Schrödinger Bridge Methods for Image Reconstruction

Sunny Liu, Zach Salehe, J. Roberto Tello Ayala, Michael Xiang

Abstract

Image reconstruction tasks such as denoising and de-blurring involve recovering a clean image from a corrupted observation, which is a problem inherently ill-posed due to ambiguity and information loss. Schrödinger Bridge (SB) methods offer a promising approach by framing reconstruction as an entropy-regularized optimal transport problem. Instead of seeking a deterministic transport map, SB computes the most likely stochastic evolution between two probability distributions under a diffusion prior. In this paper, we seek to implement and study a four-term Schrödinger Bridge scheme that involves path matching, forward endpoint alignment, backward endpoint alignment, and cycle loss. We evaluate the method on the MNIST dataset, comparing it to other industry standard SB's, and we present promising findings that suggest that our four-term SB is competitive with other SB's in image reconstruction.

Code: https://github.com/zachsalehe/cs2840_project_team8

1. Introduction

Image reconstruction tasks, like denoising and de-blurring, have become fundamental problems in computer vision. But they remain challenging to solve due to being an *ill-posed* problem: multiple clean images can correspond to the same degraded observation. Classical approaches relied on reconstruction through a linear term ([Bertero and Boccacci, 1998]). However linear methods fail when the compression or corruption are not linear, such as in JPEG2000. Recently, diffusion models have achieved impressive results for image restoration by generating the image from random noise. However, for the task of image reconstruction, starting the generative process from random noise is not only computationally expensive but does not take into account the task in hand as corrupted inputs are not completely random noise.

By generalizing diffusion models through optimal transport (OT), the Schrödinger Bridge framework has allowed to describe random processes that evolve between two probability distributions. Thus allowing to circumvent starting from pure random noise by modeling the path from the *degraded* to the *clean* image, rather than between random noise and clean data. Although directly learning nonlinear diffusion bridges between degraded and clean distributions has proven successful to overcome the non-linearity of the classic formulations, in both paired (where one knows the true noise) and unpaired settings, there is an implicit assumption that mass must be preserved through solving the classical OT problem.

Mass preservation in the OT objective can be problematic in the image reconstruction task as many degradations destroy information or change support: cropping/inpainting/occlusion (pixels vanish), heavy JPEG or motion blur (high-freq content erased), down/up-sampling (aliasing/merging), unpaired domain shift (some poses/attributes exist in the degraded but not in clean image, and vice-versa). In probabilistic terms, parts of the degraded marginal p_Y can't be transported 1-to-1 into p_X, and parts of p_X have no preimage in p_Y, thus resulting in a mass mismatch. Balanced OT/SB enforces exact marginal conservation at endpoints. Unmatched regions get "smeared" to the nearest plausible places and promotes over-smoothing or artifacts. In this work we introduce a method that generalizes previous work by solving an unbalanced optimal transport to allow for more real transformations.

In this work, we introduce a four-term Schrödinger Bridge objective that addresses these limitations

through three key innovations. First, we employ *unbalanced* optimal transport at the endpoints, using Sinkhorn divergence with a finite mass-destruction parameter τ to permit genuine information loss and creation rather than forcing mass conservation. Second, we combine this with explicit path regularization through SB energy matching, ensuring physically plausible diffusion trajectories rather than endpoint-only shortcuts. Third, we add cycle consistency to promote reversible mappings without requiring paired supervision. Unlike UNSB, which relies on adversarial critics for marginal matching, our approach uses stable, differentiable metric losses. Our experiments on MNIST inpainting demonstrate that this combination produces reconstructions competitive with existing methods while offering a theoretically principled framework for handling support mismatch.

2. Related Work

Score-based generative models (SGMs) [Song et al., 2021], i.e., diffusion models, have become the gold-standard in recent years for various generative tasks, such as high-fidelity image synthesis. At a high level, SGM aims to learn a nonlinear function that transforms a sample distribution (typically Gaussian) into a complex, and often intractable, distribution (e.g., a large image dataset). In practice, this is achieved by first diffusing data to noise (approximately Gaussian) through a stochastic differential equation (SDE), then training a model using the score function to reverse this diffusion process by way of iterative denoising.

Early SGMs were not without their limitations: 1) the diffusion process had to obey a simple (e.g., linear) form in order to compute the analytic score function, and 2) a relatively large number of time steps were required such that the diffused data was approximately Gaussian. As a result, these models were particularly slow, limiting their early use compared to earlier generative models. To overcome such limitations, recent works [Bortoli et al., 2023, Chen et al., 2023] have taken inspiration from Schrodinger Bridge (SB), which, as an entropic optimal transport (OT) problem [Chen et al., 2014], seeks a pair of consistent forward and backward stochastic policies (or drifts) that transform between two arbitrary distributions over a finite time horizon. Following naturally from the similarities between the two problems, SB-inspired training of SGMs replaces the fixed forward diffusion process with a learnable stochastic bridge, which jointly optimizes the forward and reverse dynamics in order to satisfy the marginal constraints at the endpoints of the time horizon. This perspective recasts diffusion model training as an entropic OT problem, enabling more flexible and theoretically grounded learning objectives. In practice, SB-based methods can provide faster convergence, improved sample efficiency, and better control, allowing the model to adapt the diffusion process to the data, rather than relying on a hand-crafted noise schedule.

Building off of SB-inspired SGM, Image-to-Image Schrodinger Bridge (I^2SB) [Liu et al., 2023] applied such methods towards various image restoration tasks. Given the added flexibility of SB, I^2SB learns the diffusion bridges between corrupted and clean image distributions, without the need to iteratively add and remove noise. This is advantageous for image restoration, as corrupted images are far more structurally informative than pure noise.

Beyond I^2SB [Liu et al., 2023], other work have refined bridges for restoration and translation or extended the setting when we do not have aligned pairs of the same image, i.e., when we know y_i is the noisy observation and x_i is the corresponding clean target. Implicit I^2SB (I^3SB) [Sun et al., 2024] replaces the Markovian bridge with a non-Markovian scheme that conditions each step on the input corruption, matching I^2SB marginals while cutting neural function evaluations (NFE) and latency. In parallel, unpaired Image-to-Image translation via Neural SB (UNSB) shows that bridges scale to high-resolution unpaired settings by enforcing marginal consistency without paired supervision [Pang et al., 2023].

While these methods have demonstrated strong empirical performance, each has theoretical or practical constraints. I^2SB and I^3SB assume balanced marginals, which can be problematic when degradation genuinely destroys information. UNSB addresses unpaired settings but requires adversarial training, which can be unstable and sample-inefficient. Our four-term objective synthesizes ideas from these works: we retain the path regularization of I^2SB/I^3SB but relax endpoint constraints to unbalanced OT, and we incorporate cycle consistency like UNSB but replace adversarial matching with metric-based Sinkhorn divergence, yielding a stable, non-adversarial approach to unpaired reconstruction with support mismatch.

3. Method

Our goal is to learn a pair of stochastic processes that map between degraded images p_Y and clean images p_X using a four-term Schrödinger Bridge (SB) objective. This section summarizes how the mathematical components of the objective are implemented in practice through a unified masked SDE simulator and lightweight drift networks. For further details regarding our optimal transport objective, please refer to the Appendix.

3.1 Bridge Formulation

We model the forward process (from degraded images y to clean images x) and the backward process (from x back to y) as stochastic evolutions over time:

$$\mathrm{d}X_t = f_\theta(X_t, t; \mathrm{cond})\,\mathrm{d}t + \sigma\,\mathrm{d}W_t,$$

where `cond` is the original image we must preserve outside the masked region. In practice, this means the model learns a drift field that gradually adjusts the corrupted pixels toward a more plausible reconstruction while leaving the uncorrupted pixels unchanged.

Each drift is implemented using a small U-Net that takes:

$$(x_t,\ \mathrm{cond},\ \mathrm{mask},\ t)$$

as input.

Here, x_t denotes the current image state, `mask` specifies which pixels are missing or corrupted, and t encodes the time step. The network predicts a drift field that updates only the masked region, leaving the unmasked pixels unchanged.

Each step first produces a raw stochastic update

$$X_{t+1}^{\mathrm{raw}} = X_t + f_\theta(X_t, t)\,\Delta t + \sigma\sqrt{\Delta t}\,\xi_t,$$

after which we explicitly restore the known (unmasked) pixels:

$$X_{t+1} = X_{t+1}^{\mathrm{raw}} \odot \mathrm{mask}\ +\ \mathrm{cond} \odot (1 - \mathrm{mask}).$$

This overwrite step ensures that the model never modifies the correct (unmasked) pixels, forcing all learning to focus solely on reconstructing the corrupted region.

3.2 Masked SDE Simulation

Both the forward and backward processes are simulated using a shared function, `em_path()`, which updates the image over a sequence of small time steps. At each step, the current image X_t is nudged in two ways:

- by the learned drift $f_\theta(X_t, t)$, which predicts how the masked pixels should move toward a cleaner version, and
- by optional "reference" dynamics $g(X_t)$, which can either be zero (no extra force) or a gentle pull toward the origin.

By default, our four-term SB uses a Markovian update, meaning the drift depends only on the current state X_t. For I^{3}SB, the implementation can switch to a non-Markovian mode where the drift also sees the original input at every step; this is controlled through a configuration flag rather than a different code path.

3.3 Path Term (Energy Matching)

The SB objective contains the KL divergence between the learned bridge and a reference diffusion. This reduces to the drift energy integral

$$\mathcal{L}_{\text{path}} \approx \sum_{i=1}^{N} \|f_\theta(X_{t_i}, t_i)\|^2 \, \Delta t, \tag{1}$$

which appears in code as `drift.pow(2).mean() * dt` accumulated during each EM step. This term stabilizes training by discouraging irregular drift trajectories, which prevents the model from satisfying endpoints through unrealistic shortcuts.

3.4 Endpoint Alignment via Unbalanced OT

The forward terminal marginal $\widehat{\mu}_\phi^{(1)}$ must match p_X, and the backward initial marginal $\widehat{\nu}_\theta^{(0)}$ must match p_Y. Instead of computing Sinkhorn couplings, we implement an efficient unbalanced OT surrogate:

$$S_{\text{UOT}}(A, B) = \tfrac{1}{2}\|A - B\|_2^2 + \tfrac{1}{2}\,\tau\,\|A - B\|_1, \tag{2}$$

evaluated only inside the mask. The L_1 component (scaled by τ) allows mass creation/removal and thus handles reconstruction scenarios with genuine support mismatch (cropping, erasure, aliasing). Forward and backward endpoint penalties are simply:

$$\lambda_X \, S_{\text{UOT}}(X_1^f, x), \qquad \lambda_Y \, S_{\text{UOT}}(X_0^b, y).$$

3.5 Cycle Consistency

To discourage many-to-one mappings and promote a reversible image transformation, we add a cycle loss

$$\mathcal{L}_{\text{cycle}} = \|\,\tilde{y}_{\phi,\theta}(y) - y\|_1, \quad \tilde{y}_{\phi,\theta}(y) = X_0^b\big(X_1^f(y)\big).$$

The round trip uses the same masked SDE integrator, ensuring consistency between forward and backward drifts without paired supervision.

3.6 Training Loop

Each iteration performs two updates.

Forward update:

$$\mathcal{L}_f = \mathcal{L}^{(f)}_{\text{path}} + \lambda_X \, S_{\text{UOT}}(X^f_1, x).$$

Backward update:

$$\mathcal{L}_b = \mathcal{L}^{(b)}_{\text{path}} + \lambda_Y \, S_{\text{UOT}}(X^b_0, y) + \lambda_{\text{cyc}} \, \mathcal{L}_{\text{cycle}}.$$

The forward and backward drifts are optimized independently with Adam. Because all components (drift, energy, endpoint penalties, and cycle) share the same masked SDE mechanism, switching between I^{2}SB, I^{3}SB, UNSB, or our four-term SB is achieved simply by adjusting configuration flags (balanced vs. unbalanced OT, OU vs. zero reference, Markov vs. non-Markov, enabling cycle loss).

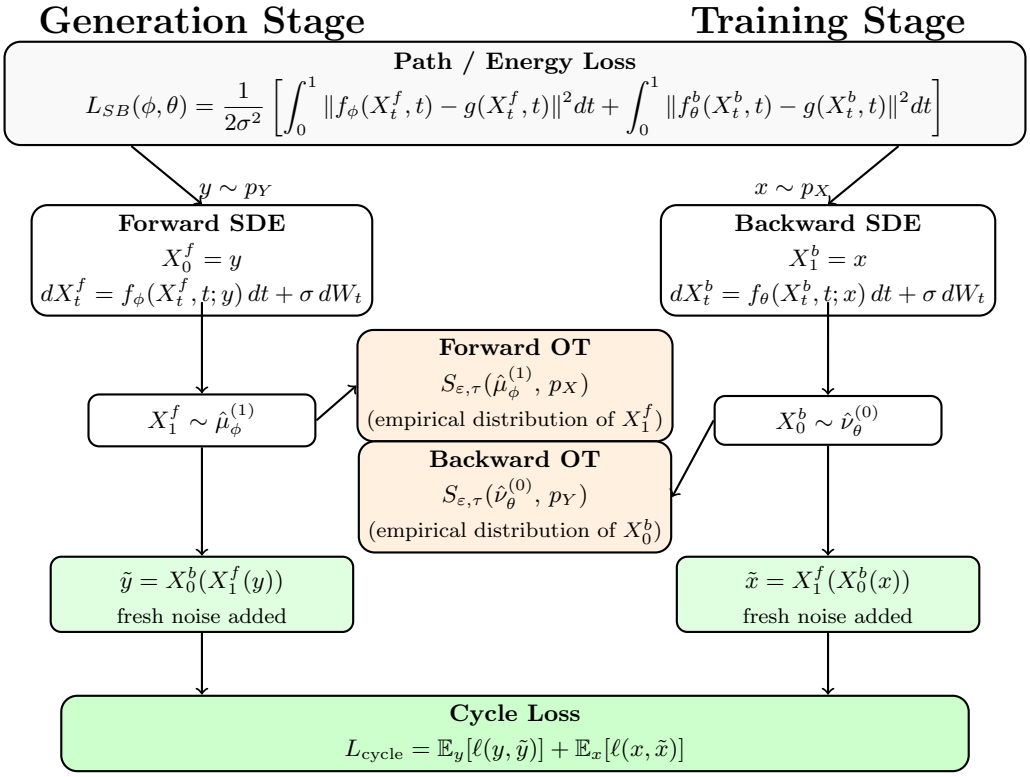

Figure 1: Four-term Schrödinger Bridge training pipeline with forward SDE, backward SDE, unbalanced optimal transport alignment at endpoints, and cycle consistency.

4. Experiments

In our experiments, we ultimately decided to focus on the task of image inpainting, as we found this to be the most difficult, as well as the most insightful, of the image restoration tasks covered in previous works. As baselines, we compared our unbalanced four-term model against I^{2}SB [Liu et al., 2023], I^{3}SB [Sun et al., 2024], and UNSB [Pang et al., 2023]. Given lack of time and compute resources, we chose to experiment in a toy setting, using smaller models and simpler datasets.

4.1 Dataset & Masking

Given our toy setting, we restricted our experiments to the MNIST handwritten digits dataset. We experimented in the case where we train and test on all digits in the dataset, as well as the case where we trained on $\{1, 2, 3, 4, 5, 7\}$ and tested on $\{0, 6, 8, 9\}$.

We constructed masks by thresholding randomly generated 2D Perlin noise, as was used in *Region-Aware Diffusion Models for Image Inpainting* [Kim et al., 2024]. We experimented in cases where the generated masks took up roughly 20% and 50% of the image area. We experimented with static center-crop masks as well.

4.2 Implementation Details

Given the lack of time needed to finetune the hyperparameters of each baseline, we decided to keep hyperparameters consistent across all models, wherever applicable, to ensure fairness. In our toy setting, we opted for a tiny U-Net backbone ($32 \to 64 \to 128 \to 64 \to 32$ features), where each model has one U-Net for forward drifts and one for backward drifts. The masked regions were replaced with zero-mean Gaussian noise with standard deviation $\sigma = 0.3$, and were evaluated at 30 timesteps along each diffusion process. The learning rate for both networks was 2×10^{-4}, and parameters were optimized using the Adam optimizer. When computing OT losses, we computed 80 Sinkhorn iterations using $\varepsilon = 0.2$ and $\tau = 0.6$. Where applicable, we used $\lambda_X = \lambda_Y = 10$, as well as $\lambda_{\text{cyc}} = 0.5$. Lastly, each model was trained for 5 epochs using a batch size of 64, where training each model took between 40 and 50 minutes on an M3 Max Macbook Pro (on MPS). For further information regarding reproducibility, please refer to the Appendix.

4.3 Metrics

In prior work [Liu et al., 2023, Sun et al., 2024, Pang et al., 2023], assessing inpainting reconstructions using FID score was commonplace. This made sense since, given the many-to-one mappings that can exist when masking natural images, computing an image quality score was more meaningful than a pairwise reconstruction score. Given that FID is trained on, and best suited for, natural images, we decided not to follow this trend, as it was not suited for a dataset like MNIST to begin with. We instead evaluate using SSIM and LPIPS scores, which compute pairwise scores between clean and reconstructed images. Since achieving per-pixel level accuracy is not the main goal of inpainting, we chose to avoid pixel-wise distances such as PSNR. Rather, SSIM assesses structural similarity and LPIPS assesses perceptual similarity, which we felt were more reasonable given our task and toy setting.

5. Results & Analysis

We evaluate our proposed four-term SB model under several masking and generalization settings on the MNIST dataset. We compare our method against the three baselines: I^2SB [Liu et al., 2023], I^3SB [Sun et al., 2024], and UNSB [Pang et al., 2023]. Across all experiments, we report structural similarity (SSIM) and perceptual similarity (LPIPS) scores, where higher SSIM and lower LPIPS indicate better reconstruction quality. We additionally provide qualitative comparisons of reconstructed digits under different corruption levels (figures for which are in the Appendix).

5.1 20% Perlin Masking

We first evaluate models trained and tested on all digits using 20% Perlin masking. As shown in Figure 2a, our model performs competitively with the best baseline. Quantitatively, our four-term SB achieves SSIM $= 0.967$ and LPIPS $= 0.033$, closely matching UNSB and outperforming the balanced SB variants. The qualitative reconstructions in Figure 3 show that all SB models can recover the overall shape, but I^3SB tends to introduce high-frequency artifacts, while our method produces smoother and more faithful completions, competitive with the I^2SB and UNSB results. We then examine a generalization setting in which models are trained on digits 1,2,3,4,5,7 but evaluated on unseen digits 0,6,8,9. As shown in

Method	SSIM ↑	LPIPS ↓
I^2SB	0.966	0.035
I^3SB	0.863	0.105
UNSB	**0.967**	**0.032**
Four-Term (Ours)	**0.967**	0.033

(a)

Method	SSIM ↑	LPIPS ↓
I^2SB	0.950	0.044
I^3SB	0.871	0.097
UNSB	**0.951**	0.041
Four-Term (Ours)	0.949	**0.039**

(b)

Figure 2: (a) Test evaluations when using 20% area Perlin masking, when trained and tested on all digits. (b) Test evaluations when using 20% area Perlin masking, when trained on $\{1, 2, 3, 4, 5, 7\}$ and tested on $\{0, 6, 8, 9\}$.

Figure 2b, all methods degrade due to the covariate shift, but our method remains competitive, achieving the lowest LPIPS of 0.039. The reconstructions in Figure 4 highlight that I^2SB becomes overly smooth and I^3SB suffers from noise amplification, while our model maintains shape consistency even when large portions of the image are occluded. This performance is competitive with the results of UNSB.

5.2 50% Perlin Masking

We repeat the above experiment but increase the corruption to 50% Perlin masking. This significantly increases difficulty because half the pixels are removed in complex spatial patterns.

In the in-distribution setting (Figure 5a), our method outperforms all baselines, with SSIM = 0.860 and LPIPS = 0.116. Notably, both balanced SB variants degrade substantially under high corruption, and I^3SB especially had high levels of noise (e.g., I^3SB LPIPS = 0.277). Figure 7 shows that under severe occlusion, I^2SB and UNSB tended to either hallucinate inconsistent strokes or fail to reconstruct digit topology, while our method was more accurate in reconstructing the digits with cleaner boundaries and reduced ghosting.

In the out-of-distribution evaluation (Figure 5b), reconstruction becomes even more challenging due to the combination of heavy masking and unseen digits. Nonetheless, our method again achieves the strongest perceptual performance (LPIPS = 0.137) while UNSB obtains marginally higher SSIM. However, Figure 8 shows similar visual results for UNSB and our method, whereas I^2SB and I^3SB suffer the same issues of ghost lines and high noise, respectively, as we have seen before.

5.3 Center-Crop Masking

Finally, we evaluate models on center-crop masking, which removes a contiguous central region rather than a random spatial pattern. This corruption forces the model to rely on global digit structure rather than local texture cues. Figure 6a shows that when trained and tested on all digits, our model achieves the best SSIM performance (SSIM = 0.877). UNSB achieves a marginally lower LPIPS but tended to produce blurrier digit interiors based on the qualitative samples, whereas our method retains sharper contours (Figure 9).

In the digit-generalization setting (Figure 6b), our method also maintains competitive SSIM and clearly outperforms the baselines on LPIPS. Qualitative samples demonstrate that I^2SB often over-smooths the filled-in region and create blurry lines, while I^3SB reintroduces noise and structural distortions. Our unbalanced four-term formulation appears to better propagate long-range structural information into the missing central region and maintain sharper edges, similar to the performance of UNSB (Figure 10).

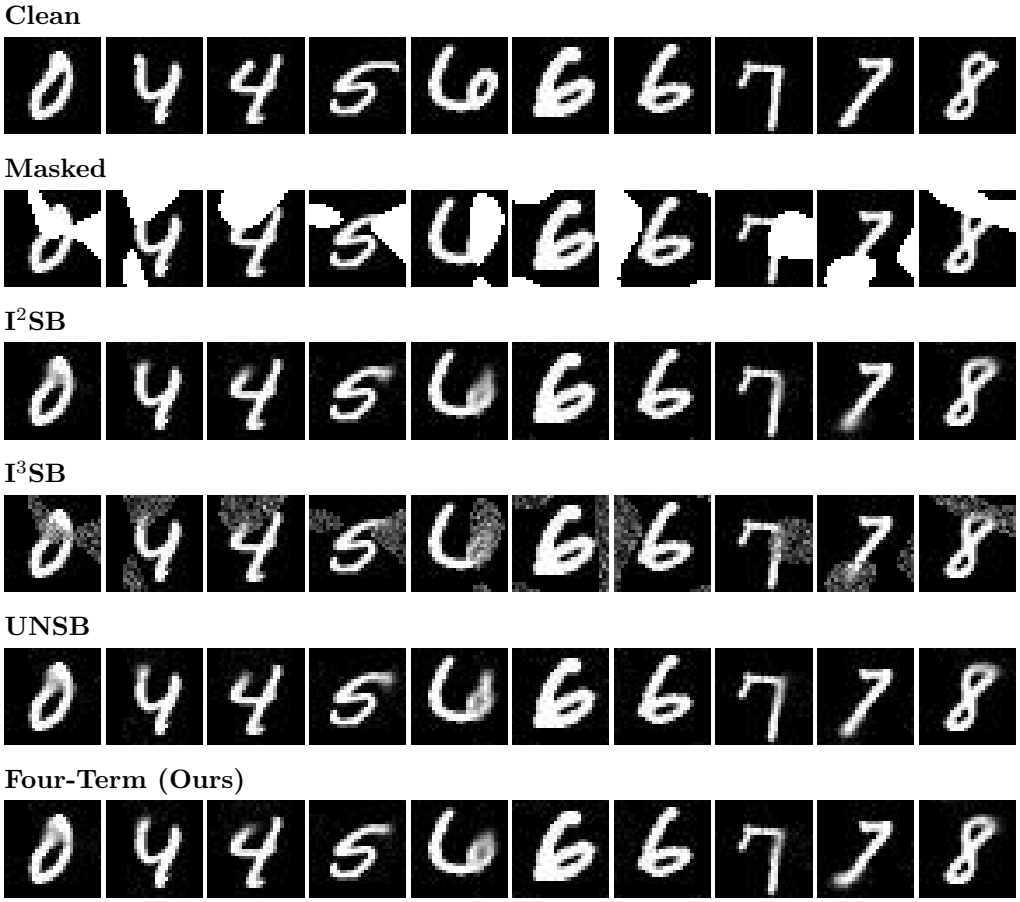

Figure 3: Reconstructed test examples when using 20% area Perlin masking, when trained and tested on all digits.

5.4 Summary of Trends

Across all experimental settings, several consistent trends emerge:

- Balanced SB methods degrade significantly with higher corruption. They frequently oversmooth or introduce noise, confirming that strict mass preservation is detrimental when large parts of the input are missing.
- UNSB provides a strong baseline, especially in LPIPS, but sometimes produces blurrier reconstructions relative to ours.
- Our unbalanced four-term SB model consistently achieves best or near-best performance compared to UNSB in nearly all settings, with especially strong perceptual quality under high corruption or domain shift.
- Generalization to unseen digits is challenging, but our method consistently preserves global structure that is competitive with all of the baselines.

Overall, the results indicate that relaxing mass preservation and incorporating our additional terms leads to more robust reconstruction behavior, particularly in scenarios with severe information loss or distributional mismatch.

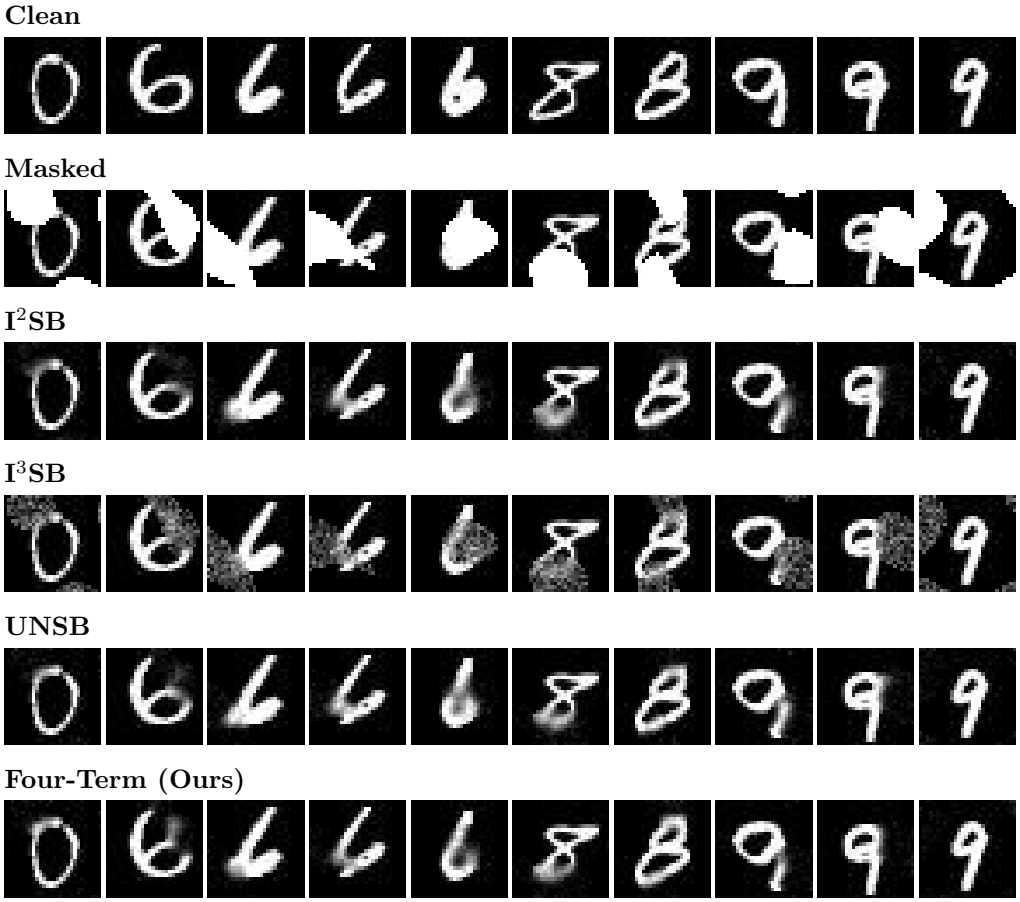

Figure 4: Reconstructed test examples when using 20% area Perlin masking, when trained on $\{1, 2, 3, 4, 5, 7\}$ and tested on $\{0, 6, 8, 9\}$.

6. Discussion

While we show that our method that our unbalanced four-term model can match and even outperform the current state-of-the-art, our work is not without its limitations. Due to a lack of time and computational resources, our experiments were confined to a toy setting using tiny models and a limit dataset of images. To provide a better assessment of our method and better baseline comparisons, one should train full-sized models on an extensive dataset of natural images, such as ImageNet. Additionally, due to our constraints, we were unable to fully finetune the hyperparameters of each model; further improvements in each model may be expected if this were the case.

Given that we confine ourselves to the MNIST dataset, we do not believe our work to pose any ethical or fairness issues. We do acknowledge, however, that larger image inpainting models which have been trained on large natural image datasets can be misused for content manipulation or provide unfair outputs. Additionally, such natural image datasets may raise further concerns regarding its retrieval, representation, etc.

7. Conclusion & Future Work

We investigated Schrödinger Bridge methods for image reconstruction on the MNIST dataset and developed a four-term SB that combines path/energy matching, forward endpoint alignment, backward endpoint alignment, and cycle loss. While our method offers theoretical advantages in handling support

mismatch, our experiments demonstrate that it produces competitive reconstructions on par with existing methods. The method provides a theoretically motivated, interpretable alternative that matches or exceeds the performance of existing Schrödinger Bridge approaches.

Several promising directions remain for future work. First, integrating stronger learned priors, such as modern score-based diffusion models, may significantly enhance reconstruction quality. Second, learned coefficients for the four SB terms could improve stability and convergence speed. Finally, we only trained on the MNIST dataset, so another possible future direction would involve testing our four-term SB on a larger dataset.

Overall, Schrödinger Bridges provide a flexible and theoretically grounded framework for inverse problems, and our findings suggest there remains good room for further advancing SB-based image reconstruction.

References

[Bertero and Boccacci, 1998] Bertero, M. and Boccacci, P. (1998). *Introduction to Inverse Problems in Imaging*. Taylor & Francis.

[Bortoli et al., 2023] Bortoli, V. D., Thornton, J., Heng, J., and Doucet, A. (2023). Diffusion schrödinger bridge with applications to score-based generative modeling.

[Chen et al., 2023] Chen, T., Liu, G.-H., and Theodorou, E. A. (2023). Likelihood training of schrödinger bridge using forward-backward sdes theory.

[Chen et al., 2014] Chen, Y., Georgiou, T. T., and Pavon, M. (2014). On the relation between optimal transport and schrödinger bridges: A stochastic control viewpoint. *Journal of Optimization Theory and Applications*, 169:671–691.

[Kim et al., 2024] Kim, S., Suh, S., and Lee, M. (2024). Rad: Region-aware diffusion models for image inpainting.

[Liu et al., 2023] Liu, G.-H., Vahdat, A., Huang, D.-A., Theodorou, E. A., Nie, W., and Anandkumar, A. (2023). I^2sb: Image-to-image schrödinger bridge.

[Pang et al., 2023] Pang, T., Wang, Y., Liu, Q., Lin, M., and Yan, S. (2023). Unpaired image-to-image translation via neural schrödinger bridge.

[Song et al., 2021] Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2021). Score-based generative modeling through stochastic differential equations.

[Sun et al., 2024] Sun, Y., Chen, Y., Gao, R., Song, J., and Ermon, S. (2024). Implicit image-to-image schrödinger bridge for image restoration.

A. Optimal Transport Objective

We train forward and backward Schrödinger bridges between a degraded domain p_Y and a clean domain p_X using the following *four-term* objective:

$$\mathcal{L}(\phi,\theta) = \underbrace{\mathcal{L}_{\mathrm{SB}}(\phi,\theta)}_{\text{path/energy matching}} + \lambda_X\, S_{\varepsilon,\tau}\big(\hat{p}_\phi^{(1)},\, p_X\big) \;+\; \lambda_Y\, S_{\varepsilon,\tau}\big(\hat{p}_\theta^{(0)},\, p_Y\big) \;+\; \lambda_{\mathrm{cyc}}\,\mathcal{L}_{\mathrm{cycle}}. \tag{3}$$

Forward/Backward samplers and round-trip maps

Forward (degraded → clean). Given $y\sim p_Y$, simulate

$$\mathrm{d}X_t^f \;=\; f_\phi\big(X_t^f, t; y\big)\,\mathrm{d}t + \sigma\,\mathrm{d}W_t, \qquad X_0^f = y, \tag{4}$$

and denote the terminal (time-1) *marginal distribution* of the forward process by

$$\widehat{\mu}_\phi^{(1)}(A) := \mathbb{P}\big(X_1^f \in A\big), \qquad \forall A \in \mathcal{B}(\mathbb{R}^d).$$

Backward (clean → degraded). Given $x \sim p_X$, simulate (in reversed clock or via an equivalent forward SDE)

$$\mathrm{d}X_t^b \;=\; f_\theta^b\big(X_t^b, t; x\big)\,\mathrm{d}t + \sigma\,\mathrm{d}W_t, \qquad X_1^b = x, \tag{5}$$

and denote the initial (time-0) *marginal distribution* of the backward process by

$$\widehat{\nu}_\theta^{(0)}(A) := \mathbb{P}\big(X_0^b \in A\big), \qquad \forall A \in \mathcal{B}(\mathbb{R}^d).$$

Round-trip (cycle) maps. Using fresh noise in each leg,

$$\tilde{y}_{\phi,\theta}(y) := X_0^b\big(X_1^f(y)\big), \qquad \tilde{x}_{\phi,\theta}(x) := X_1^f\big(X_0^b(x)\big). \tag{6}$$

No paired supervision is required: expectations are taken over $y \sim p_Y$ and $x \sim p_X$ independently.

Explanation of each term in (3)

(1) Path/energy matching $\mathcal{L}_{\mathrm{SB}}(\phi,\theta)$. Let R be a reference diffusion with drift $g(x,t)$ and diffusion scale $\sigma > 0$. By Girsanov,

$$\mathrm{KL}(P\|R) = \frac{1}{2\sigma^2}\,\mathbb{E}_P \int_0^1 \big\|b_t(X_t) - g(X_t,t)\big\|_2^2\,\mathrm{d}t.$$

We use its discretized, symmetric surrogate to train both drifts:

$$\mathcal{L}_{\mathrm{SB}}(\phi,\theta) \;=\; \frac{1}{2\sigma^2}\,\mathbb{E}\bigg[\int_0^1 \big\|f_\phi(X_t^f,t;y) - g(X_t^f,t)\big\|_2^2\,\mathrm{d}t + \int_0^1 \big\|f_\theta^b(X_t^b,t;x) - g(X_t^b,t)\big\|_2^2\,\mathrm{d}t\bigg]. \tag{7}$$

This anchors the *entire path* to physically consistent diffusion dynamics, preventing endpoint-only shortcuts.

(2) Forward endpoint alignment $S_{\varepsilon,\tau}\big(\hat{p}_\phi^{(1)}, p_X\big)$. We align the forward terminal *marginal distribution* $\widehat{\mu}_\phi^{(1)}$ to the clean data distribution p_X via the *unbalanced* Sinkhorn divergence $S_{\varepsilon,\tau}$, which softly allows

mass change (creation/annihilation) through $\tau > 0$:

$$S_{\varepsilon,\tau}(\mu,\nu) := \mathrm{UOT}_{\varepsilon,\tau}(\mu,\nu) - \tfrac{1}{2}\,\mathrm{UOT}_{\varepsilon,\tau}(\mu,\mu) - \tfrac{1}{2}\,\mathrm{UOT}_{\varepsilon,\tau}(\nu,\nu), \tag{8}$$

$$\mathrm{UOT}_{\varepsilon,\tau}(\mu,\nu) := \inf_{\gamma\geq 0} \langle C,\gamma\rangle + \varepsilon\,\mathrm{KL}\big(\gamma \parallel \mu\otimes\nu\big) + \tau\Big[\mathrm{KL}\big(\gamma\mathbf{1} \parallel \mu\big) + \mathrm{KL}\big(\gamma^\top\mathbf{1} \parallel \nu\big)\Big], \tag{9}$$

where C is a ground cost (e.g., squared Euclidean or a perceptual/feature cost), γ is a soft coupling, and $\gamma\mathbf{1}, \gamma^\top\mathbf{1}$ are its marginals.

(3) Backward endpoint alignment $S_{\varepsilon,\tau}\big(\hat{p}_\theta^{(0)}, p_Y\big)$. The mirror image of (2): we tie the backward initial *marginal distribution* $\widehat{\nu}_\theta^{(0)}$ to the degraded data distribution p_Y via $S_{\varepsilon,\tau}\big(\widehat{\nu}_\theta^{(0)},\, p_Y\big)$.

(4) Cycle loss $\mathcal{L}_{\mathrm{cycle}}$. A sample-wise self-consistency term that discourages many-to-one collapse and promotes a quasi-invertible mapping:

$$\mathcal{L}_{\mathrm{cycle}} = \mathbb{E}_{y\sim p_Y}\,\ell\big(y, \tilde{y}_{\phi,\theta}(y)\big) + \mathbb{E}_{x\sim p_X}\,\ell\big(x, \tilde{x}_{\phi,\theta}(x)\big), \tag{10}$$

with ℓ chosen as, e.g., ℓ_1 plus a perceptual loss. This works in both paired and unpaired regimes.

Remarks (limits). As $\tau\to\infty$ the endpoint penalties become *balanced* (mass-preserving), recovering the classical SB/I^{2}SB assumption. Setting λ_{cyc}=0 removes the sample-wise invertibility pressure.

B. Reproducibility

Our project's codebase can be found here: https://github.com/zachsalehe/cs2840_project_team8. We did not use random seeds when training our models. We did, however, use random seeds when generating our Perlin masks, and we generate 10k masks when initializing the dataset class. This means that, even though images and masks were randomly chosen for each batch, the masks were chosen from identical pools of 10k masks. Given the simplicity of our dataset, we believe that final results would be near-identical across different runs. Beyond common Python libraries such as `numpy` and `torch`, the only non-common library we imported in our code was `lpips`, used for computing LPIPS scores during our model evaluations.

C. Additional testing cases

Method	SSIM ↑	LPIPS ↓
I^{2}SB	0.850	0.128
I^{3}SB	0.623	0.277
UNSB	0.846	0.118
Four-Term (Ours)	**0.860**	**0.116**

(a)

Method	SSIM ↑	LPIPS ↓
I^{2}SB	**0.817**	0.152
I^{3}SB	0.634	0.269
UNSB	0.816	0.141
Four-Term (Ours)	0.803	**0.137**

(b)

Figure 5: (a) Test evaluations when using 50% area Perlin masking, when trained and tested on all digits. (b) Test evaluations when using 50% area Perlin masking, when trained on $\{1,2,3,4,5,7\}$ and tested on $\{0,6,8,9\}$.

Method	SSIM ↑	LPIPS ↓
I^2SB	0.868	0.116
I^3SB	0.806	0.194
UNSB	0.869	**0.098**
Four-Term (Ours)	**0.877**	0.100

(a)

Method	SSIM ↑	LPIPS ↓
I^2SB	**0.834**	0.139
I^3SB	0.794	0.189
UNSB	0.828	0.140
Four-Term (Ours)	0.806	**0.125**

(b)

Figure 6: (a) Test evaluations when using center-crop masking, when trained and tested on all digits. (b) Test evaluations when using center-crop masking, when trained on $\{1, 2, 3, 4, 5, 7\}$ and tested on $\{0, 6, 8, 9\}$.

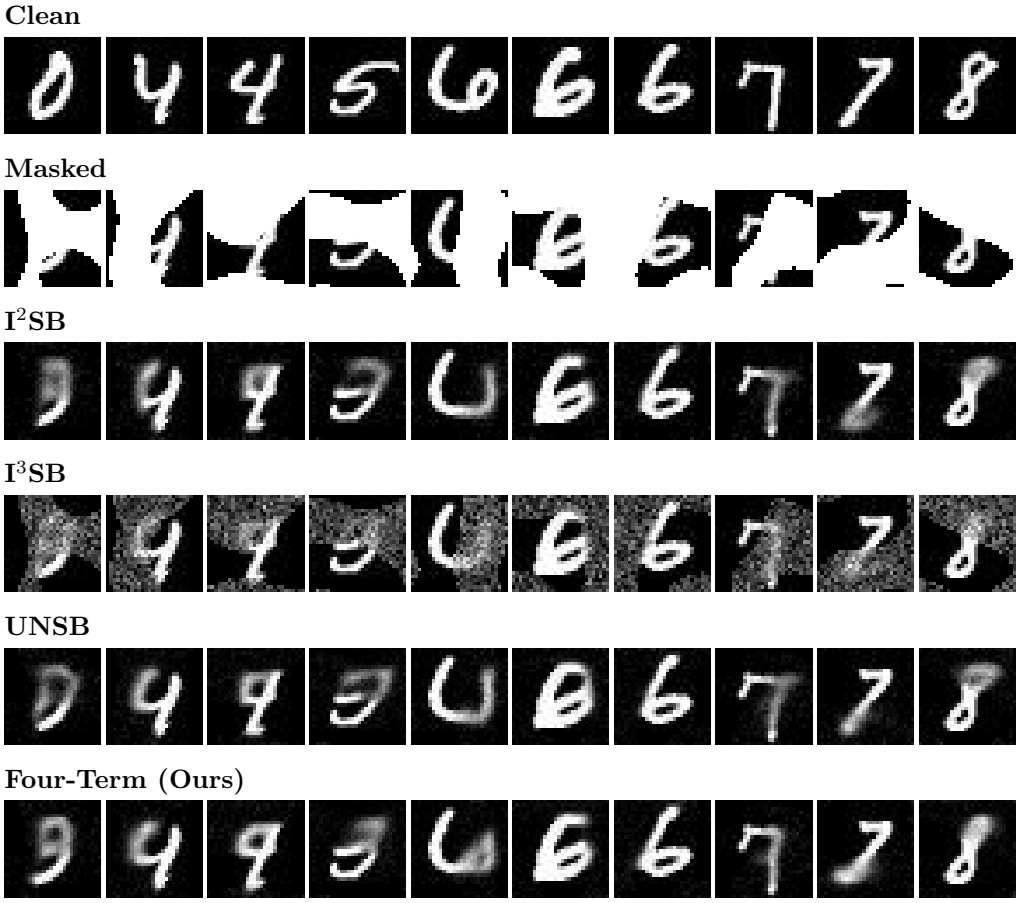

Figure 7: Reconstructed test examples when using 50% area Perlin masking, when trained and tested on all digits.

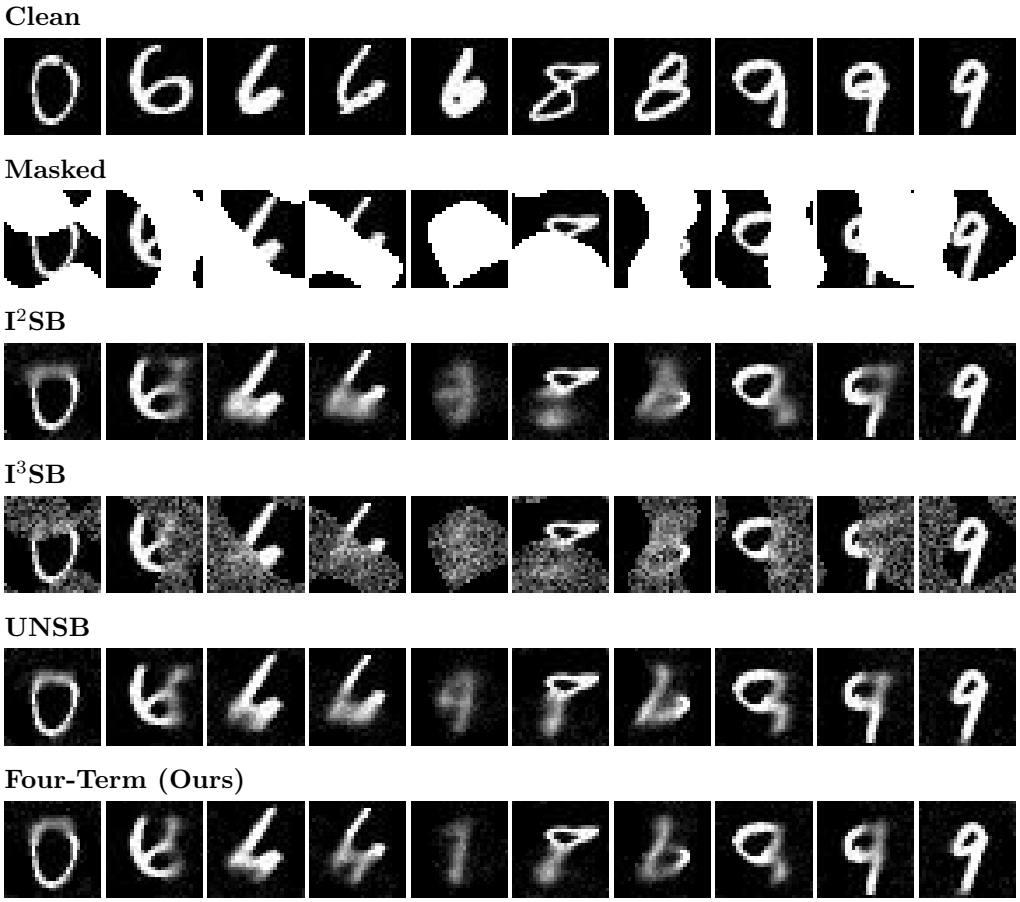

Figure 8: Reconstructed test examples when using 50% area Perlin masking, when trained on $\{1, 2, 3, 4, 5, 7\}$ and tested on $\{0, 6, 8, 9\}$.

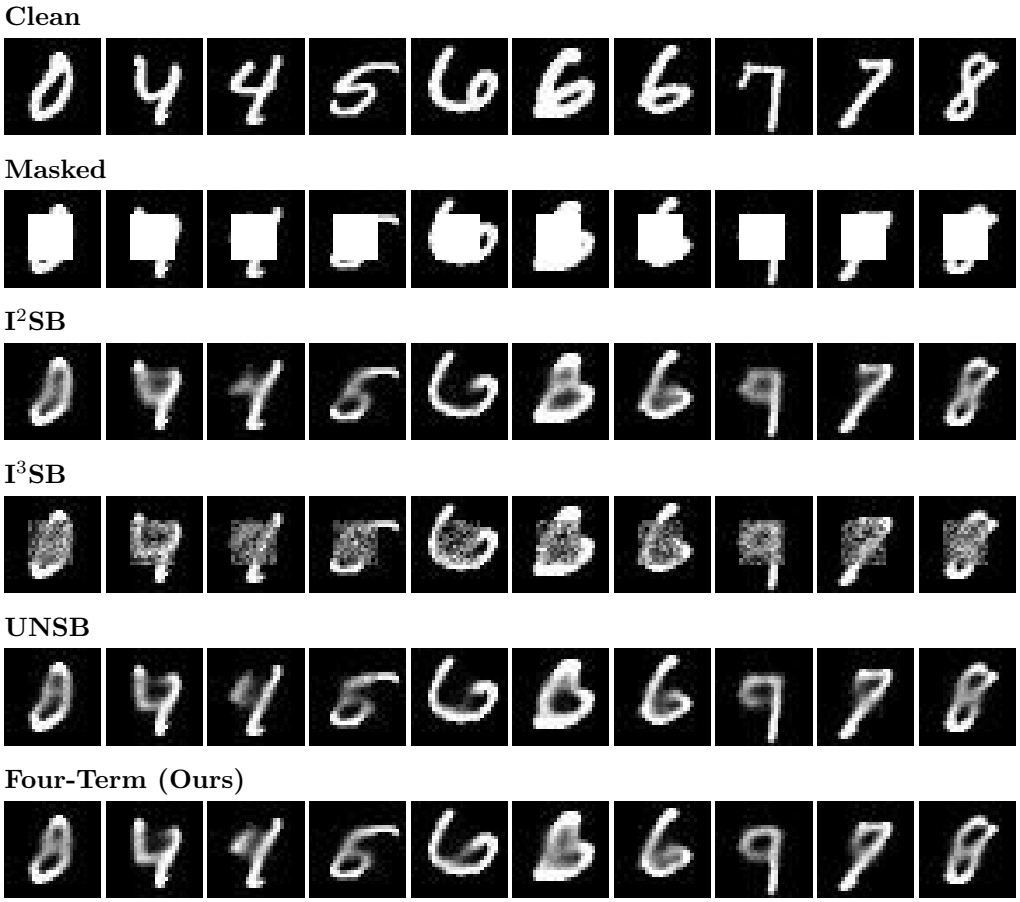

Figure 9: Reconstructed test examples when using center-crop masking, when trained and tested on all digits.

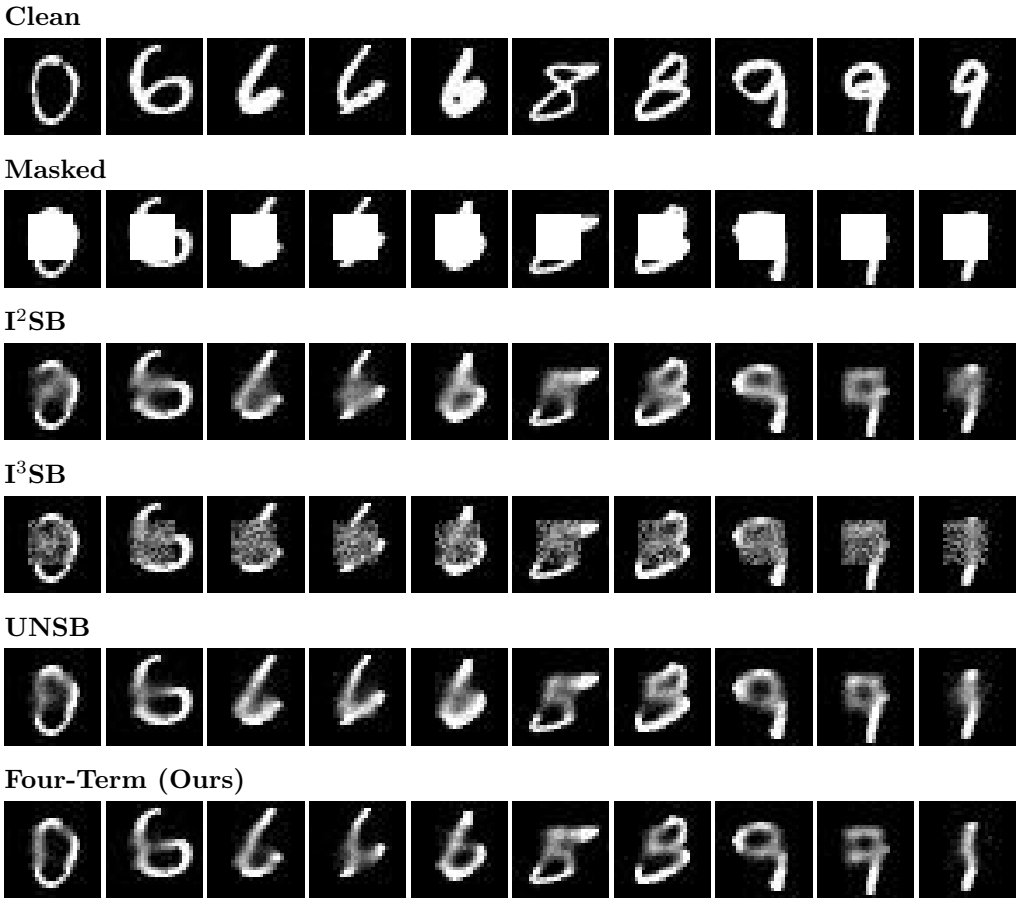

Figure 10: Reconstructed test examples when using center-crop masking, when trained on $\{1, 2, 3, 4, 5, 7\}$ and tested on $\{0, 6, 8, 9\}$.