t \mid \boldsymbol{y})$ for $t = T, T-1 \ldots, 0$, using the unconditional discrete diffusion sampler $p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1})$ as the proposal function.

We maintain a batch of $J = 20$ particles $\{\boldsymbol{x}^{(j)}\}$. At time t, we sample $\boldsymbol{x}_t^{(j)} \sim p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}^{(j)})$ by the pre-trained discrete diffusion model and estimate the likelihood $p(\boldsymbol{y} \mid \boldsymbol{x}_t^{(j)}) = \mathbb{E}_{\boldsymbol{x}_0 \sim p\left(\boldsymbol{x}_0 \mid \boldsymbol{x}_t^{(j)}\right)} p(\boldsymbol{y} \mid \boldsymbol{x}_0)$ by Monte Carlo sampling. We then resample the particles $\{\boldsymbol{x}_t^{(j)}\}$ according to their weights $w_t^{(j)} = p(\boldsymbol{y}|\boldsymbol{x}_t^{(j)})/p(\boldsymbol{y}|\boldsymbol{x}_{t+1}^{(j)})$. In practice, we find that we have to carefully tune a hyperparameter β, where w_t^{β}. Otherwise, the resampling step can easily degenerate to finding arg max or uniform random sampling.

D.4.2.3 Hyperparameters

We use an annealing noise schedule of $\eta_k = \eta_{\min}^{\frac{k}{K-1}} \eta_{\max}^{1-\frac{k}{K-1}}$ with $\eta_{\min} = 10^{-4}$ and $\eta_{\max} = 20$. We run SGDD for K iterations. In each likelihood sampling step, we run Metropolis-Hastings for T steps, while in each prior sampling step, we run a few-step Euler discrete diffusion sampler with H steps. The hyperparameters used for each experiment are listed in Table D.7. We also include the dimension of data spaces $\mathcal{X}^n$ for each experiment in Table D.7, where $|\mathcal{X}| = N$.

Table D.7: **Hyperparameters of SGDD used in each experiment.**

	MNIST XOR	MNIST AND
Metropolis-Hastings T	2000	5000
SGDD iterations K	50	100
Euler sampler H	20	20
Sequence length n	1024	1024
Vocab size N	2	2

Appendix E

APPENDIX FOR CHAPTER 8

E.1 Tables of Main Results

We present the main experimental results in Tables E.1, E.2, E.3, E.4, E.5, and E.6. **Bold** indicates the best across all PnPDP methods.

Table E.1: **Results on linear inverse scattering.** PSNR and SSIM of different algorithms on linear inverse scattering. Noise level $\sigma_y = 10^{-4}$.

Number of receivers	360			180			60		
Methods	PSNR ↑	SSIM ↑	Meas err (%) ↓	PSNR ↑	SSIM ↑	Meas err (%) ↓	PSNR ↑	SSIM ↑	Meas err (%) ↓
Traditional									
FISTA-TV	32.126 (2.139)	0.979 (0.009)	1.23 (0.25)	26.523 (2.678)	0.914 (0.040)	2.65 (0.30)	20.938 (2.513)	0.709 (0.103)	6.05 (0.65)
PnPDP									
DDRM	32.598 (1.825)	0.929 (0.012)	1.04 (0.26)	28.080 (1.516)	0.890 (0.019)	1.57 (0.39)	20.436 (1.210)	0.545 (0.037)	3.04 (0.92)
DDNM	36.381 (1.098)	0.935 (0.017)	0.78 (0.22)	35.024 (0.993)	0.895 (0.027)	0.58 (0.16)	**29.235** (3.376)	0.917 (0.022)	0.28 (0.07)
ΠGDM	27.925 (3.211)	0.889 (0.072)	2.74 (1.23)	26.412 (3.430)	0.816 (0.114)	3.66 (1.79)	20.074 (2.608)	0.540 (0.198)	6.90 (3.38)
DPS	32.061 (2.163)	0.846 (0.127)	4.35 (1.19)	31.798 (2.163)	0.862 (0.123)	4.28 (1.20)	27.372 (3.415)	0.813 (0.133)	4.53 (1.31)
LGD	27.901 (2.346)	0.812 (0.037)	1.17 (0.20)	27.837 (2.337)	0.803 (0.034)	1.06 (0.16)	20.491 (3.031)	0.552 (0.077)	1.45 (0.68)
DiffPIR	34.241 (2.310)	**0.988** (0.006)	1.11 (0.24)	34.010 (2.269)	**0.987** (0.006)	1.04 (0.23)	26.321 (3.272)	0.918 (0.028)	1.27 (0.23)
PnP-DM	33.914 (2.054)	**0.988** (0.006)	1.21 (0.25)	31.817 (2.073)	0.981 (0.008)	1.42 (0.26)	24.715 (2.874)	0.909 (0.046)	2.20 (0.34)
DAPS	34.641 (1.693)	0.957 (0.006)	1.03 (0.25)	33.160 (1.704)	0.944 (0.009)	1.11 (0.25)	25.875 (3.110)	0.885 (0.030)	1.51 (0.25)
RED-diff	**36.556** (2.292)	0.981 (0.005)	0.89 (0.23)	**35.411** (2.166)	0.984 (0.004)	0.87 (0.21)	27.072 (3.330)	**0.935** (0.037)	1.18 (0.23)
FPS	33.242 (1.602)	0.870 (0.026)	**0.70** (0.01)	29.624 (1.651)	0.710 (0.040)	**0.37** (0.01)	21.323 (1.445)	0.460 (0.030)	**0.15** (0.02)
MCGdiff	30.937 (1.964)	0.751 (0.029)	**0.70** (0.01)	28.057 (1.672)	0.631 (0.042)	0.38 (0.01)	21.004 (1.571)	0.445 (0.028)	0.21 (0.06)

Table E.2: **Results on compressed sensing MRI.** Mean and standard deviation are reported over 94 test cases.

Subsampling ratio	×4						×8					
Measurement type	Simulated (noiseless)			Raw			Simulated (noiseless)			Raw		
Methods	PSNR ↑	SSIM ↑	Data Misfit ↓	PSNR ↑	SSIM ↑	Data Misfit ↓	PSNR ↑	SSIM ↑	Data Misfit ↓	PSNR ↑	SSIM ↑	Data Misfit ↓
Traditional												
Wavelet+ℓ_1	29.45 (1.776)	0.690 (0.121)	0.306 (0.049)	26.47 (1.508)	0.598 (0.122)	31.601 (15.286)	25.97 (1.761)	0.575 (0.105)	0.318 (0.042)	24.08 (1.602)	0.511 (0.106)	22.362 (10.733)
TV	27.03 (1.635)	0.518 (0.123)	5.748 (1.283)	26.22 (1.578)	0.509 (0.123)	32.269 (15.414)	24.12 (1.900)	0.432 (1.112)	5.087 (1.049)	23.70 (1.857)	0.427 (0.112)	23.048 (10.854)
End-to-end												
Residual U-Net	32.27 (1.810)	0.808 (0.080)	–	31.70 (1.970)	0.785 (0.095)	–	29.75 (1.675)	0.750 (0.088)	—	29.36 (1.746)	0.733 (0.100)	–
E2E-VarNet	33.40 (2.097)	0.836 (0.079)	–	31.71 (2.540)	0.756 (0.102)	–	30.67 (1.761)	0.769 (0.085)	—	30.45 (1.940)	0.736 (0.103)	–
PnPDP												
CSGM	28.78 (6.173)	0.710 (0.147)	**1.518** (0.433)	25.17 (6.246)	0.582 (0.167)	31.642 (15.382)	26.15 (6.383)	0.625 (0.158)	1.142 (1.078)	21.17 (8.314)	0.425 (0.192)	**22.088** (10.740)
ScoreMRI	25.97 (1.681)	0.468 (0.087)	10.828 (1.731)	25.60 (1.618)	0.463 (0.086)	33.697 (15.209)	25.01 (1.526)	0.405 (0.079)	8.360 (1.381)	24.74 (1.481)	0.403 (0.080)	24.028 (10.663)
RED-diff	29.36 (7.710)	0.733 (0.131)	0.509 0.077)	**28.71** (2.755)	0.626 (0.126)	31.591 (15.368)	26.76 (6.696)	0.647 (0.124)	**0.485** (0.068)	**27.33** (2.441)	0.563 (0.117)	22.336 (10.838)
DiffPIR	28.31 (1.598)	0.632 (0.107)	10.545 (2.466)	27.60 (1.470)	0.624 (0.111)	34.015 (15.522)	26.78 (1.556)	0.588 (0.113)	7.787 (1.741)	26.26 (1.458)	0.590 (0.113)	24.208 (10.922)
DPS	26.13 (4.247)	0.620 (0.105)	9.900 (2.925)	25.83 (2.197)	0.548 (0.116)	35.095 (15.967)	20.82 (4.777)	0.536 (0.111)	6.737 (1.928)	23.00 (3.205)	0.507 (0.109)	24.842 (11.263)
DAPS	31.48 (1.988)	0.762 (0.089)	1.566 (0.390)	28.61 (2.197)	**0.689** (0.102)	**31.115** (15.497)	29.01 (1.712)	0.681 (0.098)	1.280 (0.301)	27.10 (2.034)	**0.629** (0.107)	22.729 (10.926)
PnP-DM	**31.80** (3.473)	**0.780** (0.096)	4.701 (0.675)	27.62 (3.425)	0.679 (0.117)	32.261 (15.169)	**29.33** (3.081)	**0.704** (0.105)	3.421 (0.504)	25.28 (3.102)	0.607 (0.117)	22.879 (10.712)

E.1.1 Extended Evaluation of CS-MRI

For compressed sensing MRI, achieving good performance on distortion metrics such as PSNR and SSIM is not always a sufficient signal for high-quality reconstruction, as hallucinations might lead to wrong diagnoses. We quantify the degree of hallucination by employing a pathology detector on the reconstructed images of different methods. Specifically, we finetune a medium-size YOLOv11 model [159] on a training set of fully sampled images with the fastMRI+ pathology annotations [354] (22 classes in total). We calculate the mAP50 metric over the reconstructed results

Table E.3: **Generalization results on compressed sensing MRI with ×4 acceleration and raw measurements.** Mean and standard deviation are reported over 94 test cases.

Generalization	Vertical → Horizontal			Knee → Brain			×4 → ×8		
Methods	PSNR ↑	SSIM ↑	Data Misfit ↓	PSNR ↑	SSIM ↑	Data Misfit ↓	PSNR ↑	SSIM ↑	Data Misfit ↓
Traditional									
Wavelet+ℓ_1	27.75 (1.683)	0.627 (0.133)	31.744 (15.362)	25.96 (1.253)	0.747 (0.026)	7.986 (0.965)	24.08 (1.602)	0.511 (0.106)	22.362 (10.733)
TV	28.18 (1.777)	0.533 (0.138)	32.311 (15.487)	25.56 (1.302)	0.686 (0.049)	8.396 (0.990)	23.70 (1.857)	0.427 (0.112)	23.048 (10.854)
End-to-end									
Residual U-Net	22.06 (1.682)	0.603 (0.049)	–	30.07 (1.364)	0.881 (0.019)	–	23.93 (2.176)	0.610 (0.064)	–
E2E-VarNet	22.13 (2.925)	0.543 (0.103)	–	31.97 (1.452)	0.857 (0.038)	–	24.59 (2.012)	0.637 (0.069)	–
PnPDP									
CSGM	26.56 (3.647)	0.629 (0.129)	31.866 (15.479)	27.19 (7.521)	0.779 (0.189)	7.779 (1.043)	21.17 (8.314)	0.425 (0.192)	**22.088** (10.740)
ScoreMRI	25.60 (1.647)	0.473 (0.091)	33.707 (15.274)	28.52 (0.885)	0.674 (0.045)	9.472 (0.948)	24.74 (1.481)	0.403 (0.080)	24.028 (10.663)
RED-diff	**28.95** (2.480)	0.628 (0.126)	**31.740** (15.421)	**30.61** (0.982)	0.811 (0.048)	**7.750** (0.996)	**27.33** (2.441)	0.563 (0.117)	22.336 (10.838)
DiffPIR	27.93 (1.502)	0.637 (0.113)	34.188 (15.479)	27.75 (0.854)	0.823 (0.026)	10.972 (1.016)	26.26 (1.458)	0.590 (0.113)	24.208 (10.922)
DPS	26.77 (1.546)	0.571 (0.117)	35.233 (16.006)	26.77 (1.137)	0.738 (0.031)	10.806 (1.159)	23.00 (3.205)	0.507 (0.109)	24.842 (11.263)
DAPS	28.78 (2.209)	**0.696** (0.105)	32.198 (15.538)	29.29 (0.911)	0.882 (0.025)	8.255 (0.986)	27.10 (2.034)	**0.629** (0.107)	22.729 (10.926)
PnP-DM	27.93 (3.444)	0.689 (0.121)	32.391 (15.235)	29.96 (0.984)	**0.882** (0.028)	8.789 (0.978)	25.28 (3.102)	0.607 (0.117)	22.879 (10.712)

Table E.4: **Results on black hole imaging.** PSNR and Chi-squared of different algorithms on black hole imaging. Gain and phase noise and thermal noise are added based on the EHT library.

Observation time ratio	3%				10%				100%			
Methods	PSNR ↑	Blur PSNR ↑	$\bar{\chi}^2_{\text{cp}}$	$\bar{\chi}^2_{\text{logca}}$	PSNR ↑	Blur PSNR ↑	$\bar{\chi}^2_{\text{cp}}$	$\bar{\chi}^2_{\text{logca}}$	PSNR ↑	Blur PSNR ↑	$\bar{\chi}^2_{\text{cp}}$	$\bar{\chi}^2_{\text{logca}}$
Traditional												
SMILI	18.51 (1.39)	23.08 (2.12)	1.478 (0.428)	4.348 (3.827)	20.85 (2.90)	25.24 (3.86)	1.209 (0.169)	21.788 (12.491)	22.67 (3.13)	27.79 (4.02)	1.878 (0.952)	17.612 (10.299)
EHT-Imaging	21.72 (3.39)	25.66 (5.04)	**1.507** (0.485)	1.695 (0.539)	22.67 (3.46)	26.66 (3.93)	**1.166** (0.156)	**1.240** (0.205)	24.28 (3.63)	28.57 (4.52)	**1.251** (0.250)	1.259 (0.316)
PnPDP												
DPS	24.20 (3.72)	**30.83** (5.58)	8.024 (24.336)	5.007 (5.750)	24.36 (3.72)	30.79 (5.75)	13.052 (43.087)	6.614 (26.789)	25.86 (3.90)	**32.94** (6.19)	8.759 (37.784)	5.456 (24.185)
LGD	22.51 (3.76)	28.50 (5.49)	15.825 (16.838)	12.862 (12.663)	22.08 (3.75)	27.48 (5.09)	10.775 (21.684)	13.375 (56.397)	21.22 (3.64)	26.06 (4.98)	13.239 (17.231)	13.233 (39.107)
RED-diff	20.74 (2.62)	26.10 (3.35)	6.713 (6.925)	9.128 (19.052)	22.53 (3.02)	27.67 (4.53)	2.488 (2.925)	4.916 (13.221)	23.77 (4.13)	29.13 (6.22)	1.853 (0.938)	2.050 (2.361)
PnPDM	**24.25** (3.45)	30.49 (4.93)	2.201 (1.352)	**1.668** (0.551)	**24.57** (3.47)	**30.80** (5.22)	1.433 (0.417)	1.336 (0.478)	**26.07** (3.70)	32.88 (6.02)	1.311 (0.195)	**1.199** (0.221)
DAPS	23.54 (3.28)	29.48 (4.88)	3.647 (3.287)	2.329 (1.354)	23.99 (3.56)	30.10 (5.13)	1.545 (0.705)	2.253 (9.903)	25.60 (3.64)	32.78 (5.68)	1.300 (0.324)	1.229 (0.532)
DiffPIR	24.12 (3.25)	30.45 (4.88)	14.085 (14.105)	10.545 (8.860)	23.84 (3.59)	30.04 (5.03)	5.374 (3.733)	5.205 (5.556)	25.01 (4.64)	31.86 (6.56)	3.271 (1.623)	2.970 (1.202)

Table E.5: **Results on FWI.** Mean and standard deviation are reported over 10 test cases. †: initialized from data blurred by Gaussian filters with $\sigma = 20$. $*$: one test case is excluded from the results due to numerical instability.

Methods	Relative ℓ_2 ↓	PSNR ↑	SSIM ↑	Data Misfit ↓
Traditional				
Adam	0.333 (0.086)	9.968 (2.083)	0.305 (0.120)	115.14 (52.10)
Adam†	0.089 (0.021)	21.273 (2.045)	0.679 (0.073)	15.89 (10.16)
LBFGS†	0.070 (0.023)	23.398 (2.749)	0.704 (0.077)	9.18 (6.47)
PnPDP				
DPS	0.250 (0.154)	14.111 (6.820)	0.491 (0.161)	155.08 (92.17)
LGD	0.244 (0.024)	12.288 (0.889)	0.341 (0.047)	258.47 (26.40)
DiffPIR	0.204 (0.129)	**16.113** (6.962)	**0.554** (0.191)	**88.53** (56.91)
DAPS†	**0.201** (0.103)	14.914 (4.184)	0.321 (0.067)	111.13 (71.33)
PnP-DM	0.259 (0.075)	11.983 (2.269)	0.431 (0.073)	308.84 (26.34)
REDDiff	0.319 (0.102)	10.372 (2.650)	0.280 (0.108)	94.67 (41.33)

on 14 selected volumes with severe knee pathologies, which includes 171 test images in total. For each method, we report the Precision, Recall, and mAP50 metrics for detection, and PSNR, SSIM, and Data Misfit for reconstruction, as shown in Table E.8. We also provide the rankings based on mAP50 and PSNR. Overall, the two rankings are correlated, which means that better pixel-wise accuracy indeed leads to a more accurate diagnosis. However, there are a few algorithms for which

Table E.6: **Results on Navier-Stokes equation.** Relative ℓ_2 error of different algorithms on 2D Navier-Stokes inverse problem, reported over 10 test cases. $*$: one or two test cases are excluded from the results due to numerical instability.

Subsampling ratio	×2			×4			×8		
Measurement noise	σ = 0.0	σ = 1.0	σ = 2.0	σ = 0.0	σ = 1.0	σ = 2.0	σ = 0.0	σ = 1.0	σ = 2.0
Traditional									
EKI	0.577 (0.138)	0.609 (0.119)	0.673 (0.107)	0.579 (0.145)	0.669 (0.131)	0.805 (0.112)	0.852 (0.167)	0.940(0.115)	1.116(0.090)
PnPDP									
DPS-fGSG	1.687 (0.156)	1.612 (0.173)	1.454 (0.154)	1.203* (0.122)	1.209* (0.116)	1.200* (0.100)	1.246* (0.108)	1.221* (0.082)	1.260 (0.117)
DPS-cGSG	2.203* (0.314)	2.117 (0.295)	1.746 (0.191)	1.175* (0.079)	1.133* (0.095)	1.114* (0.144)	1.186* (0.117)	1.204* (0.115)	1.218 (0.113)
DPG	0.325 (0.188)	0.408* (0.173)	0.466 (0.171)	0.322 (0.200)	0.361 (0.187)	**0.454** (0.207)	0.596 (0.301)	0.591 (0.262)	0.846 (0.251)
SCG	0.908 (0.600)	0.928 (0.557)	0.966 (0.546)	0.869 (0.513)	0.926 (0.546)	0.929 (0.505)	1.260 (0.135)	1.284 (0.117)	1.347 (0.141)
EnKG	**0.120** (0.085)	**0.191** (0.057)	**0.294** (0.061)	**0.115** (0.064)	**0.271** (0.053)	0.522 (0.136)	**0.287** (0.273)	**0.546** (0.212)	**0.773** (0.170)

Table E.7: **Table of metrics we use to capture the computation complexity of each algorithm.**

Metric	**Description**
# $\text{Fwd}_{\text{total}}$	total forward model evaluations
# DM_{total}	total diffusion model evaluations
# $\text{Fwd Grad}_{\text{total}}$	total forward model gradient evaluations
# $\text{DM Grad}_{\text{total}}$	total diffusion model gradient evaluations
$\text{Cost}_{\text{total}}$	total runtime
# Fwd_{seq}	sequential forward model evaluations
# DM_{seq}	sequential diffusion model evaluations
# $\text{Fwd Grad}_{\text{seq}}$	sequential forward model gradient evaluations
# $\text{DM Grad}_{\text{seq}}$	sequential diffusion model gradient evaluations
Cost_{seq}	sequential runtime

Table E.8: **Diagnostic performance of compressed sensing MRI reconstructions.**

Methods	Precision	Recall	mAP50	mAP50 Ranking	PSNR	SSIM	Data Misfit	PSNR Ranking
Traditional								
Wavelet+ℓ_1	0.532	0.332	0.385	9	28.16 (1.724)	0.685 (0.064)	23.501 (10.475)	8
TV	0.447	0.251	0.263	11	28.31 (1.834)	0.662 (0.079)	24.182 (10.613)	7
End-to-End								
Residual U-Net	0.482	0.462	0.439	8	31.62 (1.635)	0.803 (0.050)	–	2
E2E-VarNet	**0.610**	0.514	**0.500**	**1**	**32.25** (1.901)	**0.805** (0.056)	–	**1**
PnPDP								
CSGM	0.501	0.528	0.454	6	27.34 (2.770)	0.673 (0.082)	23.483 (10.651)	9
ScoreMRI	0.412	0.554	0.470	5	26.86 (2.583)	0.547 (0.092)	25.677 (10.491)	10
RED-diff	0.478	0.468	0.448	7	31.56 (2.337)	0.764 (0.080)	**23.406** (10.571)	3
DiffPIR	0.536	0.484	0.496	3	28.41 (1.403)	0.632 (0.061)	26.376 (10.555)	6
DPS	0.346	0.380	0.362	10	26.49 (1.550)	0.540 (0.067)	27.603 (11.127)	11
DAPS	0.514	0.556	0.480	4	30.15 (1.429)	0.725 (0.053)	23.978 (10.630)	4
PnP-DM	0.527	**0.579**	**0.500**	**1**	29.85 (2.934)	0.730 (0.056)	24.324 (10.413)	5
Fully sampled	0.573	0.581	0.535	–	–	–	23.721 (10.824)	–

the two rankings disagree: Residual U-Net, Score MRI, and RED-diff. The best methods for pathology detection are E2E-VarNet and PnP-DM.

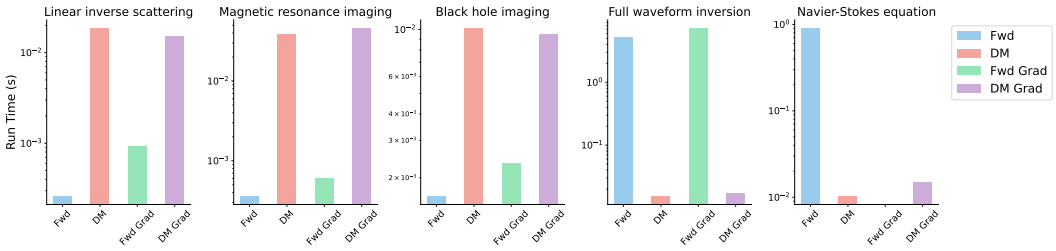


Figure E.1: **Computational characteristics of each forward model.** Fwd: runtime of a single forward model evaluation tested on a single A100 GPU. DM: runtime of a single diffusion model evaluation. Fwd Grad: runtime of a single forward model gradient evaluation. DM Grad: runtime of a single diffusion model gradient evaluation. Note that the inverse problem of the Navier-Stokes equation only permits black-box access to the forward model, so its Fwd Grad has no value.

E.2 Inverse Problem Details

E.2.1 Linear Inverse Scattering

Problem Details Consider a 2D object with permittivity distribution $\epsilon(\boldsymbol{r})$ in a bounded sample plane $\Omega \in \mathbb{R}^2$, which is immersed in the background medium with permittivity ϵ_b. The permittivity contrast is given by $\Delta\epsilon(\boldsymbol{r}) = \epsilon(\boldsymbol{r}) - \epsilon_b$. At each time, the object is illuminated by an incident light field $\boldsymbol{f}_{\text{in}}(\boldsymbol{r})$ emitted by one of $N > 0$ transmitters, and the scattered light field $\boldsymbol{f}_{\text{sc}}(\boldsymbol{r})$ is measured by $M > 0$ receivers. We adopt the experimental setup in [274] where the transmitters and receivers are arranged along a circle $\Gamma \in \mathbb{R}^2$ that surrounds the object. Here, $\boldsymbol{r} := (x, y)$ denotes the spatial coordinates. Under the first Born approximation [315], the interaction between the light and the object is governed by the following equation

$$\boldsymbol{f}_{\text{tot}}(\boldsymbol{r}) = \boldsymbol{f}_{\text{in}}(\boldsymbol{r}) + \int_{\Omega} g(\boldsymbol{r} - \boldsymbol{r}')\, s(\boldsymbol{r}')\, \boldsymbol{f}_{\text{in}}(\boldsymbol{r}')\, d\boldsymbol{r}', \quad \boldsymbol{r} \in \Omega, \tag{E.1}$$

where $\boldsymbol{f}_{\text{tot}}(\boldsymbol{r})$ is the total light field, and $s(\boldsymbol{r}) = \frac{1}{4\pi} k^2 \Delta\boldsymbol{\epsilon}(\boldsymbol{r})$ is the scattering potential. Here, $k = 2\pi/\lambda$ is the wavenumber in free space, and λ is the wavelength of the illumination. In the 2D space, the Green's function is given by

$$g(\boldsymbol{r} - \boldsymbol{r}') = \frac{i}{4} H_0^{(1)}(k_b \|\boldsymbol{r} - \boldsymbol{r}'\|_2) \tag{E.2}$$

where $k_b = \sqrt{\epsilon_b} k$ is the wavenumber of the background medium, and $H_0^{(1)}$ is the zero-order Hankel function of the first kind. Given the total field $\boldsymbol{f}_{\text{tot}}$ inside the sample domain Ω, the scattered field at the sensor plane Γ is given by

$$\boldsymbol{f}_{\text{sc}}(\boldsymbol{r}) = \int_{\Omega} g(\boldsymbol{r} - \boldsymbol{r}')\, s(\boldsymbol{r}')\, \boldsymbol{f}_{\text{tot}}(\boldsymbol{r}')\, d\boldsymbol{r}', \quad \boldsymbol{r} \in \Gamma. \tag{E.3}$$

Note that $\boldsymbol{r}$ denotes the sensor location in Γ, and the integral is computed over Ω.

By discretizing Equation (E.1) and Equation (E.3), we obtain a vectorized system that describes the linear inverse scattering problem. We denote the 2D vectorized permittivity distribution of the object by $\boldsymbol{x}_0 := s(\boldsymbol{r})$, and the corresponding measurement by $\boldsymbol{y}_{\text{sc}} = \boldsymbol{f}_{\text{sc}}(\boldsymbol{r})$ for notational consistency.

The forward model can thus be written as

$$\boldsymbol{y}_{\text{sc}} = \boldsymbol{H}(\boldsymbol{f}_{\text{tot}} \odot \boldsymbol{x}_0) + \boldsymbol{n} = \boldsymbol{A}\boldsymbol{x}_0 + \boldsymbol{n}, \tag{E.4}$$

where $\boldsymbol{f}_{\text{tot}} = \boldsymbol{G}(\boldsymbol{f}_{\text{in}} \odot \boldsymbol{x}_0)$, matrices $\boldsymbol{G}$ and $\boldsymbol{H}$ are discretizations of the Green's function at Γ and Ω, respectively, and $\boldsymbol{A} := \boldsymbol{H}\text{diag}(\boldsymbol{f}_{\text{tot}})$. We split and concatenate the real and imaginary parts of $\boldsymbol{A}$, and pre-compute the singular value decomposition of $\boldsymbol{A}$ to facilitate the plug-and-play diffusion methods that exploit SVD of linear inverse problems.

We set the physical size of test images to 18cm×18cm, and the wavelength of the illumination to $\lambda = 0.84$cm as specified in [272]. The forward model consists of $N = 20$ transmitters, placed uniformly on a circle of radius $R = 1.6$m. We further assume the background medium to be air with permittivity $\epsilon_b = 1$. We specify the number of receivers to be $M = 360, 180, 60$ in our experiments.

Related Work Linear inverse scattering aims to reconstruct the spatial distribution of an object's dielectric permittivity from the measurements of the light it scatters [148, 315]. This problem arises in various applications, such as ultrasound imaging [37], optical microscopy [62, 275], and digital holography [35]. Due to the physical constraints on the number and placement of sensors, the problem is often ill-posed, as the scattered light field is undersampled. Linear inverse scattering is commonly formulated as a linear inverse problem using scattering models based on the first Born [315] or Rytov [84] approximations. These models enable efficient computation and facilitate the use of convex optimization algorithms. On the other hand, nonlinear approaches have been developed to image strongly scattering objects [56, 150, 188, 208, 282], although these methods generally have a higher computational complexity. Deep learning-based methods have also been explored for linear inverse scattering. A common approach is to train convolutional neural networks (CNNs) to directly invert the scattering process by learning an inverse mapping from the measurements to permittivity distribution [179, 183, 274, 322]. Recent research has extended these efforts to more advanced deep learning techniques, such as neural fields [43, 193] and deep image priors [359].

E.2.2 Compressed Sensing Multi-Coil MRI

Problem Details We refer readers to Section 2.2 for details on the compressed sensing multi-coil MRI problem. We use the raw multi-coil k-space data from the fastMRI knee dataset [338]. We then estimate the coil sensitivity maps of each slice using the ESPIRiT [285] method implemented in `SigPy`[1]. Since different volumes in the dataset have different shapes, we adopt the preprocessing procedure in [145], leading to images with a shape of 320×320. The ground truth image is given by calculating the magnitude image of the Minimum Variance Unbiased Estimator (MVUE), which is used for all numbers reported in Table E.2 and Table E.3. The MVUE images are also used as ground truths for training the end-to-end deep learning methods Residual U-Net and E2E-VarNet [267].

Related Work Compressed sensing magnetic resonance imaging (CS-MRI) is a medical imaging technology that enables high-resolution visualization of human tissues with faster acquisition time than traditional MRI [204]. Instead of fully sampling the measurement space (a.k.a. k-space), CS-MRI only takes sparse measurements and then solves an inverse problem that recovers the underlying image [205]. The traditional approach is to solve a regularized optimization problem that involves a data-fit term and a regularization term, such as the total variation (TV) [29], and the ℓ_1-norm after a sparsifying transformation, such as the Wavelet transform [207] and dictionary decomposition [140, 243, 340]). End-to-end deep learning methods have also demonstrated strong performance in MRI reconstruction. Prior works have proposed unrolled networks [2, 123, 191, 255, 331], U-Net-based networks [142, 174], GAN-based networks [235, 329], among others [190, 202, 279, 305, 360]. These learning methods have achieved state-of-the-art performance on the fastMRI dataset [338]. Another line of work is to employ image denoisers as plug-and-play prior [145, 190, 273] Recently, diffusion model-based methods have been designed for CS-MRI reconstruction [64, 65, 201].

E.2.3 Black Hole Imaging

Related Work The Event Horizon Telescope (EHT) Collaboration aims to image black holes using a global network of radio telescopes operating at around a 1mm wavelength. Using traditional imaging techniques, the EHT Collaboration has successfully imaged the supermassive black holes M87* [71, 93] and SgrA* [92]. The classical imaging algorithm is CLEAN [67, 134], as implemented in the `DIFMAP`

[1] `https://github.com/mikgroup/sigpy` (BSD-3-Clause license)

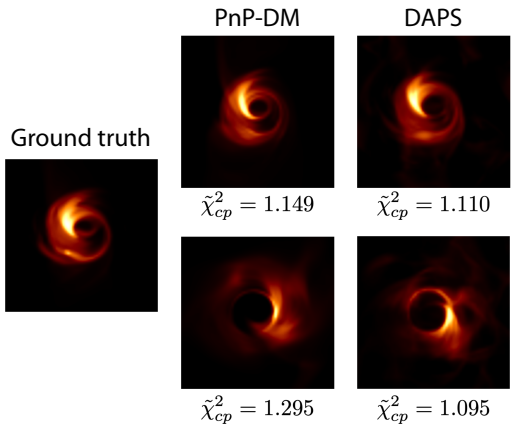


Figure E.2: **Multi-modal example on black hole imaging.** The image shows two image modes discovered by DAPS and PnP-DM.

[256, 257] software. `DIFMAP` is an inverse modeling approach that starts with the "dirty" image (given by the inverse Fourier transform of the visibilities) and iteratively deconvolves the image with an estimate point-spread function to "clean" the image. Since `DIFMAP` often requires a human-in-the-loop, we choose not to present results from `DIFMAP`. The EHT has also developed and used regularized maximum-likelihood approaches, namely `eht-imaging` [49–51] and `SMILI` [5–7]. Although they regularize and optimize the image differently [71], `eht-imaging` and `SMILI` both iteratively update an estimated image to agree with the measured data and regularization assumptions. Because of the simple regularization they choose to use, these baseline methods are limited in the amount of visual detail they can recover and do not recover detail far beyond the intrinsic resolution of the measurements. Some deep-learning-based regularization approaches have been proposed for VLBI [86, 101, 102], but most plug-and-play inverse diffusion solvers have not been validated on black hole imaging.

Multi-Modal Solutions As previously discussed, the non-convex and sparse measurement characteristics of black hole imaging may lead to multi-modal posterior distributions. While the solutions potentially look quite different from the ground truth, they may fit the measurements equally well and exhibit high prior likelihood. Figure E.2 illustrates two modes of solutions discovered by DAPS and PnP-DM. This multi-modal behavior has not been extensively formulated or discussed in previous literature, and we believe it represents a phenomenon worthy of further investigation.

E.2.4 Full Waveform Inversion

Problem Details We use an open-source software, `Devito` [199], for both the forward and adjoint modeling of FWI[2]. We discretize a physical domain of 2.54km $\times$ 1.27km with a 128×128 mesh, leading to a horizontal spacing of ≈20m and a vertical spacing of ≈10m. The time step is set to 0.001s which satisfies the Courant–Friedrichs–Lewy (CFL) condition [76]. We use a Ricker wavelet with a central frequency of 5Hz to excite the wavefield and model it for 1s. The natural boundary condition is set for the top boundary (free surface), which will generate reflected waves, while the absorbing boundary condition [68] is set for the rest boundaries to avoid artificial reflections. The absorbing boundary width is set to 80 grid points.

Inferring the subsurface velocity from observed data at receivers defines the inverse problem. We place 129 receivers evenly near the free surface (at a depth of 10m) to model a realistic scenario. We excite 16 sources evenly at a depth of 1270m. This source-receiver geometry is designed for the entire physical medium to be sampled by seismic waves, making it theoretically feasible to invert for v. `Devito` uses the adjoint-state method to estimate the gradient by cross-correlating the forward and adjoint wavefields at zero time lag [232]:

$$\nabla_{\boldsymbol{m}} \frac{1}{2} \left\| \boldsymbol{P}_r \boldsymbol{A}(\boldsymbol{m})^{-1} \boldsymbol{P}_s^T \boldsymbol{q}_s - \boldsymbol{y} \right\|_2^2 = \sum_{t=1}^{N_t} \boldsymbol{g}[t] \boldsymbol{h}_{tt}[t], \tag{E.5}$$

where $\boldsymbol{m} = \frac{1}{\boldsymbol{x}_0^2}$, N_t is the number of computational time steps, and $\boldsymbol{h}_{tt}$ is the second-order time derivative of the adjoint wavefield that solves

$$\boldsymbol{A}^T(\boldsymbol{m})\boldsymbol{h} = \boldsymbol{P}_r^T(\boldsymbol{P}_r \boldsymbol{g} - \boldsymbol{y}).$$

Recall that $\boldsymbol{g} = \boldsymbol{A}(\boldsymbol{m})^{-1} \boldsymbol{P}_s^T \boldsymbol{q}_s$ is the synthetic pressure wavefield.

E.2.5 Navier-Stokes Equation

To generate the training and test samples, we first draw independent identically distributed samples from the Gaussian random field $\mathcal{N}\left(\boldsymbol{0}, (-\Delta + 9\boldsymbol{I})^{-4}\right)$, where $-\Delta$ denotes the negative Laplacian. Then, we evolve them according to Equation (8.6) for 5 time units to get the final vorticity field, which generates an empirical distribution of the vorticity field with rich flow features. We set the forcing function as $f(\boldsymbol{x}) = -4\cos(4\boldsymbol{x}_2)$ where $\boldsymbol{x}_2$ indicates the second dimension of $\boldsymbol{x}$.

[2] See `https://www.devitoproject.org/examples/seismic/tutorials/03_fwi.html` for a tutorial.

E.3 Pre-Trained Diffusion Model Details

We train diffusion models following the pipeline from [151], using U-Net architectures from [85] and [266]. Detailed network configurations can be found in Table E.9.

Table E.9: **Model card for pre-trained diffusion models.**

	Inverse scattering	Black hole	MRI	FWI	2D Navier-Stokes
Input resolution	128×128	64×64	$2 \times 320 \times 320$	128×128	128×128
# Attention blocks in encoder/decoder	5	3	5	5	5
# Residual blocks per resolution	1	1	1	1	1
Attention resolutions	{16}	{16}	{16}	{16}	{16}
# Parameters	26.8M	20.0M	26.0M	26.8M	26.8M
# Training steps	50,000	50,000	100,000	50,000	50,000

E.3.1 Algorithms and Parameter Choices

E.3.1.1 Problem-Specific Baselines

Linear Inverse Scattering We include FISTA-TV [272] as a traditional optimization-based method. We set batch size $B = 20$ and $\tau = 5 \times 10^{-7}$ for all experiments.

Compressed Sensing Multi-Coil MRI We utilize both traditional methods, such as Wavelet+ℓ_1 [204, 205] and TV, as well as end-to-end models like Residual U-Net and E2E-VarNet [267]. For the traditional methods, we apply the same hyperparameter search strategy for fine-tuning, while the end-to-end models are trained using the Adam optimizer with a learning rate of 1×10^{-4} until convergence.

Black Hole Imaging We use `SMILI` [5–7] and `eht-imaging` [49–51] as our baseline methods. To ensure compatibility with the default hyperparameters of these methods, we preprocess the test dataset accordingly.

Full Waveform Inversion A classic baseline for full waveform inversion is LBFGS [187]. We set the maximum iteration to 5 and perform 100 global update steps with a Wolfe line search. The second baseline we consider is the Adam optimization algorithm [162]. We implement the Adam optimizer with a learning rate of 0.02 with the learning rate decay to minimize the data misfit term. For the traditional method, the initialization is a smoothed version of the ground truth, which is blurred using a Gaussian filter with $\sigma = 20$. We perform the inversion for 300 iterations.

Navier-Stokes Equation The traditional baseline we implement is the Ensemble Kalman Inversion (EKI) first proposed in [143]. It is implemented with 2,048

particles, 500 update steps, and an adaptive step size used in [166] to ensure a similar computation budget. Additional baselines include DPS-fGSG and DPS-cGSG, which are natural extensions based on DPS that replace gradient by zeroth-order gradient estimation first introduced in [356]. More specifically, we use the forward and central Gaussian Smoothed Gradient estimation technique [21].

E.3.1.2 Hyperparameter Selection

To ensure sufficient tuning of the hyperparameters for each algorithm, we employ a hybrid strategy combining grid search with Bayesian optimization and an early termination technique, using a small validation dataset. Specifically, we first perform a coarse grid search to narrow down the search space and then apply Bayesian optimization. For problems where the forward model is fast, such as linear inverse scattering, MRI, and black hole imaging, we conduct 50 to 100 iterations of Bayesian optimization to select the best hyperparameters. For computationally intensive problems such as full waveform inversion and Navier-Stokes equation, we use 10-30 iterations of Bayesian optimization combined with an early termination technique [180], based on data misfit. The details of the search spaces for Bayesian optimization and the optimized hyperparameter choices are listed in Table E.10.

Table E.10: **Hyperparameter search space and final choices of the diffusion-model-based algorithms on all five inverse problems.** Columns marked with task names present the chosen values for the reported main results in Appendix E.1. These values are selected by a hybrid hyperparameter search strategy described in Appendix E.3.1.2.

Methods / Parameters	Search space	Linear inverse scattering (360 / 180 / 60)	Black hole	MRI (Sim. / Raw)	FWI	2D Navier-Stokes
DPS						
Guidance scale	$[10^{-3}, 10^{3}]$	280/380/625	0.003	0.589/0.428	10^{-2}	–
LGD						
Guidance scale	$[10^{-3}, 10^{4}]$	3200/6400/13000	0.0082	–	11.73	–
# MC samples	[1, 20]	20	8	–	5	–
RED-diff						
Learning rate	$[10^{-4}, 1.0]$	0.04	0.05	4×10^{-2} / 2.96×10^{-2}	0.01	–
Regularization λ_{base}	$[10^{-3}, 1.0]$	0.0005	0.25	2.33×10^{-1} / 2.72×10^{-3}	0.1	–
Regularization schedule	constant, linear, sqrt	constant	constant	sqrt	linear	–
Gradient weight	$[10^{-2}, 10^{2}]$	1500	0.0004	6.68×10^{1} / 1.7×10^{-2}	1	–
DiffPIR						
# sampling steps	$\{200, 400, \ldots, 1000\}$	200	1000	1000	1000	–
Regularization λ	$[1, 10^{5}]$	$4 \times 10^{-4}/2 \times 10^{-4}/10^{-4}$	113.6	163 / 1.31	80.6	–
Stochasticity ζ	$[10^{-5}, 1]$	1	0.34	0.114 / 0.478	0.11	–
Noise level σ_y	$[10^{-2}, 10^{1}]$	0.01	1.4	1.05×10^{-2} / 1.36×10^{-1}	0.28	–
PnP-DM						
Annealing step	[50, 200]	100	100	100	150	–
Annealing sigma max	[10, 50]	10	10	10	25	–
Annealing decay rate	[0.60, 0.99]	0.9	0.93	0.93	0.99	–
Langevin step size	$[10^{-6}, 10^{-3}]$	$2 \times 10^{-5}/4 \times 10^{-5}/10^{-4}$	10^{-5}	10^{-6}	3×10^{-4}	–
Langevin step number	[10, 500]	200	200	200	10	–
Noise level	$[10^{-4}, 10^{1}]$	10^{-4}	1	1.02×10^{-3} / 1.15×10^{-2}	1	–
DAPS						
Annealing step	[50, 200]	200	100	200	150	–
Diffusion step	[1, 10]	10	5	5	5	–
Langevin step size	$[10^{-6}, 10^{-3}]$	$4 \times 10^{-5}/8 \times 10^{-5}/2 \times 10^{-4}$	10^{-4}	1.03×10^{-5} / 1.52×10^{-5}	3×10^{-4}	-
Langevin step number	[10, 500]	50	20	100	50	-
Noise level	$[10^{-4}, 10^{1}]$	10^{-4}	1	1.63×10^{-3} / 4.77×10^{-3}	1	–
Step size decay	[0.1, 1]	1/1/0.5	1	1	1	-
DDRM						
Stochasticity η	[0, 1]	0.85	–	–	–	–
DDNM						
Stochasticity η	[0, 1]	0.95	–	–	–	–
# time-travel steps L	[0, 5]	1	–	–	–	–
ΠGDM						
Stochasticity η	[0, 1]	0.2	–	–	–	–
FPS						
Stochasticity η	[0, 1]	0.9	–	–	–	–
# particles	[1, 20]	20	–	–	–	–
MCGdiff						
# particles	[1, 64]	16	–	–	–	–
DPS-fGSG						
Guidance scale	$[10^{-2}, 10^{2}]$	–	–	–	–	0.1
DPS-cGSG						
Guidance scale	$[10^{-2}, 10^{2}]$	–	–	–	–	0.1
DPG						
# MC samples	$\{1000, 2000, \ldots, 6000\}$	–	–	–	–	4000
Guidance scale	$[10^{-1}, 10^{3}]$	–	–	–	–	64
SCG						
# MC samples	$\{128, 256, 512\}$	–	–	–	–	512
EnKG						
Guidance scale	$\{1.0, 2.0, 4.0\}$	-	–	–	–	2.0
# particles	$\{512, 1024, 2048\}$	-	–	–	–	2048

BIBLIOGRAPHY

[1] Jonas Adler and Oktem Ozan. "Learned Primal-Dual Reconstruction." In: *IEEE Transactions on Medical Imaging* PP (July 2017). DOI: 10.1109/TMI.2018.2799231.

[2] Hemant K. Aggarwal, Merry P. Mani, and Mathews Jacob. "MoDL: Model-Based Deep Learning Architecture for Inverse Problems." In: *IEEE Transactions on Medical Imaging* 38.2 (2019), pp. 394–405. DOI: 10.1109/TMI.2018.2865356.

[3] Hemant Kumar Aggarwal and Mathews Jacob. "J-MoDL: Joint Model-Based Deep Learning for Optimized Sampling and Reconstruction." In: *IEEE Journal of Selected Topics in Signal Processing* 14.6 (2020), pp. 1151–1162. DOI: 10.1109/JSTSP.2020.3004094.

[4] Rizwan Ahmad, Charles Bouman, Gregery Buzzard, Stanley Chan, Sizhuo Liu, Edward Reehorst, and Philip Schniter. "Plug-and-Play Methods for Magnetic Resonance Imaging: Using Denoisers for Image Recovery." In: *IEEE Signal Processing Magazine* 37 (Jan. 2020), pp. 105–116. DOI: 10.1109/MSP.2019.2949470.

[5] Kazunori Akiyama, Shiro Ikeda, Mollie Pleau, Vincent L. Fish, Fumie Tazaki, Kazuki Kuramochi, Avery E. Broderick, Jason Dexter, Monika Mościbrodzka, Michael Gowanlock, Mareki Honma, and Sheperd S. Doeleman. "Superresolution Full-Polarimetric Imaging for Radio Interferometry with Sparse Modeling." In: *The Astronomical Journal* 153.4 (Mar. 2017), p. 159. DOI: 10.3847/1538-3881/aa6302. URL: https://dx.doi.org/10.3847/1538-3881/aa6302.

[6] Kazunori Akiyama, Kazuki Kuramochi, Shiro Ikeda, Vincent L. Fish, Fumie Tazaki, Mareki Honma, Sheperd S. Doeleman, Avery E. Broderick, Jason Dexter, Monika Mościbrodzka, Katherine L. Bouman, Andrew A. Chael, and Masamichi Zaizen. "Imaging the Schwarzschild-Radius-Scale Structure of M87 with the Event Horizon Telescope Using Sparse Modeling." In: *The Astrophysical Journal* 838.1 (Mar. 2017), p. 1. DOI: 10.3847/1538-4357/aa6305. URL: https://dx.doi.org/10.3847/1538-4357/aa6305.

[7] Kazunori Akiyama, Fumie Tazaki, Kotaro Moriyama, Ilje Cho, Shiro Ikeda, Mahito Sasada, Hiroki Okino, and Mareki Honma. *SMILI: Sparse Modeling Imaging Library for Interferometry*. Version v0.0.0. Mar. 2019. DOI: 10.5281/zenodo.2616725. URL: https://doi.org/10.5281/zenodo.2616725.

[8] Cagan Alkan, Morteza Mardani, Shreyas Vasanawala, and John M. Pauly. "Learning to Sample MRI via Variational Information Maximization." In:

NeurIPS 2020 Workshop on Deep Learning and Inverse Problems. 2020. URL: https://openreview.net/forum?id=1AOReNkDmh_.

[9] Brian D. O. Anderson. "Reverse-Time Diffusion Equation Models." In: *Stochastic Processes and their Applications* 12.3 (1982), pp. 313–326. ISSN: 0304-4149. DOI: 10.1016/0304-4149(82)90051-5. URL: https://www.sciencedirect.com/science/article/pii/0304414982900515.

[10] Liana Apostolova and Paul Thompson. "Brain Mapping as a Tool to Study Neurodegeneration." In: *Neurotherapeutics: The journal of the American Society for Experimental NeuroTherapeutics* 4 (Aug. 2007), pp. 387–400. DOI: 10.1016/j.nurt.2007.05.009.

[11] Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. "Structured Denoising Diffusion Models in Discrete State-Spaces." In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan. 2021. URL: https://openreview.net/forum?id=h7-XixPCAL.

[12] Cagla D. Bahadir, Alan Q. Wang, Adrian V. Dalca, and Mert R. Sabuncu. "Deep-Learning-Based Optimization of the Under-Sampling Pattern in MRI." In: *IEEE Transactions on Computational Imaging* 6 (2020), pp. 1139–1152. DOI: 10.1109/TCI.2020.3006727.

[13] Tim Bakker, Herke van Hoof, and Max Welling. "Experimental Design for MRI by Greedy Policy Search." In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. NIPS '20. Vancouver, BC, Canada: Curran Associates Inc., 2020. ISBN: 9781713829546. URL: https://dl.acm.org/doi/abs/10.5555/3495724.3497315.

[14] Guha Balakrishnan, Amy Zhao, Mert R. Sabuncu, John Guttag, and Adrian V. Dalca. "VoxelMorph: A Learning Framework for Deformable Medical Image Registration." In: *IEEE Transactions on Medical Imaging* 38.8 (2019), pp. 1788–1800. DOI: 10.1109/TMI.2019.2897538.

[15] Krishna Balasubramanian, Sinho Chewi, Murat A Erdogdu, Adil Salim, and Shunshi Zhang. "Towards a Theory of Non-Log-Concave Sampling:First-Order Stationarity Guarantees for Langevin Monte Carlo." In: *Proceedings of Thirty Fifth Conference on Learning Theory*. Ed. by Po-Ling Loh and Maxim Raginsky. Vol. 178. Proceedings of Machine Learning Research. PMLR, July 2022, pp. 2896–2923. URL: https://proceedings.mlr.press/v178/balasubramanian22a.html.

[16] Fan Bao, Chongxuan Li, Jiacheng Sun, and Jun Zhu. "Why Are Conditional Generative Models Better Than Unconditional Ones?" In: *NeurIPS 2022 Workshop on Score-Based Methods*. 2022. URL: https://openreview.net/forum?id=sbDyvrvvKn7.

[17] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. *Conditional Image Generation with Score-Based Diffusion Models*. 2021. arXiv: 2111.13606 [cs.LG]. URL: https://arxiv.org/abs/2111.13606.

[18] Jonathan Baxter and Peter L. Bartlett. "Infinite-Horizon Policy-Gradient Estimation." In: *Journal of Artificial Intelligence Research* 15.1 (Nov. 2001), pp. 319–350. ISSN: 1076-9757. DOI: 10.1613/jair.806.

[19] Amir Beck and Marc Teboulle. "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems." In: *SIAM Journal on Imaging Sciences* 2.1 (2009), pp. 183–202. DOI: 10.1137/080716542. eprint: https://doi.org/10.1137/080716542. URL: https://doi.org/10.1137/080716542.

[20] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. *Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation*. 2013. arXiv: 1308.3432 [cs.LG]. URL: https://arxiv.org/abs/1308.3432.

[21] Albert S. Berahas, Liyuan Cao, Krzysztof Choromanski, and Katya Scheinberg. "A Theoretical and Empirical Comparison of Gradient Approximations in Derivative-free Optimization." In: *Foundations of Computational Mathematics* 22.2 (2022), pp. 507–560. DOI: 10.1007/s10208-021-09513-z.

[22] Brady Bhalla*, Zachary Huang*, Elyes Serghine*, Bingliang Zhang, Zihui Wu, and Katherine L. Bouman. "Text-Guided Image Restoration via a Unified Plug-and-Play Diffusion Framework." In: *Computational Cameras and Displays (CCD) Workshop, CVPR 2025* (2025).

[23] Lindy Blackburn, Dominic W. Pesce, Michael D. Johnson, Maciek Wielgus, Andrew A. Chael, Pierre Christian, and Sheperd S. Doeleman. "Closure Statistics in Interferometric Data." In: *The Astrophysical Journal* 894.1 (May 2020), p. 31. DOI: 10.3847/1538-4357/ab8469. URL: https://dx.doi.org/10.3847/1538-4357/ab8469.

[24] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. *Stable Video Diffusion: Scaling Latent Video Diffusion Models to Large Datasets*. 2023. arXiv: 2311.15127 [cs.CV]. URL: https://arxiv.org/abs/2311.15127.

[25] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. "Align Your Latents: High-Resolution Video Synthesis with Latent Diffusion Models." In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 22563–22575. DOI: 10.1109/CVPR52729.2023.02161.

[26] Kai Tobias Block, Martin Uecker, and Jens Frahm. "Undersampled Radial MRI with Multiple Coils. Iterative Image Reconstruction Using a Total Variation Constraint." In: *Magnetic Resonance in Medicine* 57.6 (2007), pp. 1086–1098. DOI: 10.1002/mrm.21236. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/mrm.21236. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.21236.

[27] Sergey Bobkov and Prasad Tetali. "Modified Logarithmic Sobolev Inequalities in Discrete Settings." In: *Journal of Theoretical Probability* 19 (June 2006), pp. 289–336. DOI: 10.1007/s10959-006-0016-3.

[28] Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. "Diffusion Schrödinger Bridge with Applications to Score-Based Generative Modeling." In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan. 2021. URL: https://openreview.net/forum?id=9BnCwiXB0ty.

[29] Charles Bouman and Ken Sauer. "A Generalized Gaussian Image Model for Edge-Preserving MAP Estimation." In: *IEEE Transactions on Image Processing* 2.3 (1993), pp. 296–310. DOI: 10.1109/83.236536.

[30] Charles A. Bouman. *Foundations of Computational Imaging: A Model-Based Approach*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2022. DOI: 10.1137/1.9781611977134. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611977134. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611977134.

[31] Charles A. Bouman and Gregery T. Buzzard. "Generative Plug and Play: Posterior Sampling for Inverse Problems." In: *2023 59th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. 2023, pp. 1–7. DOI: 10.1109/Allerton58177.2023.10313413.

[32] Katherine L. Bouman, Michael D. Johnson, Daniel Zoran, Vincent L. Fish, Sheperd S. Doeleman, and William T. Freeman. "Computational Imaging for VLBI Image Reconstruction." In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 913–922. DOI: 10.1109/CVPR.2016.105.

[33] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers." In: *Foundations and Trends in Machine Learning* 3 (Jan. 2011), pp. 1–122. DOI: 10.1561/2200000016.

[34] Benjamin Boys, Mark Girolami, Jakiw Pidstrigach, Sebastian Reich, Alan Mosca, and Omer Deniz Akyildiz. "Tweedie Moment Projected Diffusions for Inverse Problems." In: *Transactions on Machine Learning Research* (2024). Featured Certification. ISSN: 2835-8856. URL: https://openreview.net/forum?id=4unJi0qrTE.

[35] David J. Brady, Kerkil Choi, Daniel L. Marks, Ryoichi Horisaki, and Sehoon Lim. "Compressive Holography." In: *Optics Express* 17.15 (July 2009), pp. 13040–13049. DOI: 10.1364/OE.17.013040. URL: https://opg.optica.org/oe/abstract.cfm?URI=oe-17-15-13040.

[36] Andrew Brock, Jeff Donahue, and Karen Simonyan. "Large Scale GAN Training for High Fidelity Natural Image Synthesis." In: *International Conference on Learning Representations*. 2019. URL: https://openreview.net/forum?id=B1xsqj09Fm.

[37] Michael M. Bronstein, Alexander M. Bronstein, Michael Zibulevsky, and Haim Azhari. "Reconstruction in Diffraction Ultrasound Tomography Using Nonuniform FFT." In: *IEEE Transactions on Medical Imaging* 21.11 (2002), pp. 1395–1401. DOI: 10.1109/TMI.2002.806423.

[38] Christopher P. Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. *Understanding disentangling in β-VAE*. 2018. arXiv: 1804.03599 [stat.ML]. URL: https://arxiv.org/abs/1804.03599.

[39] John C. Butcher. *Numerical Methods for Ordinary Differential Equations*. Third. Hoboken, New Jersey: John Wiley & Sons, 2016. ISBN: 9781119121503. DOI: 10.1002/9781119121534. URL: https://onlinelibrary.wiley.com/doi/book/10.1002/9781119121534.

[40] Andrew Campbell, Joe Benton, Valentin De Bortoli, Tom Rainforth, George Deligiannidis, and Arnaud Doucet. "A Continuous Time Framework for Discrete Denoising Models." In: *Advances in Neural Information Processing Systems*. Ed. by Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho. 2022. URL: https://openreview.net/forum?id=DmT862YAieY.

[41] Emmanuel J. Candes, Justin Romberg, and Terrence Tao. "Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information." In: *IEEE Transactions on Information Theory* 52.2 (2006), pp. 489–509. DOI: 10.1109/TIT.2005.862083.

[42] Emmanuel Candès, Xiaodong Li, and Mahdi Soltanolkotabi. "Phase Retrieval from Coded Diffraction Patterns." In: *Applied and Computational Harmonic Analysis* 39 (Oct. 2013). DOI: 10.1016/j.acha.2014.09.004.

[43] Ruiming Cao, Nikita Divekar, James Nuñez, Srigokul Upadhyayula, and Laura Waller. "Neural Space–Time Model for Dynamic Multi-Shot Imaging." In: *Nature Methods* 21 (Sept. 2024), pp. 2336–2341. DOI: 10.1038/s41592-024-02417-0.

[44] Gabriel Cardoso, Yazid Janati El Idrissi, Sylvain Le Corff, and Eric Moulines. "Monte Carlo Guided Denoising Diffusion Models for Bayesian Linear Inverse Problems." In: *The Twelfth International Conference on Learning*

Representations. 2024. URL: https://openreview.net/forum?id=nHESwXvxWK.

[45] Manuel Jorge Cardoso, Wenqi Li, Richard Brown, Nic Ma, Eric Kerfoot, Yiheng Wang, Benjamin Murrey, Andriy Myronenko, Can Zhao, Dong Yang, Vishwesh Nath, Yufan He, Ziyue Xu, Ali Hatamizadeh, Wentao Zhu, Yun Liu, Mingxin Zheng, Yucheng Tang, Isaac Yang, and Andrew Feng. "MONAI: An Open-Source Framework for Deep Learning in Healthcare." In: (Nov. 2022). DOI: 10.48550/arXiv.2211.02701.

[46] Jose A. Carrillo, Franca Hoffmann, Andrew M. Stuart, and Urbain Vaes. "Consensus Based Sampling." In: *Studies in Applied Mathematics* 148.3 (2022), pp. 1069–1140. DOI: 10.1111/sapm.12470. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/sapm.12470. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/sapm.12470.

[47] James Carton and Benjamin Giese. "A Reanalysis of Ocean Climate Using Simple Ocean Data Assimilation (SODA)." In: *Monthly Weather Review* 136 (Aug. 2008). DOI: 10.1175/2007MWR1978.1.

[48] Neil K. Chada, Andrew M. Stuart, and Xin T. Tong. "Tikhonov Regularization within Ensemble Kalman Inversion." In: *SIAM Journal on Numerical Analysis* 58.2 (2020), pp. 1263–1294. DOI: 10.1137/19M1242331. eprint: https://doi.org/10.1137/19M1242331. URL: https://doi.org/10.1137/19M1242331.

[49] Andrew Chael, Katie Bouman, Michael Johnson, Maciek Wielgus, Lindy Blackburn, Chi-kwan Chan, Joseph Rachid Farah, Daniel Palumbo, and Dominic Pesce. *eht-imaging: v1.1.0: Imaging Interferometric Data with Regularized Maximum Likelihood*. Version v1.1.0. Mar. 2019. DOI: 10.5281/zenodo.2614016. URL: 10.5281/zenodo.2614016.

[50] Andrew A. Chael, Michael D. Johnson, Katherine L. Bouman, Lindy L. Blackburn, Kazunori Akiyama, and Ramesh Narayan. "Interferometric Imaging Directly with Closure Phases and Closure Amplitudes." In: *The Astrophysical Journal* 857.1 (Apr. 2018), p. 23. DOI: 10.3847/1538-4357/aab6a8. URL: https://dx.doi.org/10.3847/1538-4357/aab6a8.

[51] Andrew A. Chael, Michael D. Johnson, Ramesh Narayan, Sheperd S. Doeleman, John F. C. Wardle, and Katherine L. Bouman. "High-Resolution Linear Polarimetric Imaging for the Event Horizon Telescope." In: *The Astrophysical Journal* 829.1 (Sept. 2016), p. 11. DOI: 10.3847/0004-637X/829/1/11. URL: https://dx.doi.org/10.3847/0004-637X/829/1/11.

[52] Stanley Chan, Xiran Wang, and Omar Elgendy. "Plug-and-Play ADMM for Image Restoration: Fixed Point Convergence and Applications." In: *IEEE Transactions on Computational Imaging* PP (May 2016). DOI: 10.1109/TCI.2016.2629286.

[53] Pascal Chang, Jingwei Tang, Markus Gross, and Vinicius C. Azevedo. "How I Warped Your Noise: a Temporally-Correlated Noise Prior for Diffusion Models." In: *The Twelfth International Conference on Learning Representations*. 2024. URL: https://openreview.net/forum?id=pzElnMrgSD.

[54] Nicolas Chauffert, Philippe Ciuciu, Jonas Kahn, and Pierre Weiss. "Variable Density Sampling with Continuous Trajectories." In: *SIAM Journal on Imaging Sciences* 7.4 (2014), pp. 1962–1992. DOI: 10.1137/130946642. eprint: https://doi.org/10.1137/130946642. URL: https://doi.org/10.1137/130946642.

[55] Liuhan Chen, Zongjian Li, Bin Lin, Bin Zhu, Qian Wang, Shenghai Yuan, Xing Zhou, Xinhua Cheng, and Li Yuan. *OD-VAE: An Omni-Dimensional Video Compressor for Improving Latent Video Diffusion Model*. 2024. arXiv: 2409.01199 [cs.CV]. URL: https://arxiv.org/abs/2409.01199.

[56] Michael Chen, David Ren, Hsiou-Yuan Liu, Shwetadwip Chowdhury, and Laura Waller. "Multi-Layer Born Multiple-Scattering Model for 3D phase Microscopy." In: *Optica* 7 (Apr. 2020), pp. 394–403. DOI: 10.1364/OPTICA.383030.

[57] Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru Zhang. "Sampling is as Easy as Learning the Score: Theory for Diffusion Models with Minimal Data Assumptions." In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=zyLVMgsZ0U_.

[58] Yongxin Chen, Sinho Chewi, Adil Salim, and Andre Wibisono. "Improved Analysis for a Proximal Algorithm for Sampling." In: *Proceedings of Thirty Fifth Conference on Learning Theory*. Ed. by Po-Ling Loh and Maxim Raginsky. Vol. 178. Proceedings of Machine Learning Research. PMLR, July 2022, pp. 2984–3014. URL: https://proceedings.mlr.press/v178/chen22c.html.

[59] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. "Diffusion Policy: Visuomotor Policy Learning via Action Diffusion." In: *Proceedings of Robotics: Science and Systems*. Daegu, Republic of Korea, July 2023. DOI: 10.15607/RSS.2023.XIX.026.

[60] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. "ILVR: Conditioning Method for Denoising Diffusion Probabilistic Models." In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 14347–14356. DOI: 10.1109/ICCV48922.2021.01410.

[61] Jooyoung Choi, Jungbeom Lee, Chaehun Shin, Sungwon Kim, Hyunwoo Kim, and Sungroh Yoon. "Perception Prioritized Training of Diffusion Models." In: *2022 IEEE/CVF Conference on Computer Vision and Pattern*

Recognition (CVPR). 2022, pp. 11462–11471. DOI: 10.1109/CVPR52688.2022.01118.

[62] Wonshik Choi, Christopher Fang-Yen, Kamran Badizadegan, Seungeun Oh, Niyom Lue, Ramachandra R. Dasari, and Michael S. Feld. "Tomographic Phase Microscopy." In: *Nature Methods* 4.9 (Sept. 2007), pp. 717–719. DOI: 10.1038/nmeth1078.

[63] Wenda Chu, Zihui Wu, Yifan Chen, Yang Song, and Yisong Yue. *Split Gibbs Discrete Diffusion Posterior Sampling*. 2025. arXiv: 2503.01161 [cs.LG]. URL: https://arxiv.org/abs/2503.01161.

[64] Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. "Diffusion Posterior Sampling for General Noisy Inverse Problems." In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=OnD9zGAGT0k.

[65] Hyungjin Chung and Jong Chul Ye. "Score-Based Diffusion Models for Accelerated MRI." In: *Medical Image Analysis* 80 (May 2022), p. 102479. DOI: 10.1016/j.media.2022.102479.

[66] Pieter Hendrik van Cittert. "Die Wahrscheinliche Schwingungsverteilung in Einer von Einer Lichtquelle Direkt Oder Mittels Einer Linse Beleuchteten Ebene." In: *Physica* 1.1 (1934), pp. 201–210. ISSN: 0031-8914. DOI: 10.1016/S0031-8914(34)90026-4. URL: https://www.sciencedirect.com/science/article/pii/S0031891434900264.

[67] Barry G. Clark. "An Efficient Implementation of the Algorithm "CLEAN"." In: *Astronomy and Astrophysics, vol. 89, no. 3, Sept. 1980, p. 377, 378.* 89 (1980), p. 377. URL: https://adsabs.harvard.edu/full/1980A%26A....89..377C.

[68] Robert Clayton and Björn Engquist. "Absorbing Boundary Conditions for Acoustic and Elastic Wave Equations." In: *Bulletin of the Seismological Society of America* 67.6 (Dec. 1977), pp. 1529–1540. ISSN: 0037-1106. DOI: 10.1785/BSSA0670061529. eprint: https://pubs.geoscienceworld.org/ssa/bssa/article-pdf/67/6/1529/5320934/bssa0670061529.pdf. URL: https://doi.org/10.1785/BSSA0670061529.

[69] Florentin Coeurdoux, Nicolas Dobigeon, and Pierre Chainais. "Plug-and-Play Split Gibbs Sampler: Embedding Deep Generative Priors in Bayesian Inference." In: *IEEE Transactions on Image Processing* 33 (2024), pp. 3496–3507. DOI: 10.1109/TIP.2024.3404338.

[70] Regev Cohen, Yochai Blau, Daniel Freedman, and Ehud Rivlin. "It Has Potential: Gradient-Driven Denoisers for Convergent Solutions to Inverse Problems." In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan. 2021. URL: https://openreview.net/forum?id=MYvpQVjCK0_.

[71] The Event Horizon Telescope Collaboration. "First M87 Event Horizon Telescope Results. IV. Imaging the Central Supermassive Black Hole." In: *The Astrophysical Journal Letters* 875.1 (Apr. 2019), p. L4. DOI: 10.3847/2041-8213/ab0e85. URL: https://dx.doi.org/10.3847/2041-8213/ab0e85.

[72] The Event Horizon Telescope Collaboration. "First M87 Event Horizon Telescope Results. V. Physical Origin of the Asymmetric Ring." In: *The Astrophysical Journal Letters* 875.1 (Apr. 2019), p. L5. DOI: 10.3847/2041-8213/ab0f43. URL: https://dx.doi.org/10.3847/2041-8213/ab0f43.

[73] Liam Connor, Katherine L Bouman, Vikram Ravi, and Gregg Hallinan. "Deep Radio-Interferometric Imaging with POLISH: DSA-2000 and Weak Lensing." In: *Monthly Notices of the Royal Astronomical Society* 514.2 (May 2022), pp. 2614–2626. ISSN: 0035-8711. DOI: 10.1093/mnras/stac1329. eprint: https://academic.oup.com/mnras/article-pdf/514/2/2614/44147595/stac1329.pdf. URL: https://doi.org/10.1093/mnras/stac1329.

[74] Robert L. Cook. "Stochastic Sampling in Computer Graphics." In: 5.1 (Jan. 1986), pp. 51–72. ISSN: 0730-0301. DOI: 10.1145/7529.8927. URL: https://doi.org/10.1145/7529.8927.

[75] Simon L. Cotter, Gareth O. Roberts, Andrew M. Stuart, and David White. "MCMC Methods for Functions: Modifying Old Algorithms to Make Them Faster." In: *Statistical Science* 28.3 (2013), pp. 424–446. DOI: 10.1214/13-STS421. URL: https://doi.org/10.1214/13-STS421.

[76] Richard Courant, Kurt Friedrichs, and Hans Lewy. "On the Partial Difference Equations of Mathematical Physics." In: *IBM Journal of Research and Development* 11.2 (1967), pp. 215–234. DOI: 10.1147/rd.112.0215.

[77] Giannis Daras, Weili Nie, Karsten Kreis, Alex Dimakis, Morteza Mardani, Nikola Borislavov Kovachki, and Arash Vahdat. "Warped Diffusion: Solving Video Inverse Problems with Image Diffusion Models." In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024. URL: https://openreview.net/forum?id=LH94zPv8cu.

[78] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. "Sequential Monte Carlo Samplers." In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 68.3 (May 2006), pp. 411–436. ISSN: 1369-7412. DOI: 10.1111/j.1467-9868.2006.00553.x. eprint: https://academic.oup.com/jrsssb/article-pdf/68/3/411/49795343/jrsssb_68_3_411.pdf. URL: https://doi.org/10.1111/j.1467-9868.2006.00553.x.

[79] Mauricio Delbracio and Peyman Milanfar. "Inversion by Direct Iteration: An Alternative to Denoising Diffusion for Image Restoration." In: *Transactions on Machine Learning Research* (2023). Featured Certification. ISSN: 2835-8856. URL: https://openreview.net/forum?id=VmyFF5lL3F.

[80] Chengyuan Deng, Shihang Feng, Hanchen Wang, Xitong Zhang, Peng Jin, Yinan Feng, Qili Zeng, Yinpeng Chen, and Youzuo Lin. "OpenFWI: Large-Scale Multi-structural Benchmark Datasets for Full Waveform Inversion." In: *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*. 2022. URL: https://openreview.net/forum?id=7w-a8PYPlP.

[81] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "ImageNet: A Large-Scale Hierarchical Image Database." In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[82] Yitong Deng, Winnie Lin, Lingxiao Li, Dmitriy Smirnov, Ryan D Burgert, Ning Yu, Vincent Dedun, and Mohammad H. Taghavi. "Infinite-Resolution Integral Noise Warping for Diffusion Models." In: *The Thirteenth International Conference on Learning Representations*. 2025. URL: https://openreview.net/forum?id=Y6LPWBo2HP.

[83] Arjun D. Desai, Andrew M. Schmidt, Elka B. Rubin, Christopher Michael Sandino, Marianne Susan Black, Valentina Mazzoli, Kathryn J. Stevens, Robert Boutin, Christopher Re, Garry E. Gold, Brian Hargreaves, and Akshay Chaudhari. "SKM-TEA: A Dataset for Accelerated MRI Reconstruction with Dense Image Labels for Quantitative Clinical Evaluation." In: *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*. 2021. URL: https://openreview.net/forum?id=YDMFgD_qJuA.

[84] Anthony J. Devaney. "Inverse-Scattering Theory within the Rytov Approximation." In: *Optics Letters* 6.8 (Aug. 1981), pp. 374–376. DOI: 10.1364/OL.6.000374.

[85] Prafulla Dhariwal and Alexander Quinn Nichol. "Diffusion Models Beat GANs on Image Synthesis." In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan. 2021. URL: https://openreview.net/forum?id=AAWuCvzaVt.

[86] Noe Dia, M. J. Yantovski-Barth, Alexandre Adam, Micah Bowles, Pablo Lemos, Anna M. M. Scaife, Yashar Hezaveh, and Laurence Perreault-Levasseur. *Bayesian Imaging for Radio Interferometry with Score-Based Priors*. 2023. arXiv: 2311.18012 [astro-ph.IM]. URL: https://arxiv.org/abs/2311.18012.

[87] Persi Diaconis and Laurent Saloff-Coste. "Logarithmic Sobolev Inequalities for Finite Markov Chains." In: *The Annals of Applied Probability* 6.3 (1996), pp. 695–750. DOI: 10.1214/aoap/1034968224. URL: https://doi.org/10.1214/aoap/1034968224.

[88] Lee R. Dice. "Measures of the Amount of Ecologic Association Between Species." In: *Ecology* 26.3 (1945), pp. 297–302. ISSN: 00129658, 19399170. URL: http://www.jstor.org/stable/1932409.

[89] Zehao Dou and Yang Song. "Diffusion Posterior Sampling for Linear Inverse Problem Solving: A Filtering Perspective." In: *The Twelfth International Conference on Learning Representations*. 2024. URL: https://openreview.net/forum?id=tplXNcHZs1.

[90] Bradley Efron. "Tweedie's Formula and Selection Bias." In: *Journal of the American Statistical Association* 106 (Dec. 2011), pp. 1602–1614. DOI: 10.1198/jasa.2011.tm11181.

[91] The Event Horizon Telescope Collaboration EHTC. "First M87 Event Horizon Telescope Results. III. Data Processing and Calibration." In: *The Astrophysical Journal Letters* 875.1 (Apr. 2019), p. L3. DOI: 10.3847/2041-8213/ab0c57. URL: https://dx.doi.org/10.3847/2041-8213/ab0c57.

[92] The Event Horizon Telescope Collaboration EHTC. "First Sagittarius A* Event Horizon Telescope Results. III. Imaging of the Galactic Center Supermassive Black Hole." In: *The Astrophysical Journal Letters* 930.2 (May 2022), p. L14. DOI: 10.3847/2041-8213/ac6429. URL: https://dx.doi.org/10.3847/2041-8213/ac6429.

[93] The Event Horizon Telescope Collaboration EHTC. "The Persistent Shadow of the Supermassive Black Hole of M 87-I. Observations, Calibration, Imaging, and Analysis." In: *Astronomy & Astrophysics* 681 (2024), A79. DOI: 10.1051/0004-6361/202347932.

[94] Gerrit Elsinga, Fulvio Scarano, Bernhard Wieneke, and Bas Oudheusden. "Tomographic Particle Image Velocimetry." In: *Experiments in Fluids* 41 (Dec. 2006), pp. 933–947. DOI: 10.1007/s00348-006-0212-z.

[95] Patrick Esser, Robin Rombach, and Björn Ommer. "Taming Transformers for High-Resolution Image Synthesis." In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 12868–12878. DOI: 10.1109/CVPR46437.2021.01268.

[96] Geir Evensen. "Sequential Data Assimilation with a Nonlinear Quasi-Geostrophic Model Using Monte Carlo Methods to Forecast Error Statistics." In: *Journal of Geophysical Research: Oceans* 99.C5 (1994), pp. 10143–10162. DOI: 10.1029/94JC00572. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/94JC00572. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/94JC00572.

[97] Jiaojiao Fan, Bo Yuan, and Yongxin Chen. "Improved Dimension Dependence of a Proximal Algorithm for Sampling." In: *Proceedings of Thirty Sixth Conference on Learning Theory*. Ed. by Gergely Neu and Lorenzo Rosasco. Vol. 195. Proceedings of Machine Learning Research. PMLR, July 2023, pp. 1473–1521. URL: https://proceedings.mlr.press/v195/fan23a.html.

[98] Zhiwen Fan, Liyan Sun, Xinghao Ding, Yue Huang, Congbo Cai, and John Paisley. "A Segmentation-Aware Deep Fusion Network for Compressed Sensing MRI." In: *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part VI*. Munich, Germany: Springer-Verlag, 2018, pp. 55–70. ISBN: 978-3-030-01230-4. DOI: 10.1007/978-3-030-01231-1_4. URL: https://doi.org/10.1007/978-3-030-01231-1_4.

[99] Zhenghan Fang, Sam Buchanan, and Jeremias Sulam. "What's in a Prior? Learned Proximal Networks for Inverse Problems." In: *The Twelfth International Conference on Learning Representations*. 2024. URL: https://openreview.net/forum?id=kNPcOaqC5r.

[100] Elhadji C. Faye, Mame Diarra Fall, and Nicolas Dobigeon. "Regularization by Denoising: Bayesian Model and Langevin-Within-Split Gibbs Sampling." In: *IEEE Transactions on Image Processing* 34 (2025), pp. 221–234. DOI: 10.1109/TIP.2024.3520012.

[101] Berthy Feng and Katherine Bouman. "Variational Bayesian Imaging with an Efficient Surrogate Score-Based Prior." In: *Transactions on Machine Learning Research* (2024). ISSN: 2835-8856. URL: https://openreview.net/forum?id=db2pFKVcm1.

[102] Berthy Feng, Katherine Bouman, and William Freeman. "Event-Horizon-Scale Imaging of M87* under Different Assumptions via Deep Generative Image Priors." In: *The Astrophysical Journal* 975 (Nov. 2024), p. 201. DOI: 10.3847/1538-4357/ad737f.

[103] Berthy T. Feng, Jamie Smith, Michael Rubinstein, Huiwen Chang, Katherine L. Bouman, and William T. Freeman. "Score-Based Diffusion Models as Principled Priors for Inverse Imaging." In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 10486–10497. DOI: 10.1109/ICCV51070.2023.00965.

[104] James R. Fienup. "Phase Retrieval Algorithms: A Comparison." In: *Applied Optics* 21 15 (1982), pp. 2758–69. DOI: 10.1364/AO.21.002758.

[105] Bruce Fischl. "FreeSurfer." In: *NeuroImage* 62 (Jan. 2012), pp. 774–81. DOI: 10.1016/j.neuroimage.2012.01.021.

[106] Vincent Fortin, Mabrouk Abaza, F. Anctil, and Richard Turcotte. "Why Should Ensemble Spread Match the RMSE of the Ensemble Mean? (vol 15,

pg 1708, 2014).” In: *Journal of Hydrometeorology* 16 (Feb. 2015), pp. 484–484. DOI: 10.1175/JHM-D-14-0161.1.

[107] William T. Freeman, Thouis R. Jones, and Egon C. Pasztor. “Example-Based Super-Resolution.” In: *IEEE Computer Graphics and Applications* 22.2 (2002), pp. 56–65. DOI: 10.1109/38.988747.

[108] Cong Fu, Keqiang Yan, Limei Wang, Wing Yee Au, Michael Curtis McThrow, Tao Komikado, Koji Maruhashi, Kanji Uchino, Xiaoning Qian, and Shuiwang Ji. “A Latent Diffusion Model for Protein Structure Generation.” In: *Proceedings of the Second Learning on Graphs Conference*. Ed. by Soledad Villar and Benjamin Chamberlain. Vol. 231. Proceedings of Machine Learning Research. PMLR, Nov. 2024, 29:1–29:17. URL: https://proceedings.mlr.press/v231/fu24a.html.

[109] Daniel Gabay and Bertrand Mercier. “A Dual Algorithm for the Solution of Nonlinear Variational Problems via Finite Element Approximation.” In: *Computers & Mathematics with Applications* 2.1 (1976), pp. 17–40. ISSN: 0898-1221. DOI: 10.1016/0898-1221(76)90003-1. URL: https://www.sciencedirect.com/science/article/pii/0898122176900031.

[110] Urs Gamper, Peter Boesiger, and Sebastian Kozerke. “Compressed Sensing in Dynamic MRI.” In: *Magnetic Resonance in Medicine* 59.2 (2008), pp. 365–373. DOI: 10.1002/mrm.21477.

[111] Alfredo Garbuno-Inigo, Franca Hoffmann, Wuchen Li, and Andrew M. Stuart. “Interacting Langevin Diffusions: Gradient Structure and Ensemble Kalman Sampler.” In: *SIAM Journal on Applied Dynamical Systems* 19.1 (2020), pp. 412–441. DOI: 10.1137/19M1251655. eprint: https://doi.org/10.1137/19M1251655. URL: https://doi.org/10.1137/19M1251655.

[112] Alfredo Garbuno-Inigo, Nikolas Nüsken, and Sebastian Reich. “Affine Invariant Interacting Langevin Dynamics for Bayesian Inference.” In: *SIAM Journal on Applied Dynamical Systems* 19.3 (2020), pp. 1633–1658. DOI: 10.1137/19M1304891. eprint: https://doi.org/10.1137/19M1304891. URL: https://doi.org/10.1137/19M1304891.

[113] Andrew Gelman, Walter R. Gilks, and Gareth O. Roberts. “Weak Convergence and Optimal Scaling of Random Walk Metropolis Algorithms.” In: *The Annals of Applied Probability* 7.1 (1997), pp. 110–120. DOI: 10.1214/aoap/1034625254. URL: https://doi.org/10.1214/aoap/1034625254.

[114] D. Geman and Chengda Yang. “Nonlinear Image Recovery with Half-quadratic Regularization.” In: *IEEE Transactions on Image Processing* 4.7 (1995), pp. 932–946. DOI: 10.1109/83.392335.

[115] Charles J. Geyer. "Practical Markov Chain Monte Carlo." In: *Statistical Science* 7.4 (1992), pp. 473–483. DOI: 10.1214/ss/1177011137. URL: https://doi.org/10.1214/ss/1177011137.

[116] Vahid Ghodrati, Jiaxin Shao, Mark Bydder, Ziwu Zhou, Wotao Yin, Kim-Lien Nguyen, Yingli Yang, and Peng Hu. "MR Image Reconstruction Using Deep Learning: Evaluation of Network Structure and Loss Functions." In: *Quantitative Imaging in Medicine and Surgery* 9 (Sept. 2019), pp. 1516–1527. DOI: 10.21037/qims.2019.08.10.

[117] Davis Gilton, Gregory Ongie, and Rebecca Willett. "Deep Equilibrium Architectures for Inverse Problems in Imaging." In: *IEEE Transactions on Computational Imaging* 7 (2021), pp. 1123–1133. DOI: 10.1109/TCI.2021.3118944.

[118] Tilmann Gneiting and Adrian Raftery. "Strictly Proper Scoring Rules, Prediction, and Estimation." In: *Journal of the American Statistical Association* 102 (Mar. 2007), pp. 359–378. DOI: 10.1198/016214506000001437.

[119] Alexandros Graikos, Nikolay Malkin, Nebojsa Jojic, and Dimitris Samaras. "Diffusion Models as Plug-and-Play Priors." In: *Advances in Neural Information Processing Systems*. Ed. by Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho. 2022. URL: https://openreview.net/forum?id=yhlMZ3iR7Pu.

[120] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. "Vector Quantized Diffusion Model for Text-to-Image Synthesis." In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 10686–10696. DOI: 10.1109/CVPR52688.2022.01043.

[121] Justin P. Haldar, Diego Hernando, and Zhi-Pei Liang. "Compressed-Sensing MRI With Random Encoding." In: *IEEE Transactions on Medical Imaging* 30.4 (2011), pp. 893–903. DOI: 10.1109/TMI.2010.2085084.

[122] Gregg Hallinan et al. "The DSA-2000 — A Radio Survey Camera." In: *Bulletin of the AAS* 51.7 (Sept. 2019). https://baas.aas.org/pub/2020n7i255. URL: https://arxiv.org/abs/1907.07648.

[123] Kerstin Hammernik, Teresa Klatzer, Erich Kobler, Michael P. Recht, Daniel K. Sodickson, Thomas Pock, and Florian Knoll. "Learning a Variational Network for Reconstruction of Accelerated MRI Data." In: *Magnetic Resonance in Medicine* 79.6 (2018), pp. 3055–3071. DOI: 10.1002/mrm.26977. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/mrm.26977. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.26977.

[124] Hado van Hasselt, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In: *Proceedings of the Thirtieth AAAI*

Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA. AAAI Press, 2016, pp. 2094–2100. URL: https://aaai.org/papers/10295-deep-reinforcement-learning-with-double-q-learning/.

[125] Wilfred K. Hastings. "Monte Carlo Sampling Methods Using Markov Chains and Their Applications." In: *Biometrika* 57.1 (Apr. 1970), pp. 97–109. ISSN: 0006-3444. DOI: 10.1093/biomet/57.1.97. eprint: https://academic.oup.com/biomet/article-pdf/57/1/97/23940249/57-1-97.pdf. URL: https://doi.org/10.1093/biomet/57.1.97.

[126] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep Residual Learning for Image Recognition." In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[127] Yinnian He and Weiwei Sun. "Stability and Convergence of the Crank–Nicolson/Adams–Bashforth Scheme for the Time-Dependent Navier–Stokes Equations." In: *SIAM Journal on Numerical Analysis* 45.2 (2007), pp. 837–869. DOI: 10.1137/050639910. eprint: https://doi.org/10.1137/050639910. URL: https://doi.org/10.1137/050639910.

[128] Tobias Heimann, Bryan J. Morrison, Martin A. Styner, Marc Niethammer, and S. Warfield. "Segmentation of Knee Images: A Grand Challenge." In: *Proc. MICCAI Workshop on Medical Image Analysis for the Clinic*. Beijing, China. 2010, pp. 207–214.

[129] Bastian Hilder, Mark Peletier, Mikola Schlottke, Guido Schneider, and Upanshu Sharma. "An FIR Inequality for Markov Jump Processes on Discrete State Spaces." In: (2017). URL: https://research.tue.nl/en/studentTheses/an-fir-inequality-for-markov-jump-processes-on-discrete-state-spa.

[130] Bastian Hilder, Mark A. Peletier, Upanshu Sharma, and Oliver Tse. "An Inequality Connecting Entropy Distance, Fisher Information and Large Deviations." In: *Stochastic Processes and their Applications* 130.5 (2020), pp. 2596–2638. ISSN: 0304-4149. DOI: 10.1016/j.spa.2019.07.012. URL: https://www.sciencedirect.com/science/article/pii/S0304414919300687.

[131] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising Diffusion Probabilistic Models." In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., 2020, pp. 6840–6851. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

[132] Jonathan Ho and Tim Salimans. "Classifier-Free Diffusion Guidance." In: *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*. 2021. URL: https://openreview.net/forum?id=qw8AKxfYbI.

[133] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. "Video Diffusion Models." In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh. Vol. 35. Curran Associates, Inc., 2022, pp. 8633–8646. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/39235c56aef13fb05a6adc95eb9d8d66-Paper-Conference.pdf.

[134] Jan A. Högbom and Tim J. Cornwell. "Aperture Synthesis with a Non-Regular Distribution of Interferometer Baselines." In: *Astronomy and Astrophysics* 500 (1974), pp. 55–66. URL: https://api.semanticscholar.org/CorpusID:116934453.

[135] Andrew Hoopes, Malte Hoffmann, Bruce Fischl, John Guttag, and Adrian V. Dalca. "HyperMorph: Amortized Hyperparameter Learning for Image Registration." In: *Information Processing in Medical Imaging: 27th International Conference, IPMI 2021, Virtual Event, June 28–June 30, 2021, Proceedings*. Berlin, Heidelberg: Springer-Verlag, 2021, pp. 3–17. ISBN: 978-3-030-78190-3. DOI: 10.1007/978-3-030-78191-0_1. URL: https://doi.org/10.1007/978-3-030-78191-0_1.

[136] Seyed Amir Hossein Hosseini, Burhaneddin Yaman, Steen Moeller, Mingyi Hong, and Mehmet Akçakaya. "Dense Recurrent Neural Networks for Accelerated MRI: History-Cognizant Unrolling of Optimization Algorithms." In: *IEEE Journal of Selected Topics in Signal Processing* PP (June 2020), pp. 1–1. DOI: 10.1109/JSTSP.2020.3003170.

[137] Pieter Houtekamer and Fuqing Zhang. "Review of the Ensemble Kalman Filter for Atmospheric Data Assimilation." In: *Monthly Weather Review* 144 (June 2016). DOI: 10.1175/MWR-D-15-0440.1.

[138] Daniel Huang, Jiaoyang Huang, Sebastian Reich, and Andrew Stuart. "Efficient derivative-free Bayesian inference for large-scale inverse problems." In: *Inverse Problems* 38 (Oct. 2022). DOI: 10.1088/1361-6420/ac99fa.

[139] Daniel Zhengyu Huang, Tapio Schneider, and Andrew M. Stuart. "Iterated Kalman Methodology for Inverse Problems." In: *Journal of Computational Physics* 463 (2022), p. 111262. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2022.111262. URL: https://www.sciencedirect.com/science/article/pii/S0021999122003242.

[140] Yue Huang, John Paisley, Qin Lin, Xinghao Ding, Xueyang Fu, and Xiao-Ping Zhang. "Bayesian Nonparametric Dictionary Learning for Compressed

Sensing MRI." In: *IEEE Transactions on Image Processing* 23.12 (2014), pp. 5007–5019. DOI: 10.1109/TIP.2014.2360122.

[141] Yujia Huang, Adishree Ghatare, Yuanzhe Liu, Ziniu Hu, Qinsheng Zhang, Chandramouli Shama Sastry, Siddharth Gururani, Sageev Oore, and Yisong Yue. "Symbolic Music Generation with Non-Differentiable Rule Guided Diffusion." In: *Proceedings of the 41st International Conference on Machine Learning*. Ed. by Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp. Vol. 235. Proceedings of Machine Learning Research. PMLR, July 2024, pp. 19772–19797. URL: https://proceedings.mlr.press/v235/huang24g.html.

[142] Chang Min Hyun, Hwa Pyung Kim, Sung Min Lee, Sungchul Lee, and Jin Keun Seo. "Deep Learning for Undersampled MRI Reconstruction." In: *Physics in Medicine & Biology* 63.13 (June 2018), p. 135007. DOI: 10.1088/1361-6560/aac71a. URL: https://dx.doi.org/10.1088/1361-6560/aac71a.

[143] Marco Iglesias, Kody Law, and Andrew Stuart. "Ensemble Kalman Methods for Inverse Problems." In: *Inverse Problems* 29 (Mar. 2013), p. 045001. DOI: 10.1088/0266-5611/29/4/045001.

[144] Marco A Iglesias. "A Regularizing Iterative Ensemble Kalman Method for PDE-constrained Inverse Problems." In: *Inverse Problems* 32.2 (2016), p. 025002. DOI: 10.1088/0266-5611/32/2/025002.

[145] Ajil Jalal, Marius Arvinte, Giannis Daras, Eric Price, Alex Dimakis, and Jonathan Tamir. "Robust Compressed Sensing MRI with Deep Generative Priors." In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan. 2021. URL: https://openreview.net/forum?id=wHoIjrT6MMb.

[146] Kyong Hwan Jin, Michael Unser, and Kwang Moo Yi. "Self-Supervised Deep Active Accelerated MRI." In: *arXiv preprint arXiv:1901.04547* (2019).

[147] Zahra Kadkhodaie and Eero P. Simoncelli. "Stochastic Solutions for Linear Inverse Problems Using the Prior Implicit in a Denoiser." In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan. 2021. URL: https://openreview.net/forum?id=x5hh6N9bUUb.

[148] Avinash C. Kak and Malcolm Slaney. *Principles of Computerized Tomographic Imaging*. Society for Industrial and Applied Mathematics, 2001. DOI: 10.1137/1.9780898719277. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9780898719277. URL: https://epubs.siam.org/doi/abs/10.1137/1.9780898719277.

[149] Ulugbek Kamilov, Charles Bouman, Gregery Buzzard, and Brendt Wohlberg. "Plug-and-Play Methods for Integrating Physical and Learned Models in Computational Imaging: Theory, Algorithms, and Applications." In: *IEEE Signal Processing Magazine* 40 (Jan. 2023), pp. 85–97. DOI: 10.1109/MSP.2022.3199595.

[150] Ulugbek Kamilov, Ioannis Papadopoulos, Morteza Shoreh, Alexandre Goy, Cedric Vonesch, Michael Unser, and Demetri Psaltis. "Learning Approach to Optical Tomography." In: *Optica* 2 (May 2015), pp. 517–522. DOI: 10.1364/OPTICA.2.000517.

[151] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. "Elucidating the Design Space of Diffusion-Based Generative Models." In: *Advances in Neural Information Processing Systems*. Ed. by Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho. 2022. URL: https://openreview.net/forum?id=k7FuTOWMOc7.

[152] Tero Karras, Samuli Laine, and Timo Aila. "A Style-Based Generator Architecture for Generative Adversarial Networks." In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 4396–4405. URL: https://api.semanticscholar.org/CorpusID:54482423.

[153] Sergey Kastryulin, Jamil Zakirov, Denis Prokopenko, and Dmitry V. Dylov. *PyTorch Image Quality: Metrics for Image Quality Assessment*. 2022. DOI: 10.48550/ARXIV.2208.14818. URL: https://arxiv.org/abs/2208.14818.

[154] Rosemary Kates, David Atkinson, and Michael N. Brant-Zawadzki. "Fluid-Attenuated Inversion Recovery (FLAIR): Clinical Prospectus of Current and Future Applications." In: *Topics in Magnetic Resonance Imaging: TMRI* 86 (1996), pp. 389–396. URL: https://pubmed.ncbi.nlm.nih.gov/9402679/.

[155] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. "Denoising Diffusion Restoration Models." In: *Advances in Neural Information Processing Systems*. Ed. by Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho. 2022. URL: https://openreview.net/forum?id=kxXvopt9pWK.

[156] Bahjat Kawar, Gregory Vaksman, and Michael Elad. "SNIPS: Solving Noisy Inverse Problems Stochastically." In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan. 2021. URL: https://openreview.net/forum?id=pBKOx_dxYAN.

[157] Michael Kellman, Emrah Bostan, Michael Chen, and Laura Waller. "Data-Driven Design for Fourier Ptychographic Microscopy." In: *2019 IEEE Inter-*

national Conference on Computational Photography (ICCP). 2019, pp. 1–8. DOI: 10.1109/ICCPHOT.2019.8747339.

[158] Michael R. Kellman, Emrah Bostan, Nicole A. Repina, and Laura Waller. "Physics-Based Learned Design: Optimized Coded-Illumination for Quantitative Phase Imaging." In: *IEEE Transactions on Computational Imaging* 5.3 (2019), pp. 344–353. DOI: 10.1109/TCI.2019.2905434.

[159] Rahima Khanam and Muhammad Hussain. *YOLOv11: An Overview of the Key Architectural Enhancements*. 2024. arXiv: 2410.17725 [cs.CV]. URL: https://arxiv.org/abs/2410.17725.

[160] Jeongsol Kim, Geon Yeong Park, Hyungjin Chung, and Jong Chul Ye. "Regularization by Texts for Latent Diffusion Inverse Solvers." In: *The Thirteenth International Conference on Learning Representations*. 2025. URL: https://openreview.net/forum?id=TtUh0TOlGX.

[161] Won Jun Kim, Hyungjin Chung, Jaemin Kim, Sangmin Lee, Byeongsu Sim, and Jong Chul Ye. "Derivative-Free Diffusion Manifold-Constrained Gradient for Unified XAI." In: *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*. June 2025, pp. 23795–23805. URL: https://openaccess.thecvf.com/content/CVPR2025/html/Kim_Derivative-Free_Diffusion_Manifold-Constrained_Gradient_for_Unified_XAI_CVPR_2025_paper.html.

[162] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization." In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://arxiv.org/abs/1412.6980.

[163] Diederik P. Kingma and Max Welling. "Auto-Encoding Variational Bayes." In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2014. URL: http://arxiv.org/abs/1312.6114.

[164] Florian Knoll, Tullie Murrell, Anuroop Sriram, Nafissa Yakubova, Jure Zbontar, Michael Rabbat, Aaron Defazio, Matthew J. Muckley, Daniel K. Sodickson, C. Lawrence Zitnick, and Michael P. Recht. "Advancing Machine Learning for MR Image Reconstruction with an Open Competition: Overview of the 2019 fastMRI Challenge." In: *Magnetic Resonance in Medicine* 84.6 (2020), pp. 3054–3070. DOI: 10.1002/mrm.28338. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/mrm.28338. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.28338.

[165] Florian Knoll et al. “fastMRI: A Publicly Available Raw K-space and DICOM Dataset of Knee Images for Accelerated MR Image Reconstruction Using Machine Learning.” In: *Radiology: Artificial Intelligence* 2.1 (2020). PMID: 32076662, e190007. DOI: 10.1148/ryai.2020190007. eprint: https://doi.org/10.1148/ryai.2020190007. URL: https://doi.org/10.1148/ryai.2020190007.

[166] Nikola Kovachki and Andrew Stuart. “Ensemble Kalman Inversion: A Derivative-Free Technique for Machine Learning Tasks.” In: *Inverse Problems* 35 (Aug. 2019). DOI: 10.1088/1361-6420/ab1c3a.

[167] Taesung Kwon, Gookho Song, Yoosun Kim, Jong Chul Ye, and Mooseok Jang. *Seeing Video Through Optical Scattering Media Using Spatio-Temporal Diffusion Models*. 2024. URL: https://openreview.net/forum?id=DHCp41nv1M.

[168] Taesung Kwon and Jong Chul Ye. “Solving Video Inverse Problems Using Image Diffusion Models.” In: *The Thirteenth International Conference on Learning Representations*. 2025. URL: https://openreview.net/forum?id=TRWxFUzK9K.

[169] Taesung Kwon and Jong Chul Ye. *VISION-XL: High Definition Video Inverse Problem Solver Using Latent Image Diffusion Models*. 2024. arXiv: 2412.00156 [cs.CV]. URL: https://arxiv.org/abs/2412.00156.

[170] Rémi Laumont, Valentin De Bortoli, Andrés Almansa, Julie Delon, Alain Durmus, and Marcelo Pereyra. “Bayesian Imaging Using Plug & Play Priors: When Langevin Meets Tweedie.” In: *SIAM Journal on Imaging Sciences* 15.2 (2022), pp. 701–737. DOI: 10.1137/21M1406349. eprint: https://doi.org/10.1137/21M1406349. URL: https://doi.org/10.1137/21M1406349.

[171] Paul C. Lauterbur. “Image Formation by Induced Local Interactions: Examples Employing Nuclear Magnetic Resonance.” In: *Nature* 242 (1973), pp. 190–191. DOI: 10.1038/242190a0.

[172] Yann Lecun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. “Gradient-Based Learning Applied to Document Recognition.” In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324. DOI: 10.1109/5.726791.

[173] Dongwook Lee, Jaejun Yoo, Sungho Tak, and Jong Chul Ye. “Deep Residual Learning for Accelerated MRI Using Magnitude and Phase Networks.” In: *IEEE Transactions on Biomedical Engineering* 65.9 (2018), pp. 1985–1995. DOI: 10.1109/TBME.2018.2821699.

[174] Dongwook Lee, Jaejun Yoo, and Jong Chul Ye. “Deep Residual Learning for Compressed Sensing MRI.” In: *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. 2017, pp. 15–18. DOI: 10.1109/ISBI.2017.7950457.

[175] Holden Lee, Jianfeng Lu, and Yixin Tan. "Convergence for Score-Based Generative Modeling with Polynomial Complexity." In: *Advances in Neural Information Processing Systems*. Ed. by Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho. 2022. URL: https://openreview.net/forum?id=dUSI4vFyMK.

[176] Yin Tat Lee, Ruoqi Shen, and Kevin Tian. "Structured Logconcave Sampling with a Restricted Gaussian Oracle." In: *Proceedings of Thirty Fourth Conference on Learning Theory*. Vol. 134. Proceedings of Machine Learning Research. PMLR, Aug. 2021, pp. 2993–3050. URL: https://proceedings.mlr.press/v134/lee21a.html.

[177] Victor Lempitsky, Andrea Vedaldi, and Dmitry Ulyanov. "Deep Image Prior." In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 9446–9454. DOI: 10.1109/CVPR.2018.00984.

[178] Aviad Levis, Daeyoung Lee, Joel A. Tropp, Charles F. Gammie, and Katherine L. Bouman. "Inference of Black Hole Fluid-Dynamics from Sparse Interferometric Measurements." In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 2320–2329. DOI: 10.1109/ICCV48922.2021.00234.

[179] Lianlin Li, Long Gang Wang, Fernando L. Teixeira, Che Liu, Arye Nehorai, and Tie Jun Cui. "DeepNIS: Deep Neural Network for Nonlinear Electromagnetic Inverse Scattering." In: *IEEE Transactions on Antennas and Propagation* 67.3 (2019), pp. 1819–1825. DOI: 10.1109/TAP.2018.2885437.

[180] Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. "Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization." In: *Journal of Machine Learning Research* 18.185 (2018), pp. 1–52. URL: http://jmlr.org/papers/v18/16-558.html.

[181] Xiang Li, Soo Min Kwon, Ismail R. Alkhouri, Saiprasad Ravishankar, and Qing Qu. *Decoupled Data Consistency with Diffusion Purification for Image Restoration*. 2024. arXiv: 2403.06054 [eess.IV]. URL: https://arxiv.org/abs/2403.06054.

[182] Xiner Li, Yulai Zhao, Chenyu Wang, Gabriele Scalia, Gokcen Eraslan, Surag Nair, Tommaso Biancalani, Shuiwang Ji, Aviv Regev, Sergey Levine, and Masatoshi Uehara. *Derivative-Free Guidance in Continuous and Discrete Diffusion Models with Soft Value-Based Decoding*. 2024. arXiv: 2408.08252 [cs.LG]. URL: https://arxiv.org/abs/2408.08252.

[183] Yunzhe Li, Yujia Xue, and Lei Tian. "Deep Speckle Correlation: A Deep Learning Approach toward Scalable Imaging through Scattering Media." In: *Optica* 5 (Sept. 2018), pp. 1181–1190. DOI: 10.1364/OPTICA.5.001181.

[184] Zongyi Li, Nikola Borislavov Kovachki, Kamyar Azizzadenesheli, Burigede liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. "Fourier

Neural Operator for Parametric Partial Differential Equations." In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=c8P9NQVtmnO.

[185] Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, and Anima Anandkumar. "Physics-informed Neural Operator for Learning Partial Differential Equations." In: *ACM/JMS Journal of Data Science* 1.3 (2024), pp. 1–27. DOI: 10.1145/3648506.

[186] Zhi-Pei Liang and Paul C. Lauterbur. *Principles of Magnetic Resonance Imaging: A Signal Processing Perspective*. IEEE Press series in biomedical engineering. SPIE Optical Engineering Press, 2000. ISBN: 9780819435163. URL: https://books.google.com/books?id=sRyEQgAACAAJ.

[187] Dong C. Liu and Jorge Nocedal. "On the Limited Memory BFGS Method for Large Scale Optimization." In: *Mathematical Programming* 45.1–3 (Aug. 1989), pp. 503–528. ISSN: 0025-5610. DOI: 10.1007/BF01589116.

[188] Hsiou-Yuan Liu, Dehong Liu, Hassan Mansour, Petros T. Boufounos, Laura Waller, and Ulugbek S. Kamilov. "SEAGLE: Sparsity-Driven Image Reconstruction Under Multiple Scattering." In: *IEEE Transactions on Computational Imaging* 4.1 (2018), pp. 73–86. DOI: 10.1109/TCI.2017.2764461.

[189] Jiaming Liu, Rushil Anirudh, Jayaraman J. Thiagarajan, Stewart He, K. Aditya Mohan, Ulugbek S. Kamilov, and Hyojin Kim. "DOLCE: A Model-Based Probabilistic Diffusion Framework for Limited-Angle CT Reconstruction." In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 10464–10474. DOI: 10.1109/ICCV51070.2023.00963.

[190] Jiaming Liu, Yu Sun, Cihat Eldeniz, Weijie Gan, Hongyu An, and Ulugbek S. Kamilov. "RARE: Image Reconstruction Using Deep Priors Learned Without Groundtruth." In: *IEEE Journal of Selected Topics in Signal Processing* 14.6 (2020), pp. 1088–1099. DOI: 10.1109/JSTSP.2020.2998402.

[191] Jiaming Liu, Yu Sun, Weijie Gan, Xiaojian Xu, Brendt Wohlberg, and Ulugbek S. Kamilov. "SGD-Net: Efficient Model-Based Deep Learning With Theoretical Guarantees." In: *IEEE Transactions on Computational Imaging* 7 (2021), pp. 598–610. DOI: 10.1109/TCI.2021.3085534.

[192] Lijun Liu and Michael Gurnis. "Simultaneous Inversion of Mantle Properties and Initial Conditions Using an Adjoint of Mantle Convection." In: *Journal of Geophysical Research: Solid Earth* 113.B8 (2008). DOI: 10.1029/2008JB005594. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2008JB005594. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2008JB005594.

[193] Renhao Liu, Yu Sun, Jiabei Zhu, Lei Tian, and Ulugbek Kamilov. "Recovery of Continuous 3D Refractive Index Maps from Discrete Intensity-Only Mea-

surements Using Neural Fields." In: *Nature Machine Intelligence* 4 (Sept. 2022), pp. 1–11. DOI: 10.1038/s42256-022-00530-3.

[194] Yaning Liu, Fenglin Niu, Min Chen, and Wencai Yang. "3-D Crustal and Uppermost Mantle Structure beneath NE China Revealed by Ambient Noise Adjoint Tomography." In: *Earth and Planetary Science Letters* 461 (2017), pp. 20–29. ISSN: 0012-821X. DOI: 10.1016/j.epsl.2016.12.029. URL: https://www.sciencedirect.com/science/article/pii/S0012821X16307427.

[195] Yiming Liu, Yanwei Pang, Ruiqi Jin, Yonghong Hou, and Xuelong Li. "Reinforcement Learning and Transformer for Fast Magnetic Resonance Imaging Scan." In: *IEEE Transactions on Emerging Topics in Computational Intelligence* 8.3 (2024), pp. 2310–2323. DOI: 10.1109/TETCI.2024.3358180.

[196] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. "Deep Learning Face Attributes in the Wild." In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 3730–3738. DOI: 10.1109/ICCV.2015.425.

[197] Ilya Loshchilov and Frank Hutter. "Decoupled Weight Decay Regularization." In: *International Conference on Learning Representations*. 2019. URL: https://openreview.net/forum?id=Bkg6RiCqY7.

[198] Aaron Lou, Chenlin Meng, and Stefano Ermon. "Discrete Diffusion Modeling by Estimating the Ratios of the Data Distribution." In: *Forty-first International Conference on Machine Learning*. 2024. URL: https://openreview.net/forum?id=CNicRIVIPA.

[199] Mathias Louboutin, Michael Lange, Fabio Luporini, Navjot Kukreja, Philipp A. Witte, Felix J. Herrmann, Paulius Velesko, and Gerard J. Gorman. "Devito (v3.1.0): An Embedded Domain-specific Language for Finite Differences and Geophysical Exploration." In: *Geoscientific Model Development* 12.3 (2019), pp. 1165–1187. DOI: 10.5194/gmd-12-1165-2019. URL: https://www.geosci-model-dev.net/12/1165/2019/.

[200] Jan Lukas, Jessica Fridrich, and Miroslav Goljan. "Digital Camera Identification from Sensor Pattern Noise." In: *IEEE Transactions on Information Forensics and Security* 1.2 (2006), pp. 205–214. DOI: 10.1109/TIFS.2006.873602.

[201] Guanxiong Luo, Moritz Blumenthal, Martin Heide, and Martin Uecker. "Bayesian MRI Reconstruction with Joint Uncertainty Estimation Using Diffusion Models." In: *Magnetic Resonance in Medicine* 90 (Mar. 2023), pp. 295–311. DOI: 10.1002/mrm.29624.

[202] Guanxiong Luo, Na Zhao, Wenhao Jiang, Edward Hui, and Peng Cao. "MRI Reconstruction Using Deep Bayesian Estimation." In: *Magnetic Resonance in Medicine* 84 (Apr. 2020), pp. 2246–2261. DOI: 10.1002/mrm.28274.

[203] Yanchen Luo, Junfeng Fang, Sihang Li, Zhiyuan Liu, Jiancan Wu, An Zhang, Wenjie Du, and Xiang Wang. "Text-Guided Small Molecule Generation via Diffusion Model." In: *iScience* 27 (Sept. 2024), p. 110992. DOI: 10.1016/j.isci.2024.110992.

[204] Michael Lustig, David Donoho, and John Pauly. "Sparse MRI: The Application of Compressed Sensing for Rapid MR Imaging." In: *Magnetic resonance in medicine : official journal of the Society of Magnetic Resonance in Medicine / Society of Magnetic Resonance in Medicine* 58 (Dec. 2007), pp. 1182–95. DOI: 10.1002/mrm.21391.

[205] Michael Lustig, David L. Donoho, Juan M. Santos, and John M. Pauly. "Compressed Sensing MRI." In: *IEEE Signal Processing Magazine* 25.2 (2008), pp. 72–82. DOI: 10.1109/MSP.2007.914728.

[206] Yi-An Ma, Tianqi Chen, and Emily Fox. "A Complete Recipe for Stochastic Gradient MCMC." In: *Advances in Neural Information Processing Systems*. Ed. by C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett. Vol. 28. Curran Associates, Inc., 2015. URL: https://proceedings.neurips.cc/paper_files/paper/2015/file/9a4400501febb2a95e79248486a5f6d3-Paper.pdf.

[207] Shiqian Ma, Wotao Yin, Yin Zhang, and Amit Chakraborty. "An Efficient Algorithm for Compressed MR Imaging Using Total Variation and Wavelets." In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. 2008, pp. 1–8. DOI: 10.1109/CVPR.2008.4587391.

[208] Yanting Ma, Hassan Mansour, Dehong Liu, Petros T. Boufounos, and Ulugbek S. Kamilov. "Accelerated Image Reconstruction for Nonlinear Diffractive Imaging." In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2018, pp. 6473–6477. DOI: 10.1109/ICASSP.2018.8462400.

[209] Andrew L. Maas, Awni Y. Hannun, Andrew Y. Ng, et al. "Rectifier Nonlinearities Improve Neural Network Acoustic Models." In: *Proceedings of the Thirtieth International Conference on Machine Learning (ICML)*. Vol. 30. 1. Atlanta, GA. 2013, p. 3.

[210] Laurens van der Maaten and Geoffrey Hinton. "Visualizing Data Using t-SNE." In: *Journal of Machine Learning Research* 9.86 (2008), pp. 2579–2605. URL: http://jmlr.org/papers/v9/vandermaaten08a.html.

[211] Ross Maguire, Brandon Schmandt, Jiaqi Li, Chengxin Jiang, Guoliang Li, Justin Wilgus, and Min Chen. "Magma Accumulation at Depths of Prior Rhyolite Storage beneath Yellowstone Caldera." In: *Science (New York, N.Y.)* 378 (Dec. 2022), pp. 1001–1004. DOI: 10.1126/science.ade0347.

[212] Sébastien Marcel and Yann Rodriguez. "Torchvision the Machine-Vision Package of Torch." In: *Proceedings of the 18th ACM International Conference on Multimedia*. MM '10. Firenze, Italy: Association for Computing

Machinery, 2010, pp. 1485–1488. ISBN: 9781605589336. DOI: 10.1145/1873951.1874254. URL: https://doi.org/10.1145/1873951.1874254.

[213] Daniel Marcus, Tracy Wang, Jamie Parker, John Csernansky, John Morris, and Randy Buckner. "Open Access Series of Imaging Studies (OASIS): Cross-Sectional MRI Data in Young, Middle Aged, Nondemented, and Demented Older Adults." In: *Journal of Cognitive Neuroscience* 19 (Oct. 2007), pp. 1498–507. DOI: 10.1162/jocn.2007.19.9.1498.

[214] Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. "A Variational Perspective on Solving Inverse Problems with Diffusion Models." In: *The Twelfth International Conference on Learning Representations*. 2024. URL: https://openreview.net/forum?id=1YO4EE3SPB.

[215] Ségolène Martin, Anne Gagneux, Paul Hagemann, and Gabriele Steidl. *PnP-Flow: Plug-and-Play Image Restoration with Flow Matching*. 2024. arXiv: 2410.02423 [cs.CV]. URL: https://arxiv.org/abs/2410.02423.

[216] Filippo Martinini, Mauro Mangia, Alex Marchioni, Riccardo Rovatti, and Gianluca Setti. "A Deep Learning Method for Optimal Undersampling Patterns and Image Recovery for MRI Exploiting Losses and Projections." In: *IEEE Journal of Selected Topics in Signal Processing* 16.4 (2022), pp. 713–724. DOI: 10.1109/JSTSP.2022.3171082.

[217] Tim Meinhardt, Michael Moeller, Caner Hazirbas, and Daniel Cremers. "Learning Proximal Operators: Using Denoising Networks for Regularizing Inverse Imaging Problems." In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 1799–1808. DOI: 10.1109/ICCV.2017.198.

[218] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. "SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations." In: *International Conference on Learning Representations*. 2021. URL: https://api.semanticscholar.org/CorpusID:245704504.

[219] Bjoern Menze, András Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahaniy, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboomy, Roland Wiest, Levente Lancziy, Elizabeth Gerstnery, Marc-Andr´e Webery, Tal Arbel, Brian Avants, Nicholas Ayache, Patricia Buendia, Louis Collins, Nicolas Cordier, and Koen Van Leemput. "The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS)." In: *IEEE Transactions on Medical Imaging* 99 (Dec. 2014). DOI: 10.1109/TMI.2014.2377694.

[220] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. "Equation of State Calculations by Fast Computing Machines." In: *The Journal of Chemical Physics* 21.6 (June 1953), pp. 1087–1092. ISSN: 0021-9606. DOI: 10.1063/1.1699114. eprint:

https://pubs.aip.org/aip/jcp/article-pdf/21/6/1087/18802390/1087_1_online.pdf. URL: https://doi.org/10.1063/1.1699114.

[221] Christopher Metzler, Phillip Schniter, Ashok Veeraraghavan, and Richard Baraniuk. "prDeep: Robust Phase Retrieval with a Flexible Deep Network." In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, July 2018, pp. 3501–3510. URL: https://proceedings.mlr.press/v80/metzler18a.html.

[222] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. "V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation." In: *2016 Fourth International Conference on 3D Vision (3DV)*. 2016, pp. 565–571. DOI: 10.1109/3DV.2016.79.

[223] Yosuke Mizuno. "GRMHD Simulations and Modeling for Jet Formation and Acceleration Region in AGNs." In: *Universe* 8.2 (Jan. 2022), p. 85. ISSN: 2218-1997. DOI: 10.3390/universe8020085. URL: http://dx.doi.org/10.3390/universe8020085.

[224] Alexander Quinn Nichol and Prafulla Dhariwal. "Improved Denoising Diffusion Probabilistic Models." In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, July 2021, pp. 8162–8171. URL: https://proceedings.mlr.press/v139/nichol21a.html.

[225] Hunter Nisonoff, Junhao Xiong, Stephan Allenspach, and Jennifer Listgarten. "Unlocking Guidance for Discrete State-Space Diffusion and Flow Models." In: *The Thirteenth International Conference on Learning Representations*. 2025. URL: https://openreview.net/forum?id=XsgHl54yO7.

[226] Nikolas Nüsken and Sebastian Reich. *Note on Interacting Langevin Diffusions: Gradient Structure and Ensemble Kalman Sampler by Garbuno-Inigo, Hoffmann, Li and Stuart*. 2019. arXiv: 1908.10890 [math.DS]. URL: https://arxiv.org/abs/1908.10890.

[227] Dean Oliver and Yan Chen. "Recent Progress on Reservoir History Matching: A Review." In: *Computational Geosciences* 15 (July 2011), pp. 185–221. DOI: 10.1007/s10596-010-9194-2.

[228] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. "Neural Discrete Representation Learning." In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/7a98af17e63a0ac09ce2e96d03992fbc-Paper.pdf.

[229] Wei Peng, Li Feng, Guoying Zhao, and Fang Liu. "Learning Optimal K-space Acquisition and Reconstruction Using Physics-Informed Neural Networks." In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 20762–20771. DOI: 10.1109/CVPR52688.2022.02013.

[230] Marcelo Pereyra, Luis A. Vargas-Mieles, and Konstantinos C. Zygalakis. "The Split Gibbs Sampler Revisited: Improvements to Its Algorithmic Structure and Augmented Target Distribution." In: *SIAM Journal on Imaging Sciences* 16.4 (2023), pp. 2040–2071. DOI: 10.1137/22M1506122. eprint: https://doi.org/10.1137/22M1506122. URL: https://doi.org/10.1137/22M1506122.

[231] Luis Pineda, Sumana Basu, Adriana Romero, Roberto Calandra, and Michal Drozdzal. "Active MR K-Space Sampling with Reinforcement Learning." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*. Ed. by Anne L. Martel, Purang Abolmaesumi, Danail Stoyanov, Diana Mateus, Maria A. Zuluaga, S. Kevin Zhou, Daniel Racoceanu, and Leo Joskowicz. Cham: Springer International Publishing, 2020, pp. 23–33. ISBN: 978-3-030-59713-9. DOI: 10.1007/978-3-030-59713-9_3.

[232] Rene-Edouard Plessix. "A Review of the Adjoint-State Method for Computing the Gradient of a Functional with Geophysical Applications." In: *Geophysical Journal International* 167.2 (2006), pp. 495–503. DOI: 10.1111/j.1365-246X.2006.02978.x.

[233] Oliver Porth et al. "The Event Horizon General Relativistic Magnetohydrodynamic Code Comparison Project." In: *The Astrophysical Journal Supplement Series* 243.2 (Aug. 2019), p. 26. DOI: 10.3847/1538-4365/ab29fd. URL: https://dx.doi.org/10.3847/1538-4365/ab29fd.

[234] Oula Puonti, Juan Iglesias, and Koen Van Leemput. "Fast and Sequence-Adaptive Whole-Brain Segmentation Using Parametric Bayesian Modeling." In: *NeuroImage* 143 (Sept. 2016). DOI: 10.1016/j.neuroimage.2016.09.011.

[235] Tran Minh Quan, Thanh Nguyen-Duc, and Won-Ki Jeong. "Compressed Sensing MRI Reconstruction Using a Generative Adversarial Network With a Cyclic Loss." In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1488–1497. DOI: 10.1109/TMI.2018.2820120.

[236] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. "Learning Transferable Visual Models From Natural Language Supervision." In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, July 2021, pp. 8748–8763. URL: https://proceedings.mlr.press/v139/radford21a.html.

[237] Chaithya G. Radhakrishna, Zaccharie Ramzi, and Philippe Ciuciu. "Hybrid Learning of Non-Cartesian K-Space Trajectory and MR Image Reconstruction Networks." In: *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*. 2022, pp. 1–5. DOI: 10.1109/ISBI52829.2022.9761408.

[238] Archit Raj, Srikrishnan Vishwanathan, Bhavya Ajani, Karthik Krishnan, and Harsh Agarwal. "Automatic Knee Cartilage Segmentation Using Fully Volumetric Convolutional Neural Networks for Evaluation of Osteoarthritis." In: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. 2018, pp. 851–854. DOI: 10.1109/ISBI.2018.8363705.

[239] Rajikha Raja and Neelam Sinha. "Adaptive K-Space Sampling Design for Edge-Enhanced DCE-MRI Using Compressed Sensing." In: *Magnetic Resonance Imaging* 32 (Sept. 2014). DOI: 10.1016/j.mri.2013.12.022.

[240] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. *Hierarchical Text-Conditional Image Generation with CLIP Latents*. 2022. arXiv: 2204.06125 [cs.CV]. URL: https://arxiv.org/abs/2204.06125.

[241] Stephan Rasp, Stephan Hoyer, Alexander Merose, Ian Langmore, Peter Battaglia, Tyler Russell, Alvaro Sanchez-Gonzalez, Vivian Yang, Rob Carver, Shreya Agrawal, Matthew Chantry, Zied Ben Bouallegue, Peter Dueben, Carla Bromberg, Jared Sisk, Luke Barrington, Aaron Bell, and Fei Sha. "WeatherBench 2: A Benchmark for the Next Generation of Data-Driven Global Weather Models." In: *Journal of Advances in Modeling Earth Systems* 16.6 (2024). e2023MS004019 2023MS004019, e2023MS004019. DOI: 10.1029/2023MS004019. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2023MS004019. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2023MS004019.

[242] Saiprasad Ravishankar and Yoram Bresler. "Adaptive Sampling Design for Compressed Sensing MRI." In: *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. 2011, pp. 3751–3755. DOI: 10.1109/IEMBS.2011.6090639.

[243] Saiprasad Ravishankar and Yoram Bresler. "MR Image Reconstruction From Highly Undersampled K-Space Data by Dictionary Learning." In: *IEEE Transactions on Medical Imaging* 30.5 (2011), pp. 1028–1041. DOI: 10.1109/TMI.2010.2090538.

[244] Benjamin Remy, Francois Lanusse, Niall Jeffrey, Jia Liu, Jean-Luc Starck, Ken Osato, and Tim Schrabback. "Probabilistic Mass-Mapping with Neural Score Estimation." In: *Astronomy & Astrophysics* 672 (Dec. 2022). DOI: 10.1051/0004-6361/202243054.

[245] Yinuo Ren, Haoxuan Chen, Grant M. Rotskoff, and Lexing Ying. "How Discrete and Continuous Diffusion Meet: Comprehensive Analysis of Discrete Diffusion Models via a Stochastic Integral Framework." In: *The Thirteenth*

International Conference on Learning Representations. 2025. URL: https://openreview.net/forum?id=6awxwQEI82.

[246] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. "Stochastic Backpropagation and Approximate Inference in Deep Generative Models." In: *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*. Vol. 32. JMLR Workshop and Conference Proceedings. JMLR.org, 2014, pp. 1278–1286. URL: http://proceedings.mlr.press/v32/rezende14.html.

[247] Gareth O. Roberts and Richard L. Tweedie. "Exponential Convergence of Langevin Distributions and Their Discrete Approximations." In: *Bernoulli* 2.4 (1996), pp. 341–363. DOI: 10.2307/3318418.

[248] Yaniv Romano, Michael Elad, and Peyman Milanfar. "The Little Engine that Could: Regularization by Denoising (RED)." In: *SIAM Journal on Imaging Sciences* 10 (Nov. 2016). DOI: 10.1137/16M1102884.

[249] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. "High-Resolution Image Synthesis with Latent Diffusion Models." In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 10674–10685. DOI: 10.1109/CVPR52688.2022.01042.

[250] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation." In: *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer. 2015, pp. 234–241. DOI: 10.1007/978-3-319-24574-4_28.

[251] Litu Rout, Negin Raoof, Giannis Daras, Constantine Caramanis, Alex Dimakis, and Sanjay Shakkottai. "Solving Linear Inverse Problems Provably via Posterior Sampling with Latent Diffusion Models." In: *Thirty-seventh Conference on Neural Information Processing Systems*. 2023. URL: https://openreview.net/forum?id=XKBFdYwfRo.

[252] Ernest K. Ryu, Jialin Liu, Sicheng Wang, Xiaohan Chen, Zhangyang Wang, and Wotao Yin. "Plug-and-Play Methods Provably Converge with Properly Trained Denoisers." In: *International Conference on Machine Learning*. 2019. URL: https://api.semanticscholar.org/CorpusID:153311703.

[253] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J. Fleet, and Mohammad Norouzi. "Image Super-Resolution via Iterative Refinement." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.4 (2023), pp. 4713–4726. DOI: 10.1109/TPAMI.2022.3204461.

[254] Fritz Schick. "Tissue Segmentation: A Crucial Tool for Quantitative MRI and Visualization of Anatomical Structures." In: *Magma (New York, N.Y.)* 29 (Apr. 2016). DOI: 10.1007/s10334-016-0549-0.

[255] Jo Schlemper, Jose Caballero, Joseph V. Hajnal, Anthony N. Price, and Daniel Rueckert. "A Deep Cascade of Convolutional Neural Networks for Dynamic MR Image Reconstruction." In: *IEEE Transactions on Medical Imaging* 37.2 (2018), pp. 491–503. DOI: 10.1109/TMI.2017.2760978.

[256] Martin Shepherd. "Difmap: An Interactive Program for Synthesis Imaging." In: *Astronomical Data Analysis Software and Systems VI*. Vol. 125. 1997, p. 77. URL: https://www.cv.nrao.edu/adass/adassVI/shepherdm.html.

[257] Martin Shepherd. "Difmap: Synthesis Imaging of Visibility Data." In: *Astrophysics Source Code Library* (2011), p. 1103. URL: https://ascl.net/1103.001.

[258] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. "Mastering the Game of Go with Deep Neural Networks and Tree Search." In: *Nature* 529.7587 (Jan. 2016), pp. 484–489. DOI: 10.1038/nature16961.

[259] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. "Deep Unsupervised Learning Using Nonequilibrium Thermodynamics." In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Francis Bach and David Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 2256–2265. URL: https://proceedings.mlr.press/v37/sohl-dickstein15.html.

[260] Bowen Song, Soo Min Kwon, Zecheng Zhang, Xinyu Hu, Qing Qu, and Liyue Shen. "Solving Inverse Problems with Latent Diffusion Models via Hard Data Consistency." In: *The Twelfth International Conference on Learning Representations*. 2024. URL: https://openreview.net/forum?id=j8hdRqOUhN.

[261] Jiaming Song, Chenlin Meng, and Stefano Ermon. "Denoising Diffusion Implicit Models." In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=St1giarCHLP.

[262] Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. "Pseudoinverse-Guided Diffusion Models for Inverse Problems." In: *International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=9_gsMA8MRKQ.

[263] Jiaming Song, Qinsheng Zhang, Hongxu Yin, Morteza Mardani, Ming-Yu Liu, Jan Kautz, Yongxin Chen, and Arash Vahdat. "Loss-Guided Diffusion Models for Plug-and-Play Controllable Generation." In: *Proceedings of the 40th International Conference on Machine Learning*. Ed. by

Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett. Vol. 202. Proceedings of Machine Learning Research. PMLR, July 2023, pp. 32483–32498. URL: https://proceedings.mlr.press/v202/song23k.html.

[264] Yang Song and Stefano Ermon. "Generative Modeling by Estimating Gradients of the Data Distribution." In: *Neural Information Processing Systems*. 2019. URL: https://api.semanticscholar.org/CorpusID:196470871.

[265] Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. "Solving Inverse Problems in Medical Imaging with Score-Based Generative Models." In: *International Conference on Learning Representations*. 2022. URL: https://openreview.net/forum?id=vaRCHVj0uGI.

[266] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. "Score-Based Generative Modeling through Stochastic Differential Equations." In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=PxTIG12RRHS.

[267] Anuroop Sriram, Jure Zbontar, Tullie Murrell, Aaron Defazio, C. Lawrence Zitnick, Nafissa Yakubova, Florian Knoll, and Patricia Johnson. "End-to-End Variational Networks for Accelerated MRI Reconstruction." In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part II*. Lima, Peru: Springer-Verlag, 2020, pp. 64–73. ISBN: 978-3-030-59712-2. DOI: 10.1007/978-3-030-59713-9_7. URL: https://doi.org/10.1007/978-3-030-59713-9_7.

[268] Andrew Stuart. "Inverse Problems: A Bayesian Perspective." In: *Acta Numerica* 19 (May 2010), pp. 451–559. DOI: 10.1017/S0962492910000061.

[269] He Sun and Katherine L. Bouman. "Deep Probabilistic Imaging: Uncertainty Quantification and Multi-Modal Solution Characterization for Computational Imaging." In: *Proceedings of the AAAI Conference on Artificial Intelligence* 35.3 (May 2021), pp. 2628–2637. DOI: 10.1609/aaai.v35i3.16366. URL: https://ojs.aaai.org/index.php/AAAI/article/view/16366.

[270] He Sun, Adrian V. Dalca, and Katherine L. Bouman. "Learning a Probabilistic Strategy for Computational Imaging Sensor Selection." In: *2020 IEEE International Conference on Computational Photography (ICCP)*. 2020, pp. 1–12. DOI: 10.1109/ICCP48838.2020.9105133.

[271] Liyan Sun, Zhiwen Fan, Xinghao Ding, Yue Huang, and John Paisley. "Joint CS-MRI Reconstruction and Segmentation with a Unified Deep Network." In: *Information Processing in Medical Imaging*. Ed. by Albert C. S. Chung, James C. Gee, Paul A. Yushkevich, and Siqi Bao. Cham: Springer In-

ternational Publishing, 2019, pp. 492–504. ISBN: 978-3-030-20351-1. DOI: 10.1007/978-3-030-20351-1_38.

[272] Yu Sun, Brendt Wohlberg, and Ulugbek S. Kamilov. "An Online Plug-and-Play Algorithm for Regularized Image Reconstruction." In: *IEEE Transactions on Computational Imaging* 5.3 (2019), pp. 395–408. DOI: 10.1109/TCI.2019.2893568.

[273] Yu Sun, Zihui Wu, Yifan Chen, Berthy T. Feng, and Katherine L. Bouman. "Provable Probabilistic Imaging Using Score-Based Generative Priors." In: *IEEE Transactions on Computational Imaging* 10 (2024), pp. 1290–1305. DOI: 10.1109/TCI.2024.3449114.

[274] Yu Sun, Zhihao Xia, and Ulugbek Kamilov. "Efficient and Accurate Inversion of Multiple Scattering with Deep Learning." In: *Optics Express* 26 (May 2018), pp. 14678–14688. DOI: 10.1364/OE.26.014678.

[275] Yongjin Sung, Wonshik Choi, Christopher Fang-Yen, Kamran Badizadegan, Ramachandra Dasari, and Michael Feld. "Optical Diffraction Tomography for High Resolution Live Cell Imaging." In: *Optics Express* 17 (Jan. 2009), pp. 266–277. DOI: 10.1364/OE.17.000266.

[276] Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Daniel MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. "PDEBench: An Extensive Benchmark for Scientific Machine Learning." In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh. Vol. 35. Curran Associates, Inc., 2022, pp. 1596–1611. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/0a9747136d411fb83f0cf81820d44afb-Paper-Datasets_and_Benchmarks.pdf.

[277] Haoyue Tang, Tian Xie, Aosong Feng, Hanyu Wang, Chenyang Zhang, and Yang Bai. "Solving General Noisy Inverse Problem via Posterior Sampling: A Policy Gradient Viewpoint." In: *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*. Ed. by Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li. Vol. 238. Proceedings of Machine Learning Research. PMLR, May 2024, pp. 2116–2124. URL: https://proceedings.mlr.press/v238/tang24b.html.

[278] Zachary Teed and Jia Deng. *RAFT: Recurrent All-Pairs Field Transforms for Optical Flow*. 2020. arXiv: 2003.12039 [cs.CV]. URL: https://arxiv.org/abs/2003.12039.

[279] Kerem Tezcan, Christian Baumgartner, Roger Luechinger, Klaas Pruessmann, and Ender Konukoglu. "MR Image Reconstruction Using Deep Density Priors." In: *IEEE Transactions on Medical Imaging* PP (Dec. 2018), pp. 1–1. DOI: 10.1109/TMI.2018.2887072.

[280] A. Richard. Thompson, James M. Moran, and SpringerLink (Online service). *Interferometry and Synthesis in Radio Astronomy*. 3rd edition 2017. Astronomy and Astrophysics Library. Springer International Publishing, 2017. DOI: 10.1007/978-3-319-44431-4.

[281] Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. "Visual Autoregressive Modeling: Scalable Image Generation via Next-Scale Prediction." In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024. URL: https://openreview.net/forum?id=gojL67CfS8.

[282] Lei Tian and Laura Waller. "3D Intensity and Phase Imaging from Light Field Measurements in an LED Array Microscope." In: *Optica* 2 (Jan. 2015), pp. 104–111. DOI: 10.1364/OPTICA.2.000104.

[283] Brian L. Trippe, Jason Yim, Doug Tischer, David Baker, Tamara Broderick, Regina Barzilay, and Tommi S. Jaakkola. "Diffusion Probabilistic Modeling of Protein Backbones in 3D for the Motif-scaffolding Problem." In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=6TxBxqNME1Y.

[284] Ben Tunbridge, Ian Harrison, and Michael L. Brown. "Radio–Optical Galaxy Shape Correlations in the COSMOS Field." In: *Monthly Notices of the Royal Astronomical Society* 463.3 (Sept. 2016), pp. 3339–3353. ISSN: 0035-8711. DOI: 10.1093/mnras/stw2224. eprint: https://academic.oup.com/mnras/article-pdf/463/3/3339/18243295/stw2224.pdf. URL: https://doi.org/10.1093/mnras/stw2224.

[285] Martin Uecker, Peng Lai, Mark Murphy, Patrick Virtue, Michael Elad, Shreyas Vasanawala, and Michael Lustig. "ESPIRiT—An Eigenvalue Approach to Autocalibrating Parallel MRI: Where SENSE Meets GRAPPA." In: *Magnetic Resonance in Medicine* 71 (Mar. 2014). DOI: 10.1002/mrm.24751.

[286] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. "Instance Normalization: The Missing Ingredient for Fast Stylization." In: (July 2016). DOI: 10.48550/arXiv.1607.08022.

[287] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. *Towards Accurate Generative Models of Video: A New Metric & Challenges*. 2019. arXiv: 1812.01717 [cs.CV]. URL: https://arxiv.org/abs/1812.01717.

[288] Shreyas S. Vasanawala, Mark J. Murphy, Marcus T. Alley, Peng Lai, Kurt Keutzer, John M. Pauly, and Michael Lustig. "Practical Parallel Imaging Compressed Sensing MRI: Summary of Two Years of Experience in Accelerating Body MRI of Pediatric Patients." In: *2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. 2011, pp. 1039–1043. DOI: 10.1109/ISBI.2011.5872579.

[289] Jaganathan Vellagoundar and Ramasubba Reddy Machireddy. "A Robust Adaptive Sampling Method for Faster Acquisition of MR Images." In: *Magnetic Resonance Imaging* 33.5 (June 2015), pp. 635–643. ISSN: 0730-725X. DOI: 10.1016/j.mri.2015.01.008. URL: https://doi.org/10.1016/j.mri.2015.01.008.

[290] Santosh Vempala and Andre Wibisono. "Rapid Convergence of the Unadjusted Langevin Algorithm: Isoperimetry Suffices." In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/65a99bb7a3115fdede20da98b08a370f-Paper.pdf.

[291] Singanallur V. Venkatakrishnan, Charles A. Bouman, and Brendt Wohlberg. "Plug-and-Play Priors for Model Based Reconstruction." In: *2013 IEEE Global Conference on Signal and Information Processing*. 2013, pp. 945–948. DOI: 10.1109/GlobalSIP.2013.6737048.

[292] Pascal Vincent. "A Connection Between Score Matching and Denoising Autoencoders." In: *Neural Computation* 23.7 (2011), pp. 1661–1674. DOI: 10.1162/NECO_a_00142.

[293] Jean Virieux, Amir Asnaashari, Romain Brossier, Ludovic Métivier, Alessandra Ribodetti, and Wei Zhou. "An Introduction to Full Waveform Inversion." In: *Encyclopedia of Exploration Geophysics*. Society of Exploration Geophysicists, Jan. 2014. ISBN: 9781560803010. DOI: 10.1190/1.9781560803027.entry6. URL: https://doi.org/10.1190/1.9781560803027.entry6.

[294] Jean Virieux and Stéphane Operto. "An Overview of Full-waveform Inversion in Exploration Geophysics." In: *Geophysics* 74 (Nov. 2009), WCC1–WCC26. DOI: 10.1190/1.3238367.

[295] Maxime Vono, Nicolas Dobigeon, and Pierre Chainais. "High-Dimensional Gaussian Sampling: A Review and a Unifying Approach Based on a Stochastic Proximal Point Algorithm." In: *SIAM Review* 64.1 (2022), pp. 3–56. DOI: 10.1137/20M1371026. eprint: https://doi.org/10.1137/20M1371026. URL: https://doi.org/10.1137/20M1371026.

[296] Maxime Vono, Nicolas Dobigeon, and Pierre Chainais. "Split-and-Augmented Gibbs Sampler—Application to Large-Scale Inference Problems." In: *IEEE Transactions on Signal Processing* 67.6 (2019), pp. 1648–1661. DOI: 10.1109/TSP.2019.2894825.

[297] Austin Wang, Hongkai Zheng, Zihui Wu, Ricardo Baptista, Daniel Zhengyu Huang, and Yisong Yue. "Ensemble Kalman Sampling and Diffusion Prior in Tandem: A Split Gibbs Framework." In: *Frontiers in Probabilistic Inference: Learning Meets Sampling, ICLR 2025*. 2025. URL: https://openreview.net/forum?id=3DfCxd0yxo.

[298] Chengyan Wang, Jun Lyu, Shuo Wang, Chen Qin, Kunyuan Guo, Xinyu Zhang, Xiaotong Yu, Yan Li, Fanwen Wang, Jianhua Jin, Zhang Shi, Ziqiang Xu, Yapeng Tian, Sha Hua, Zhensen Chen, Meng Liu, Mengting Sun, Xutong Kuang, Kang Wang, and Xiaobo Qu. "CMRxRecon: A Publicly Available K-space Dataset and Benchmark to Advance Deep Learning for Cardiac MRI." In: *Scientific Data* 11 (June 2024). DOI: 10.1038/s41597-024-03525-4.

[299] Guanhua Wang, Tianrui Luo, Jon-Fredrik Nielsen, Douglas C. Noll, and Jeffrey A. Fessler. "B-Spline Parameterized Joint Optimization of Reconstruction and K-Space Trajectories (BJORK) for Accelerated 2D MRI." In: *IEEE Transactions on Medical Imaging* 41.9 (2022), pp. 2318–2330. DOI: 10.1109/TMI.2022.3161875.

[300] Guanhua Wang, Jon-Fredrik Nielsen, and Jeff Fessler. "Stochastic Optimization of Three-Dimensional Non-Cartesian Sampling Trajectory." In: *Magnetic Resonance in Medicine* 90 (Apr. 2023). DOI: 10.1002/mrm.29645.

[301] Haifeng Wang, Dong Liang, and Leslie Ying. "Pseudo 2D Random Sampling for Compressed Sensing MRI." In: *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. 2009, pp. 2672–2675. DOI: 10.1109/IEMBS.2009.5334086.

[302] Jiechao Wang, Qinqin Yang, Qizhi Yang, Lina Xu, Congbo Cai, and Shuhui Cai. "Joint Optimization of Cartesian Sampling Patterns and Reconstruction for Single-Contrast and Multi-Contrast Fast Magnetic Resonance Imaging." In: *Computer Methods and Programs in Biomedicine* (Sept. 2022). DOI: 10.1016/j.cmpb.2022.107150.

[303] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. *ModelScope Text-to-Video Technical Report*. 2023. arXiv: 2308.06571 [cs.CV]. URL: https://arxiv.org/abs/2308.06571.

[304] Lihong V. Wang and Song Hu. "Photoacoustic Tomography: In Vivo Imaging from Organelles to Organs." In: *Science* 335.6075 (2012), pp. 1458–1462. DOI: 10.1126/science.1216210. eprint: https://www.science.org/doi/pdf/10.1126/science.1216210. URL: https://www.science.org/doi/abs/10.1126/science.1216210.

[305] Shanshan Wang, Zhenghang Su, Leslie Ying, Xi Peng, Shun Zhu, Feng Liang, Dagan Feng, and Dong Liang. "Accelerating Magnetic Resonance Rmaging via Deep Learning." In: *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*. 2016, pp. 514–517. DOI: 10.1109/ISBI.2016.7493320.

[306] Yinhuai Wang, Jiwen Yu, and Jian Zhang. "Zero-Shot Image Restoration Using Denoising Diffusion Null-Space Model." In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=mRieQgMtNTQ.

[307] Zhiwen Wang, Bowen Li, Wenjun Xia, Chenyu Shen, Mingzheng Hou, Hu Chen, Yan Liu, Jiliu Zhou, and Yi Zhang. "Leaders: Learnable Deep Radial Subsampling for MRI Reconstruction." In: *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*. 2022, pp. 1–5. DOI: 10.1109/ISBI52829.2022.9761544.

[308] Zhiwen Wang, Bowen Li, Hui Yu, Zhongzhou Zhang, Maosong Ran, Wenjun Xia, Ziyuan Yang, Jingfeng Lu, Hu Chen, Jiliu Zhou, Hongming Shan, and Yi Zhang. "Promoting Fast MR Imaging Pipeline by Full-Stack AI." In: *iScience* 27.1 (2024), p. 108608. ISSN: 2589-0042. DOI: 10.1016/j.isci.2023.108608. URL: https://www.sciencedirect.com/science/article/pii/S2589004223026858.

[309] Zhiwen Wang, Wenjun Xia, Zexin Lu, Yongqiang Huang, Yan Liu, Hu Chen, Jiliu Zhou, and Yi Zhang. "One Network to Solve Them All: A Sequential Multi-Task Joint Learning Network Framework for MR Imaging Pipeline." In: *Machine Learning for Medical Image Reconstruction: 4th International Workshop, MLMIR 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, October 1, 2021, Proceedings*. 2021, pp. 76–85. ISBN: 978-3-030-88551-9. DOI: 10.1007/978-3-030-88552-6_8. URL: https://doi.org/10.1007/978-3-030-88552-6_8.

[310] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. "Image Quality Assessment: From Error Visibility to Structural Similarity." In: *IEEE Transactions on Image Processing* 13.4 (2004), pp. 600–612. DOI: 10.1109/TIP.2003.819861.

[311] Tomer Weiss, Ortal Senouf, Sanketh Vedula, Oleg Michailovich, Michael Zibulevsky, and Alex Bronstein. "PILOT: Physics-Informed Learned Optimized Trajectories for Accelerated MRI." In: *Machine Learning for Biomedical Imaging* 1 (April 2021 issue 2021), pp. 1–23. ISSN: 2766-905X. DOI: 10.59275/j.melba.2021-1a1f. URL: https://melba-journal.org/2021:006.

[312] Tomer Weiss, Sanketh Vedula, Ortal Senouf, Oleg Michailovich, Michael Zibulevsky, and Alex Bronstein. "Joint Learning of Cartesian Undersampling and Reconstruction for Accelerated MRI." In: *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020, pp. 8653–8657. DOI: 10.1109/ICASSP40776.2020.9054542.

[313] Peter W. White and ECMWF. *IFS Documentation: Part III: Dynamics and Numerical Procedures (CY21R4)*. Meteorological bulletin. ECMWF, 2000. URL: https://books.google.com/books?id=yiXczAEACAAJ.

[314] David Wiesner, David Svoboda, Martin Maška, and Michal Kozubek. "CytoPacq: A Web-Interface for Simulating Multi-Dimensional Cell Imaging." In: *Bioinformatics* 35.21 (May 2019), pp. 4531–4533. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btz417. eprint: https://academic.

oup.com/bioinformatics/article-pdf/35/21/4531/50721742/bioinformatics_35_21_4531.pdf. URL: https://doi.org/10.1093/bioinformatics/btz417.

[315] Emil Wolf. "Three-Dimensional Structure Determination of Semi-Transparent Objects from Holographic Data." In: *Optics Communications* 1.4 (1969), pp. 153–156. ISSN: 0030-4018. DOI: https://doi.org/10.1016/0030-4018(69)90052-2. URL: https://www.sciencedirect.com/science/article/pii/0030401869900522.

[316] George N. Wong, Ben S. Prather, Vedant Dhruv, Benjamin R. Ryan, Monika Mościbrodzka, Chi-kwan Chan, Abhishek V. Joshi, Ricardo Yarza, Angelo Ricarte, Hotaka Shiokawa, Joshua C. Dolence, Scott C. Noble, Jonathan C. McKinney, and Charles F. Gammie. "PATOKA: Simulating Electromagnetic Observables of Black Hole Accretion." In: *The Astrophysical Journal. Supplement Series* 259.2 (Apr. 2022). DOI: 10.3847/1538-4365/ac582e.

[317] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. "Tune-A-Video: One-Shot Tuning of Image Diffusion Models for Text-to-Video Generation." In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 7589–7599. DOI: 10.1109/ICCV51070.2023.00701.

[318] Luhuan Wu, Brian Trippe, Christian Naesseth, David Blei, and John P Cunningham. "Practical and Asymptotically Exact Conditional Sampling in Diffusion Models." In: *Advances in Neural Information Processing Systems*. Ed. by A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine. Vol. 36. Curran Associates, Inc., 2023, pp. 31372–31403. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/63e8bc7bbf1cfea36d1d1b6538aecce5-Paper-Conference.pdf.

[319] Pingyu Wu, Kai Zhu, Yu Liu, Liming Zhao, Wei Zhai, Yang Cao, and Zheng-Jun Zha. *Improved Video VAE for Latent Video Diffusion Model*. 2024. arXiv: 2411.06449 [cs.CV]. URL: https://arxiv.org/abs/2411.06449.

[320] Zihui Wu, Yu Sun, Yifan Chen, Bingliang Zhang, Yisong Yue, and Katherine Bouman. "Principled Probabilistic Imaging Using Diffusion Models as Plug-and-Play Priors." In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024. URL: https://openreview.net/forum?id=Xq9HQf7VNV.

[321] Zihui Wu, Yu Sun, Jiaming Liu, and Ulugbek Kamilov. "Online Regularization by Denoising with Applications to Phase Retrieval." In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. 2019, pp. 3887–3895. DOI: 10.1109/ICCVW.2019.00482.

[322] Zihui Wu, Yu Sun, Alex Matlock, Jiaming Liu, Lei Tian, and Ulugbek S. Kamilov. "SIMBA: Scalable Inversion in Optical Tomography Using Deep Denoising Priors." In: *IEEE Journal of Selected Topics in Signal Processing* 14.6 (2020), pp. 1163–1175. DOI: 10.1109/JSTSP.2020.2999820.

[323] Zihui Wu, Tianwei Yin, Yu Sun, Robert Frost, Andre van der Kouwe, Adrian V. Dalca, and Katherine L. Bouman. "Learning Task-Specific Strategies for Accelerated MRI." In: *IEEE Transactions on Computational Imaging* 10 (2024), pp. 1040–1054. DOI: 10.1109/TCI.2024.3410521.

[324] Bin Xia, Yulun Zhang, Shiyin Wang, Yitong Wang, Xinglong Wu, Yapeng Tian, Wenming Yang, and Luc Van Gool. "DiffIR: Efficient Diffusion Model for Image Restoration." In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 13049–13059. DOI: 10.1109/ICCV51070.2023.01204.

[325] Xiaojian Xu, Weijie Gan, Satya V. V. N. Kothapalli, Dmitriy A. Yablonskiy, and Ulugbek S. Kamilov. "CoRRECT: A Deep Unfolding Framework for Motion-Corrected Quantitative R2* Mapping." In: *Journal of Mathematical Imaging and Vision* 67.2 (Apr. 2025). ISSN: 0924-9907. DOI: 10.1007/s10851-025-01236-y. URL: https://doi.org/10.1007/s10851-025-01236-y.

[326] Xingyu Xu and Yuejie Chi. "Provably Robust Score-Based Diffusion Posterior Sampling for Plug-and-Play Image Reconstruction." In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024. URL: https://openreview.net/forum?id=SLnsoaY4u1.

[327] Shengke Xue, Zhaowei Cheng, Guangxu Han, Chaoliang Sun, Ke Fang, Yingchao Liu, Jian Cheng, Xinyu Jin, and Ruiliang Bai. "2D Probabilistic Undersampling Pattern Optimization for MR Image Reconstruction." In: *Medical Image Analysis* 77 (Jan. 2022), p. 102346. DOI: 10.1016/j.media.2021.102346.

[328] Burhaneddin Yaman, Seyed Amir Hossein Hosseini, Steen Moeller, Jutta Ellermann, Kâmil Uğurbil, and Mehmet Akçakaya. "Self-Supervised Learning of Physics-Guided reconstruction Neural Networks without Fully Sampled Reference Data." In: *Magnetic Resonance in Medicine* 84 (July 2020). DOI: 10.1002/mrm.28378.

[329] Guang Yang, Simiao Yu, Hao Dong, Greg Slabaugh, Pier Dragotti, Xujiong Ye, Fangde Liu, Simon Arridge, Jennifer Keegan, Yike Guo, and David Firmin. "DAGAN: Deep De-Aliasing Generative Adversarial Networks for Fast Compressed Sensing MRI Reconstruction." In: *IEEE Transactions on Medical Imaging* PP (Dec. 2017), pp. 1–1. DOI: 10.1109/TMI.2017.2785879.

[330] Junwei Yang, Xiao-Xin Li, Feihong Liu, Dong Nie, Pietro Lio, Haikun Qi, and Dinggang Shen. "Fast Multi-Contrast MRI Acquisition by Optimal Sampling of Information Complementary to Pre-acquired MRI Contrast." In: *IEEE Transactions on Medical Imaging* (2022), pp. 1–1. DOI: 10.1109/TMI.2022.3227262.

[331] Yan Yang, Jian Sun, Huibin Li, and Zongben Xu. "Deep ADMM-Net for Compressive Sensing MRI." In: *Advances in Neural Information Processing Systems*. Ed. by D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett. Vol. 29. Curran Associates, Inc., 2016. URL: https://proceedings.neurips.cc/paper_files/paper/2016/file/1679091c5a880faf6fb5e6087eb1b2dc-Paper.pdf.

[332] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan.Zhang, Weihan Wang, Yean Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. "CogVideoX: Text-to-Video Diffusion Models with An Expert Transformer." In: *The Thirteenth International Conference on Learning Representations*. 2025. URL: https://openreview.net/forum?id=LQzN6TRFg9.

[333] Horng-Tzer Yau. "Relative Rntropy and Hydrodynamics of Ginzburg-Landau Models." In: *Letters in Mathematical Physics* 22.1 (1991), pp. 63–80. DOI: 10.1007/BF00400379. URL: https://doi.org/10.1007/BF00400379.

[334] Chang-Han Yeh, Chin-Yang Lin, Zhixiang Wang, Chi-Wei Hsiao, Ting-Hsuan Chen, Hau-Shiang Shiu, and Yu-Lun Liu. *DiffIR2VR-Zero: Zero-Shot Video Restoration with Diffusion-based Image Restoration Models*. 2024. arXiv: 2407.01519 [cs.CV]. URL: https://arxiv.org/abs/2407.01519.

[335] Tianwei Yin*, Zihui Wu*, He Sun, Adrian V. Dalca, Yisong Yue, and Katherine L. Bouman. "End-to-End Sequential Sampling and Reconstruction for MRI." In: *Proceedings of Machine Learning for Health*. Vol. 158. Proceedings of Machine Learning Research. PMLR, Dec. 2021, pp. 261–281. URL: https://proceedings.mlr.press/v158/yin21a.html.

[336] Bo Yuan, Jiaojiao Fan, Jiaming Liang, Andre Wibisono, and Yongxin Chen. "On a Class of Gibbs Sampling over Networks." In: *Proceedings of Thirty Sixth Conference on Learning Theory*. Ed. by Gergely Neu and Lorenzo Rosasco. Vol. 195. Proceedings of Machine Learning Research. PMLR, July 2023, pp. 5754–5780. URL: https://proceedings.mlr.press/v195/yuan23a.html.

[337] Michaël Zamo and Philippe Naveau. "Estimation of the Continuous Ranked Probability Score with Limited Information and Applications to Ensemble Weather Forecasts." In: *Mathematical Geosciences* 50.2 (Feb. 2018), pp. 209–234. DOI: 10.1007/s11004-017-9709-7. URL: https://hal.science/hal-02976423.

[338] Jure Zbontar et al. “fastMRI: An Open Dataset and Benchmarks for Accelerated MRI.” In: 2018. arXiv: 1811.08839.

[339] Frederik Zernike. “The Concept of Degree of Coherence and Its Application to Optical Problems.” In: *Physica* 5.8 (1938), pp. 785–795. ISSN: 0031-8914. DOI: 10.1016/S0031-8914(38)80203-2. URL: https://www.sciencedirect.com/science/article/pii/S0031891438802032.

[340] Zhifang Zhan, Jian-Feng Cai, Di Guo, Yunsong Liu, Zhong Chen, and Xiaobo Qu. “Fast Multiclass Dictionaries Learning with Geometrical Directions in MRI Reconstruction.” In: *IEEE Transactions on Biomedical Engineering* 63.9 (2016), pp. 1850–1861. DOI: 10.1109/TBME.2015.2503756.

[341] Bingliang Zhang, Wenda Chu, Julius Berner, Chenlin Meng, Anima Anandkumar, and Yang Song. “Improving Diffusion Inverse Problem Solving with Decoupled Noise Annealing.” In: *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*. June 2025, pp. 20895–20905. URL: https://openaccess.thecvf.com/content/CVPR2025/html/Zhang_Improving_Diffusion_Inverse_Problem_Solving_with_Decoupled_Noise_Annealing_CVPR_2025_paper.html.

[342] Cheng Zhang, Judith Bütepage, Hedvig Kjellström, and Stephan Mandt. “Advances in Variational Inference.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.8 (2019), pp. 2008–2026. DOI: 10.1109/TPAMI.2018.2889774.

[343] Jian Zhang and Bernard Ghanem. “ISTA-Net: Interpretable Optimization-Inspired Deep Network for Image Compressive Sensing.” In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 1828–1837. DOI: 10.1109/CVPR.2018.00196.

[344] Jinwei Zhang, Hang Zhang, Alan Wang, Qihao Zhang, Mert Sabuncu, Pascal Spincemaille, Thanh D. Nguyen, and Yi Wang. “Extending LOUPE for K-Space Under-Sampling Pattern Optimization in Multi-Coil MRI.” In: *Machine Learning for Medical Image Reconstruction: Third International Workshop, MLMIR 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 8, 2020, Proceedings*. Lima, Peru: Springer-Verlag, 2020, pp. 91–101. ISBN: 978-3-030-61597-0. DOI: 10.1007/978-3-030-61598-7_9. URL: https://doi.org/10.1007/978-3-030-61598-7_9.

[345] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Gool, and Radu Timofte. “Plug-and-Play Image Restoration With Deep Denoiser Prior.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP (June 2021), pp. 1–1. DOI: 10.1109/TPAMI.2021.3088914.

[346] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. “Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising.” In: *IEEE Transactions on Image Processing* 26.7 (2017), pp. 3142–3155. DOI: 10.1109/TIP.2017.2662206.

[347] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. "Learning Deep CNN Denoiser Prior for Image Restoration." In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 2808–2817. DOI: 10.1109/CVPR.2017.300.

[348] Qinsheng Zhang and Yongxin Chen. "Fast Sampling of Diffusion Models with Exponential Integrator." In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=Loek7hfb46P.

[349] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric." In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 586–595. DOI: 10.1109/CVPR.2018.00068.

[350] Zhixing Zhang, Bichen Wu, Xiaoyan Wang, Yaqiao Luo, Luxin Zhang, Yinan Zhao, Peter Vajda, Dimitris Metaxas, and Licheng Yu. "AVID: Any-Length Video Inpainting with Diffusion Model." In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2024, pp. 7162–7172. DOI: 10.1109/CVPR52733.2024.00684.

[351] Zizhao Zhang, Adriana Romero, Matthew J. Muckley, Pascal Vincent, Lin Yang, and Michal Drozdzal. "Reducing Uncertainty in Undersampled MRI Reconstruction With Active Acquisition." In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 2049–2053. DOI: 10.1109/CVPR.2019.00215.

[352] Bingliang Zhang*, Zihui Wu*, Berthy T. Feng, Yang Song, Yisong Yue, and Katherine L. Bouman. *STeP: A Framework for Solving Scientific Video Inverse Problems with Spatiotemporal Diffusion Priors*. 2025. arXiv: 2504.07549 [cs.CV]. URL: https://arxiv.org/abs/2504.07549.

[353] Lin Zhao, Xiao Chen, Eric Z. Chen, Yikang Liu, Dinggang Shen, Terrence Chen, and Shanhui Sun. *JoJoNet: Joint-Contrast and Joint-Sampling-and-Reconstruction Network for Multi-Contrast MRI*. 2022. arXiv: 2210.12548 [eess.IV]. URL: https://arxiv.org/abs/2210.12548.

[354] Ruiyang Zhao, Burhaneddin Yaman, Yuxin Zhang, Russell Stewart, Austin Dixon, Florian Knoll, Zhengnan Huang, Yvonne Lui, Michael Hansen, and Matthew Lungren. "fastMRI+, Clinical Pathology Annotations for Knee and Brain Fully Sampled Magnetic Resonance Imaging Data." In: *Scientific Data* 9 (Apr. 2022), p. 152. DOI: 10.1038/s41597-022-01255-z.

[355] Guoan Zheng, Roarke Horstmeyer, and Changhuei Yang. "Wide-Field, High-Resolution Fourier Ptychographic Microscopy." In: *Nature Photonics* 7 (July 2013), pp. 739–745. DOI: 10.1038/nphoton.2013.187.

[356] Hongkai Zheng, Wenda Chu, Austin Wang, Nikola Borislavov Kovachki, Ricardo Baptista, and Yisong Yue. "Ensemble Kalman Diffusion Guidance: A Derivative-free Method for Inverse Problems." In: *Transactions*

on Machine Learning Research (2025). ISSN: 2835-8856. URL: https://openreview.net/forum?id=XPEEsKneKs.

[357] Hongkai Zheng*, Wenda Chu*, Bingliang Zhang*, Zihui Wu*, Austin Wang, Berthy Feng, Caifeng Zou, Yu Sun, Nikola Borislavov Kovachki, Zachary E Ross, Katherine Bouman, and Yisong Yue. "InverseBench: Benchmarking Plug-and-Play Diffusion Models for Scientific Inverse Problems." In: *The Thirteenth International Conference on Learning Representations*. 2025. URL: https://openreview.net/forum?id=U3PBITXNG6.

[358] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. *MagicVideo: Efficient Video Generation With Latent Diffusion Models*. 2023. arXiv: 2211.11018 [cs.CV]. URL: https://arxiv.org/abs/2211.11018.

[359] Kevin Zhou and Roarke Horstmeyer. "Diffraction Tomography with a Deep Image Prior." In: *Optics Express* 28 (Apr. 2020), pp. 12872–12896. DOI: 10.1364/OE.379200.

[360] Bo Zhu, Jeremiah Zhe Liu, Bruce Rosen, and Matthew Rosen. "Image Reconstruction by Domain Transform Manifold Learning." In: *Nature* 555.7697 (Mar. 2018), pp. 487–492. DOI: 10.1038/nature25988.

[361] Yuanzhi Zhu, Kai Zhang, Jingyun Liang, Jiezhang Cao, Bihan Wen, Radu Timofte, and Luc Van Gool. "Denoising Diffusion Models for Plug-and-Play Image Restoration." In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2023, pp. 1219–1229. DOI: 10.1109/CVPRW59228.2023.00129.

[362] Marcelo Victor Wust Zibetti, Florian Knoll, and Ravinder R. Regatte. "Alternating Learning Approach for Variational Networks and Undersampling Pattern in Parallel MRI Applications." In: *IEEE Transactions on Computational Imaging* 8 (2022), pp. 449–461. DOI: 10.1109/TCI.2022.3176129.

[363] Jiaren Zou and Yue Cao. "Joint Optimization of k-t Sampling Pattern and Reconstruction of DCE MRI for Pharmacokinetic Parameter Estimation." In: *IEEE Transactions on Medical Imaging* 41.11 (2022), pp. 3320–3331. DOI: 10.1109/TMI.2022.3184261.

[364] Kelly Zou, Simon Warfield, Aditya Bharatha, Clare Tempany, Michael Kaus, Steven Haker, William Wells, Ferenc Jolesz, and Ron Kikinis. "Statistical Validation of Image Segmentation Quality Based on a Spatial Overlap Index." In: *Academic Radiology* 11 (Feb. 2004), pp. 178–89. DOI: 10.1016/S1076-6332(03)00671-8.

[365] Zihao Zou, Jiaming Liu, Shirin Shoushtari, Yubo Wang, and Ulugbek S. Kamilov. "FLAIR: A Conditional Diffusion Framework with Applications to Face Video Restoration." In: *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2025, pp. 5228–5238. DOI: 10.1109/WACV61041.2025.00511.