

NullFlow: One-Step Generative Reconstruction

Xiao Shi[†]  Edward P. Chandler[†]  Chicago Y. Park 
Shirin Shoushtari  Ulugbek S. Kamilov 

University of Wisconsin–Madison and Washington University in St. Louis
{xiao.shi, epchandler, chicago.park, kamilov}@wisc.edu s.shirin@wustl.edu

[†]Equal contribution.

Abstract

We propose NullFlow, a principled framework for *one-step* generative image reconstruction. Our key idea is to confine the generative flow to a measurement-consistent subspace. Because the flow never leaves this subspace, NullFlow needs no separate data-fidelity corrections, unlike existing solvers. NullFlow samples in a single network evaluation by learning the flow’s *average* velocity, avoiding the step-by-step integration of the traditional flow matching methods. We prove that the average velocity of this constrained flow yields a training objective whose global minimizer is a one-step posterior sampler. We show on image inpainting that NullFlow matches state-of-the-art diffusion solvers while cutting inference from hundreds of network evaluations to one.

Index Terms— Imaging inverse problems, image reconstruction, generative priors, null-space learning, mean flows

1 Introduction

Imaging inverse problems seek to recover an unknown image $\mathbf{x} \in \mathbb{R}^n$ from measurements $\mathbf{y} \in \mathbb{R}^m$. For linear systems this is modeled as

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}, \quad (1)$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ is a known forward operator and $\mathbf{n}$ is noise. We focus on the ill-posed setting where $\text{null}(\mathbf{A}) \neq \{\mathbf{0}\}$, which covers image inpainting, accelerated MRI, and sparse-view CT. In this setting, the measurements constrain only the row-space of $\mathbf{x}$; the null-space component is invisible to $\mathbf{A}$ and must be supplied *entirely* by prior knowledge. Reconstruction is therefore not inversion of $\mathbf{A}$, but generation of plausible null-space content consistent with the measurements.

There is growing interest in learning-based methods for supplying this missing content. Early work predicted $\mathbf{x}$ directly from $\mathbf{y}$ [1–5]. Recent methods instead model the distribution of plausible images, using denoisers [6–8], diffusion [9, 10] and flow [11–14] models as priors that, plugged into inverse solvers, reach state-of-the-art reconstruction across many problems [15–18] (see [19, 20] for surveys).

Existing generative solvers are expensive due to their iterative nature. They reconstruct images by simulating a full generative trajectory, alternating prior updates from the model with data-fidelity updates that enforce measurement consistency, and reach a solution only after hundreds or thousands of network evaluations [21–28]. Mean Flows [29–31] are a recent class of methods that remove this cost for unconditional and class-conditional generation. Rather than learning the instantaneous velocity of a flow and integrating it step by step, a Mean Flow learns the *average* velocity between any two time points directly, collapsing the entire trajectory into a single network evaluation. To date, however, this idea has not been applied to imaging inverse problems.

In this letter, we propose NullFlow, which brings the Mean Flow into the null space of the forward operator, thus enabling a one-step posterior sampler. We make three contributions:

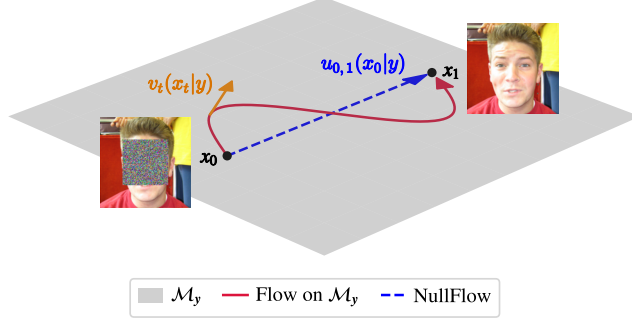


Figure 1: NullFlow is able to reconstruct in a single step without leaving the measurement-consistent subspace $\mathcal{M}_y := \{x : Ax = y\}$ by learning the average velocity $u_{0,1}(x_0|y)$ rather than the instantaneous velocity $v_t(x_t|y)$ of a flow confined to $\mathcal{M}_y$.

- *Method.* We propose the first one-step solver for imaging inverse problems based on mean flows. Confining the flow to $\text{null}(A)$ keeps every state measurement-consistent by construction, and learning its average velocity yields reconstruction in a single network evaluation.
- *Theory.* We prove that the average velocity of this constrained flow satisfies a MeanFlow identity restricted to the null space, yielding a training objective whose global minimizer is a one-step posterior sampler $x \sim p(x|y)$.
- *Experiments.* We show on image inpainting that one NullFlow sample gives the best LPIPS in a single step; averaging 100 samples approaches the MMSE estimate and exceeds the MSE-trained network in PSNR and SSIM, without retraining.

2 Proposed Method

Our method is motivated by the observation that any x can be decomposed into the row space projection and null space projection of an ill-posed A :

$$x = \underbrace{A^\dagger Ax}_{\text{row space projection}} + \underbrace{(I - A^\dagger A)x}_{\text{null space projection}},$$

where $A^\dagger$ is the pseudo-inverse of A . We write $P := I - A^\dagger A$ for the orthogonal projection onto $\text{null}(A)$, which we use throughout. In the noiseless measurement setting, the inverse problem can be viewed as reconstructing the null-space components, conditioned on the measurement y ; the set of all images that are measurement consistent is the affine subspace

$$\mathcal{M}_y := \{x : Ax = y\}. \quad (2)$$

Our objective is to learn a one-step Mean Flow, which we call NullFlow, that exists completely within $\mathcal{M}_y$ and samples from the conditional image distribution $p(x|y)$ (see Fig. 1).

With measurement noise, y no longer determines the row-space component exactly. One may instead anchor it at the least-squares estimate $A^\dagger y$ and reconstruct only the null-space component by defining the flow within $\{x' : Ax' = AA^\dagger y\}$. This is equivalent to drawing a posterior sample from the target image distribution $w \sim p(\cdot|y)$ and forming $x' = A^\dagger y + Pw$. For the remainder of the paper, we assume noiseless measurements, and use $\mathcal{M}_y$ given in (2).

2.1 Subspace-Restricted Flow Matching

We adapt the flow matching of [11] to the subspace $\mathcal{M}_y$, confining every object—path, flow, and velocity—to $\mathcal{M}_y$, whose tangent space is $\text{null}(A)$. This keeps every state measurement-consistent without a separate data-fidelity step.

We transport between two distributions supported on $\mathcal{M}_y$. The source is $x_0 \sim q(\cdot|y)$, with $x_0 = A^\dagger y + Pz$ and $z \sim \mathcal{N}(0, I)$, which pins the row-space component to the least-squares estimate

Algorithm 1 NullFlow: Training

Require: clean data $\mathbf{x}$, measurement $\mathbf{y}$, forward model $\mathbf{A}$

```

t, r = sample_t_r()
# Add noise only to manifold
e = randn(x.shape)
z = t * proj_null(e) + (1-t) * proj_null(x) + A_pinv(y)

# average velocity u from x-prediction
def u_fn(z, r, t):
    return (z - net(z, r, t)) / t

# instantaneous velocity v at time t
v = u_fn(z, t, t)

# predict u and dudt
u, dudt = jvp(u_fn, (z, r, t), (v, 0, 1))
V = u + (t - r) * stopgrad(dudt)

loss = metric(V, e_M - x)

```

Algorithm 2 NullFlow: One-Step Sampling

```

e_M = A_pinv(y) + proj_null(randn(x_shape))
x = net(e_M, r=0, t=1)

```

$\mathbf{A}^\dagger \mathbf{y}$ and fills the null space with Gaussian noise. The target is $\mathbf{x}_1 \sim p(\cdot | \mathbf{y})$, the conditional image distribution. We seek a probability path $p_t(\cdot | \mathbf{y})$ on $\mathcal{M}_{\mathbf{y}}$, with $p_0(\cdot | \mathbf{y}) = q(\cdot | \mathbf{y})$ and $p_1(\cdot | \mathbf{y}) \approx p(\cdot | \mathbf{y})$, that interpolates between them.

We take the conditional path toward a target $\mathbf{x}_1$ to be Gaussian with covariance confined to $\text{null}(\mathbf{A})$,

$$p_t(\mathbf{x} | \mathbf{x}_1, \mathbf{y}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_t(\mathbf{x}_1, \mathbf{y}), \sigma_t^2(\mathbf{x}_1) \mathbf{P}), \quad (3)$$

with time-dependent mean $\boldsymbol{\mu}_t$ and scale $\sigma_t > 0$ subject to the endpoint conditions $\boldsymbol{\mu}_0 = \mathbf{A}^\dagger \mathbf{y}$, $\sigma_0 = 1$ and $\boldsymbol{\mu}_1 = \mathbf{x}_1$, $\sigma_1 = \sigma_{\min} \ll 1$, so that $p_0 = q(\cdot | \mathbf{y})$ and p_1 concentrates at $\mathbf{x}_1$. Since $\mathbf{A} \boldsymbol{\mu}_t = \mathbf{y}$ and $\sigma_t^2 \mathbf{P}$ spreads mass only along $\text{null}(\mathbf{A})$, the path is supported on $\mathcal{M}_{\mathbf{y}}$ for all t .

To realize this path deterministically, we use the location-scale map of the Gaussian, the conditional flow

$$\psi_t(\mathbf{x} | \mathbf{y}) = \sigma_t(\mathbf{x}_1) \mathbf{P} \mathbf{x} + \boldsymbol{\mu}_t(\mathbf{x}_1, \mathbf{y}), \quad (4)$$

which scales the null-space content of $\mathbf{x}$ by σ_t and centers it at $\boldsymbol{\mu}_t$. Pushing a reference $\mathbf{P} \mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{P})$ through ψ_t reproduces (3) at every t .

The flow $\phi_t(\cdot | \mathbf{y}) : \mathcal{M}_{\mathbf{y}} \rightarrow \mathcal{M}_{\mathbf{y}}$ is generated by the velocity field $\mathbf{v}_t$, the time-dependent field whose integral curves are its trajectories, solving

$$\frac{d}{dt} \phi_t(\mathbf{x} | \mathbf{y}) = \mathbf{v}_t(\phi_t(\mathbf{x} | \mathbf{y}) | \mathbf{y}), \quad \phi_0(\mathbf{x} | \mathbf{y}) = \mathbf{x}. \quad (5)$$

The crucial constraint is that $\mathbf{v}_t \in \text{null}(\mathbf{A})$: a step along a null-space direction preserves $\mathbf{A} \mathbf{x} = \mathbf{y}$, so ϕ_t maps $\mathcal{M}_{\mathbf{y}}$ into itself and every state remains measurement-consistent.

We instantiate the mean and scale to vary linearly in t ,

$$\boldsymbol{\mu}_t(\mathbf{x}_1, \mathbf{y}) = t \mathbf{P} \mathbf{x}_1 + \mathbf{A}^\dagger \mathbf{y}, \quad \sigma_t(\mathbf{x}_1) = 1 - (1 - \sigma_{\min})t, \quad (6)$$

so that (4) becomes

$$\psi_t(\mathbf{x} | \mathbf{y}) = (1 - (1 - \sigma_{\min})t) \mathbf{P} \mathbf{x} + t \mathbf{P} \mathbf{x}_1 + \mathbf{A}^\dagger \mathbf{y}, \quad (7)$$

moving the null-space content from $\mathbf{P} \mathbf{x}$ to $\mathbf{P} \mathbf{x}_1$ while holding the anchor $\mathbf{A}^\dagger \mathbf{y}$ fixed. Differentiating (7) in t and re-expressing at the current state $\mathbf{x}$ yields the velocity field

$$\mathbf{v}_t(\mathbf{x} | \mathbf{y}) = \mathbf{P}(\mathbf{x}_1 - (1 - \sigma_{\min}) \mathbf{x}) \in \text{null}(\mathbf{A}). \quad (8)$$

Lemmas 1 and 3 in the appendix extend [11] to show $\mathbf{v}_t$ generates (3) on $\mathcal{M}_{\mathbf{y}}$, so the flow transports $q(\cdot | \mathbf{y})$ to $p(\cdot | \mathbf{y})$.

2.2 One-Step Sampling via Mean Flow

Sampling with the flow of Section 2.1 requires integrating the instantaneous velocity (8) from $t = 0$ to $t = 1$, which costs many network evaluations. To reconstruct in a single step, we instead learn the *average* velocity of the flow, following the Mean Flow framework [29], but restricted to $\mathcal{M}_y$.

Writing $\mathbf{u}$ for the average velocity and $\mathbf{v}$ for the instantaneous velocity, the *average conditional velocity* between two times $r \leq t$ is the time-average of $\mathbf{v}$ along the trajectory,

$$\mathbf{u}_{r,t}(\mathbf{x}_t|\mathbf{y}) := \frac{1}{t-r} \int_r^t \mathbf{v}_\tau(\mathbf{x}_\tau|\mathbf{y}) \, d\tau, \quad (9)$$

with $0 \leq r \leq t \leq 1$. The value of learning $\mathbf{u}_{r,t}$ directly is that it enables production of a sample by a single jump from $r = 0$ to $t = 1$, with no step-by-step integration of $\mathbf{v}$.

We convert (9) into a pointwise constraint relating $\mathbf{u}$ and $\mathbf{v}$ that is better suited for learning. Multiplying by $(t-r)$ and differentiating in t gives the *MeanFlow identity* over $\mathcal{M}_y$,

$$\mathbf{u}_{r,t}(\mathbf{x}_t|\mathbf{y}) = \mathbf{v}_t(\mathbf{x}_t|\mathbf{y}) - (t-r) \frac{d}{dt} \mathbf{u}_{r,t}(\mathbf{x}_t|\mathbf{y}), \quad (10)$$

which reduces to $\mathbf{u}_{t,t} = \mathbf{v}_t$ when $r = t$. Crucially, (10) involves only $\mathbf{v}_t$ and a derivative of $\mathbf{u}_{r,t}$, both available without integration, turning (9) into a regression target.

This identity yields a training objective whose minimizer is the desired average velocity. We parametrize $\mathbf{u}^\theta$ by a network and regress it against the right-hand side of (10), with $\mathbf{v}_t$ supplied in closed form by (8).

Proposition 1. *Let $\mathbf{A}$ be a fixed matrix, $\mathbf{y}$ the noiseless measurement, and $\mathbf{P} := \mathbf{I} - \mathbf{A}^\dagger \mathbf{A}$ the null-space projection. Assume $\mathbf{u}^\theta$ is a universal approximator of vector fields on $\text{null}(\mathbf{A})$. If $\mathbf{u}_{r,t}^{\theta^*}(\mathbf{x}|\mathbf{y})$ is the global minimizer of*

$$\mathbb{E} \left\| \mathbf{u}_{r,t}^\theta(\mathbf{z}|\mathbf{y}) - \mathbf{v} + (t-r) \frac{d}{dt} \mathbf{u}_{r,t}^\theta(\mathbf{z}|\mathbf{y}) \right\|_2^2,$$

where the expectation is over $r \leq t \sim \mathcal{U}[0, 1]$, $(\mathbf{x}_1, \mathbf{y}) \sim q(\mathbf{x}_1, \mathbf{y})$, $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and

$$\begin{aligned} \mathbf{z} &:= (1 - (1 - \sigma_{\min})t) \mathbf{P}\epsilon + t \mathbf{P}\mathbf{x}_1 + \mathbf{A}^\dagger \mathbf{y}, \\ \mathbf{v} &:= \mathbf{P}(\mathbf{x}_1 - (1 - \sigma_{\min})\epsilon). \end{aligned}$$

then $\mathbf{u}_{r,t}^{\theta^*}(\mathbf{x}|\mathbf{y})$ is a Mean Flow in the null space of $\mathbf{A}$ that samples from $p(\mathbf{x}|\mathbf{y})$.

The proof is given in the Appendix. This is the first theoretical result establishing a one-step posterior sampler for imaging inverse problems, extending the unconditional Mean Flow of [29] to the subspace-restricted flow of Section 2.1. Since $\mathbf{u}^\theta$ is a universal approximator, the global minimum of the ℓ_2 loss enforces the MeanFlow identity (10) pointwise, and the resulting field is the average velocity of a flow that transports $q(\cdot|\mathbf{y})$ to $p(\cdot|\mathbf{y})$ on $\mathcal{M}_y$. Algorithms 1 and 2 provide the implementation details of NullFlow, where reconstruction is initialized with $\mathbf{A}^\dagger \mathbf{y} + \mathbf{P}\epsilon$ and $\mathbf{u}_{0,1}^{\theta^*}$ is evaluated once.

3 Numerical Evaluation

We illustrate NullFlow on the problem of image inpainting using the FFHQ dataset. We consider noiseless center-mask inpainting on 256×256 RGB images, where a fixed 128×128 center patch is removed. During training, observed pixels are fixed and missing pixels are filled with independent Gaussian noise.

Following [31], we train the network with the PMF objective

$$\mathcal{L}_{\text{PMF}} = \mathbb{E}_{r,t,\mathbf{x},\mathbf{y},\epsilon} \left\| \mathbf{u}_{r,t}^\theta(\mathbf{x}_t|\mathbf{y}) - \mathbf{u}_{r,t}(\mathbf{x}_t|\mathbf{y}) \right\|_2^2, \quad (11)$$

where $\mathbf{u}_{r,t}(\mathbf{x}_t|\mathbf{y})$ is the target defined by (10). The total derivative in this target is evaluated efficiently using a Jacobian-vector product, avoiding explicit formation of the network Jacobian. To

Table 1: Quantitative results on natural image inpainting problem. **Best values** and **second-best** are color coded per metric.

Method	NFE ↓	LPIPS ↓	PSNR ↑	SSIM ↑
U-Net	1	0.123	26.59	0.896
DPIR	200	0.264	18.42	0.797
DPS	1000	0.157	23.89	0.829
DiffPIR	200	0.101	25.12	0.871
PnP-Flow	500	0.089	24.80	0.900
Flower	500	0.105	25.03	0.892
NullFlow	1	0.055	24.54	0.874

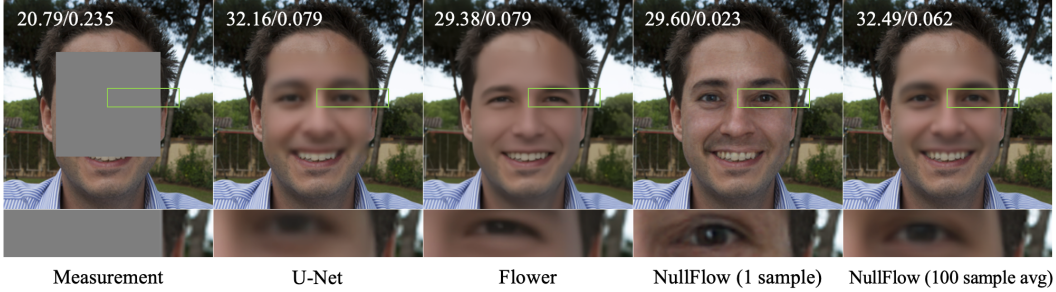


Figure 2: Reconstructions on a test image. Left to right: measurement, U-Net, Flower, a single NullFlow sample, and the average of 100 NullFlow samples. PSNR/LPIPS are shown in the top-left corner of each. A single NullFlow sample is sharp and perceptually faithful (best LPIPS), whereas averaging many samples approaches the MMSE estimate, trading perceptual quality for lower distortion and visibly resembling the MSE-trained U-Net.

improve visual fidelity, we augment $\mathcal{L}_{\text{PMF}}$ with VGG-based LPIPS [32] and frozen ConvNeXt-V2 feature regularization [33]:

$$\mathcal{L} = \mathcal{L}_{\text{PMF}} + \lambda_p \ell_p(\hat{\mathbf{x}}, \mathbf{x}) + \lambda_c \|\phi(\hat{\mathbf{x}}) - \phi(\mathbf{x})\|_2^2, \quad (12)$$

where ϕ denotes the ConvNeXt-V2 feature extractor. We set $\lambda_p = 0.4$ and $\lambda_c = 0.1$.

At inference, we initialize with the null-space-perturbed least-squares estimate $\mathbf{A}^\dagger \mathbf{y} + \mathbf{P}\epsilon$: for center-mask inpainting, this preserves the observed pixels and fills the missing center with Gaussian noise. The trained model maps this initialization to the final reconstruction in a single step.

As shown in Table 1, NullFlow achieves the best LPIPS with only one function evaluation, while maintaining competitive PSNR and SSIM. U-Net obtains the highest distortion metrics, as expected from its MSE objective, but its reconstructions are overly smooth and have worse perceptual quality. This perception–distortion tradeoff is also visible in Fig. 2: a single NullFlow sample gives sharper and more faithful details, whereas averaging many samples produces a smoother MSE/MMSE-like estimate. Fig. 3 confirms this trend quantitatively, showing that sample averaging improves PSNR and SSIM but degrades LPIPS. NullFlow thus provides both a perceptual one-step reconstruction and, through sample averaging, a distortion-oriented estimate without retraining.

4 Conclusion

We introduced NullFlow, the first one-step generative solver for imaging inverse problems. By confining a Mean Flow to the null space of the forward operator, every state stays measurement-consistent by construction and NullFlow samples from $p(\mathbf{x}|\mathbf{y})$ in a single network evaluation. On image inpainting it matches the perceptual quality of iterative diffusion and flow solvers at a fraction of the cost. Extending NullFlow to noisy measurements and to sample-varying operators are promising future directions.

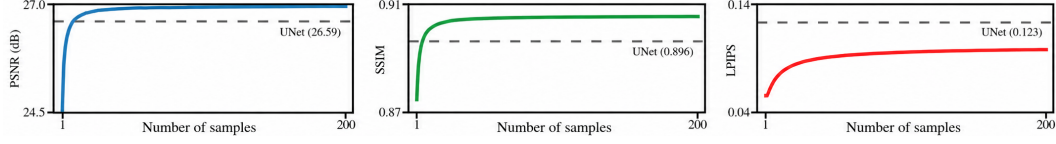


Figure 3: Effect of sample averaging on reconstruction quality. Averaging an increasing number of posterior samples approximates the MMSE estimator, leading to improved PSNR and SSIM but degraded LPIPS. The dashed lines denote the performance of a supervised U-Net trained to minimize MSE. Notably, while MMSE averaging can surpass U-Net in distortion-based metrics, it produces perceptually smoother reconstructions.

References

- [1] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang, “Image super-resolution using deep convolutional networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, 2016.
- [2] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang, “Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising,” *IEEE Trans. Image Proc.*, vol. 26, no. 7, pp. 3142–3155, 2017.
- [3] Kyong Hwan Jin, Michael T. McCann, Emmanuel Froustey, and Michael Unser, “Deep convolutional neural network for inverse problems in imaging,” *IEEE Trans. Image Proc.*, vol. 26, no. 9, pp. 4509–4522, 2017.
- [4] Johannes Schwab, Stephan Antholzer, and Markus Haltmeier, “Deep null space learning for inverse problems: convergence analysis and rates,” *Inverse Problems*, vol. 35, no. 2, pp. 025008, 2019.
- [5] Dongdong Chen and Mike E. Davies, “Deep decomposition learning for inverse imaging problems,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 510–526.
- [6] X. Xu, Y. Sun, J. Liu, B. Wohlberg, and U. S. Kamilov, “Provable convergence of plug-and-play priors with MMSE denoisers,” *IEEE Signal Process. Letters*, vol. 27, pp. 1280–1284, 2020.
- [7] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte, “Plug-and-play image restoration with deep denoiser prior,” *Proc. CVPR*, vol. 44, no. 10, pp. 6360–6376, 2021.
- [8] Marien Renaud, Jean Prost, Arthur Leclaire, and Nicolas Papadakis, “Plug-and-play image restoration with stochastic denoising regularization,” in *Proceedings of the 41st International Conference on Machine Learning*, 2024.
- [9] Jonathan Ho, Ajay Jain, and Pieter Abbeel, “Denoising diffusion probabilistic models,” *Proc. NeurIPS*, vol. 33, pp. 6840–6851, 2020.
- [10] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer, “High-resolution image synthesis with latent diffusion models,” in *Proc. CVPR*, 2022, pp. 10684–10695.
- [11] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le, “Flow matching for generative modeling,” *Proc. ICLR*, 2023.
- [12] Xingchao Liu, Chengyue Gong, and Qiang Liu, “Flow straight and fast: Learning to generate and transfer data with rectified flow,” *Proc. ICLR*, 2023.
- [13] Michael Albergo and Eric Vanden-Eijnden, “Building Normalizing Flows with Stochastic Interpolants,” in *Proc. ICLR*, 2023.
- [14] Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden, “Stochastic Interpolants: A Unifying Framework for Flows and Diffusions,” *J. Mach. Learn. Res.*, vol. 26, no. 209, pp. 1–80, 2025.
- [15] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song, “Denoising diffusion restoration models,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2022.
- [16] Hyungjin Chung, Jeongsol Kim, Michael Thompson McCann, Marc Louis Klasky, and Jong Chul Ye, “Diffusion posterior sampling for general noisy inverse problems,” in *Proc. ICLR*, 2023.
- [17] 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 *Proc. ECCV*, 2023, pp. 10498–10508.
- [18] Michael Samuel Albergo, Mark Goldstein, Nicholas Matthew Boffi, Rajesh Ranganath, and Eric Vanden-Eijnden, “Stochastic Interpolants with Data-Dependent Couplings,” in *Proc. ICML*, 2024.
- [19] U. S. Kamilov, C. A. Bouman, G. T. Buzzard, and B. Wohlberg, “Plug-and-play methods for integrating physical and learned models in computational imaging,” *IEEE Signal Process. Mag.*, vol. 40, no. 1, pp. 85–97, Jan. 2023.

- [20] Giannis Daras, Hyungjin Chung, Chieh-Hsin Lai, Yuki Mitsufuji, Jong Chul Ye, Peyman Milanfar, Alexandros G. Dimakis, and Mauricio Delbracio, “A survey on diffusion models for inverse problems,” *arXiv:2410.00083*, 2024.
- [21] Yinhuai Wang, Jiwen Yu, and Jian Zhang, “Zero-shot image restoration using denoising diffusion null-space model,” *Proc. ICLR*, 2023.
- [22] Jeongsol Kim, Bryan Sangwoo Kim, and Jong Chul Ye, “Flowdps : Flow-driven posterior sampling for inverse problems,” in *Proc. ICCV*, October 2025, pp. 12328–12337.
- [23] Mehrrsa Pourya, Bassam El Rawas, and Michael Unser, “Flower: A Flow-Matching Solver for Inverse Problems,” *Proc. ICLR*, 2026.
- [24] Ségolène Tiffany Martin, Anne Gagneux, Paul Hagemann, and Gabriele Steidl, “Pnp-flow: Plug-and-play image restoration with flow matching,” in *Proc. ICLR*, 2025.
- [25] Ricky TQ Chen and Yaron Lipman, “Flow matching on general geometries,” in *Proc. ICLR*, 2024, vol. 2024, pp. 47922–47945.
- [26] Heli Ben-Hamu, Omri Puny, Itai Gat, Brian Karrer, Uriel Singer, and Yaron Lipman, “D-flow: Differentiating through flows for controlled generation,” in *Proc. ICLR*, 2024, pp. 3462–3483.
- [27] Yasi Zhang, Peiyu Yu, Yaxuan Zhu, Yingshan Chang, Feng Gao, Ying N Wu, and Oscar Leong, “Flow priors for linear inverse problems via iterative corrupted trajectory matching,” *Proc. NeurIPS*, vol. 37, pp. 57389–57417, 2024.
- [28] Ashwini Pokle, Matthew J Muckley, Ricky TQ Chen, and Brian Karrer, “Training-free linear image inverses via flows,” *TMLR*, 2024.
- [29] Zhengyang Geng, Mingyang Deng, Xingjian Bai, J Zico Kolter, and Kaiming He, “Mean flows for one-step generative modeling,” in *Proc. NeurIPS*, 2026.
- [30] Zhengyang Geng, Yiyang Lu, Zongze Wu, Eli Shechtman, J Zico Kolter, and Kaiming He, “Improved mean flows: On the challenges of fastforward generative models,” *arXiv:2512.02012*, 2025.
- [31] Yiyang Lu, Susie Lu, Qiao Sun, Hanhong Zhao, Zhicheng Jiang, Xianbang Wang, Tianhong Li, Zhengyang Geng, and Kaiming He, “One-step latent-free image generation with pixel mean flows,” *arXiv:2601.22158*, 2026.
- [32] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proc. CVPR*, 2018, pp. 586–595.
- [33] Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie, “Convnext v2: Co-designing and scaling convnets with masked autoencoders,” in *Proc. CVPR*, 2023, pp. 16133–16142.
- [34] Zhengyang Geng, Mingyang Deng, Xingjian Bai, Zico Kolter, and Kaiming He, “Mean flows for one-step generative modeling,” *Proc. NeurIPS*, vol. 38, pp. 75460–75482, 2026.

Appendix

Lemmas 1, 2, and 3 extend Theorems 1, 2, and 3, respectively, of [11] to measurement-conditional probability paths and vector fields on $\mathcal{M}_y$. Building on this affine-subspace flow, Proposition 1 extends [29] to a Mean Flow restricted to $\mathcal{M}_y$.

A Technical Lemmas

Lemma 1. *Let $v_t(x|x_1, y)$ be conditional vector fields on $\mathcal{M}_y$ that generate probability path $p_t(x|x_1, y)$ on $\mathcal{M}_y$. For any distribution $q(x_1|y)$ on $\mathcal{M}_y$, the marginal vector field*

$$v_t(x|y) = \int_{\mathcal{M}_y} v_t(x|x_1, y) \frac{p_t(x|x_1, y)q(x_1|y)}{p_t(x|y)} \, d\text{vol}_{\mathcal{M}_y}$$

generates the marginal probability path

$$p_t(x|y) = \int_{\mathcal{M}_y} p_t(x|x_1, y)q(x_1|y) \, d\text{vol}_{\mathcal{M}_y}.$$

Proof. The proof follows that of Theorem 1 in [11], with the modification of taking integrals over the manifold, $\int d\text{vol}_{\mathcal{M}_y}$, rather than all of $\mathbb{R}^n$, ($\int dx$). $\square$

To train Flow Matching (FM) models, the following intractable loss must be minimized.

$$\mathcal{L}_{FM}(\theta) = \mathbb{E}_{t, p_t(x)} \|v_t^\theta(x|y) - v_t(x|y)\|^2.$$

The following Lemma is used to show that Conditional Flow Matching (CFM) loss

$$\mathcal{L}_{CFM}(\theta) = \mathbb{E}_{t, p(x_1), p_t(x|x_1)} \|v_t^\theta(x|y) - v_t(x|y, x_1)\|^2,$$

can be used instead of the Flow Matching (FM) loss. For more details regarding FM and CFM, we refer the reader to [11].

Lemma 2. *Assuming that $p_t(x|y) > 0$ for all $x \in \mathcal{M}_y$ and $t \in [0, 1]$, then $\nabla_\theta \mathcal{L}_{FM}(\theta) = \nabla_\theta \mathcal{L}_{CFM}(\theta)$.*

Proof. The proof follows that of Theorem 2 in [11], with the modification of taking integrals over the manifold, $\int d\text{vol}_{\mathcal{M}_y}$, rather than all of $\mathbb{R}^n$, ($\int dx$). $\square$

Lemma 3. *Let $p_t(x|x_1, y)$ be a Gaussian probability path restricted to $\mathcal{M}_y$ as defined in Eq. (3), and ψ_t its corresponding flow map as defined in Eq. (4). Then, the unique vector field that defines ψ_t has the form*

$$v_t(x|x_1, y) = \frac{\dot{\sigma}_t(x_1)}{\sigma_t(x_1)}(x - \mu_t(x_1, y)) + \dot{\mu}_t(x_1, y),$$

for $x, x_1 \in \mathcal{M}_y$. Consequently, $v_t(x|x_1, y)$ generates the Gaussian path $p_t(x|x_1, y)$.

Proof. We need to satisfy the conditional probability path $\dot{\psi}_t(x|x_1, y) = v_t(\psi_t(x|x_1, y)|x_1, y)$. Let $x \in \mathcal{M}_y$ and $z = \psi_t(x|x_1, y)$. By definition of the flow, $z \in \mathcal{M}_y$. First, notice that ψ_t is a bijection when restricted to $\mathcal{M}_y$, and so we can write $x = \psi_t^{-1}(z|x_1, y)$. Therefore,

$$v_t(z|x_1, y) = \dot{\psi}_t(\psi_t^{-1}(z|x_1, y)|x_1, y). \quad (13)$$

Since $\sigma_t > 0$, the inverse on $\mathcal{M}_y$ is given by

$$\begin{aligned} \psi_t^{-1}(z|x_1, y) &= \sigma_t^{-1}(x_1)P(z - \mu_t(x_1, y)) + A^\dagger y \\ &= \sigma_t^{-1}(x_1)(z - \mu_t(x_1, y)) + A^\dagger y, \end{aligned} \quad (14)$$

where we use that $z - \mu_t \in \text{null}(A)$ since $z, \mu_t \in \mathcal{M}_y$. We add $A^\dagger y$ to recover $z \in \mathcal{M}_y$. The time derivative of ψ_t is

$$\dot{\psi}_t(x|x_1, y) = \dot{\sigma}_t(x_1)Px + \dot{\mu}_t(x_1, y) \quad (15)$$

Plugging Eq. (13) and Eq. (14) into Eq. (15) gives:

$$\begin{aligned}
\mathbf{v}_t(\mathbf{z}|\mathbf{x}_1, \mathbf{y}) &= \dot{\psi}_t\left(\sigma_t^{-1}(\mathbf{x}_1)(\mathbf{z} - \boldsymbol{\mu}_t(\mathbf{x}_1, \mathbf{y})) + \mathbf{A}^\dagger \mathbf{y} \mid \mathbf{x}_1, \mathbf{y}\right) \\
&= \frac{\dot{\sigma}_t(\mathbf{x}_1)}{\sigma_t(\mathbf{x}_1)} \mathbf{P}(\mathbf{z} - \boldsymbol{\mu}_t(\mathbf{x}_1, \mathbf{y})) + \dot{\sigma}_t(\mathbf{x}_1) \mathbf{P} \mathbf{A}^\dagger \mathbf{y} + \dot{\boldsymbol{\mu}}_t(\mathbf{x}_1, \mathbf{y}) \\
&= \frac{\dot{\sigma}_t(\mathbf{x}_1)}{\sigma_t(\mathbf{x}_1)} (\mathbf{z} - \boldsymbol{\mu}_t(\mathbf{x}_1, \mathbf{y})) + \dot{\boldsymbol{\mu}}_t(\mathbf{x}_1, \mathbf{y}).
\end{aligned}$$

In the last line, we use $\mathbf{z} - \boldsymbol{\mu}_t \in \text{null}(\mathbf{A})$ and $\mathbf{A}^\dagger \mathbf{y} \notin \text{null}(\mathbf{A})$. $\square$

B Proof of Proposition 1

Lemmas 1–3 verify that we can learn the instantaneous velocity of a flow restricted to $\mathcal{M}_y$. Lemma 3 establishes that a vector field $\mathbf{v}_t(\mathbf{x}|\mathbf{x}_1, \mathbf{y})$ can be defined to generate the Gaussian path from $p_0(\mathbf{x}|\mathbf{x}_1, \mathbf{y})$ to $p_1(\mathbf{x}|\mathbf{x}_1, \mathbf{y})$. Lemma 1 establishes that from $\mathbf{v}_t(\mathbf{x}|\mathbf{x}_1, \mathbf{y})$, a marginal vector field $\mathbf{v}_t(\mathbf{x}|\mathbf{y})$ can be defined to generate the marginal probability path $p_t(\mathbf{x}|\mathbf{y})$. Finally, Lemma 2 establishes that such a marginal instantaneous vector field can be feasibly trained.

We now seek the average conditional velocity: using the same arguments from [34], we rewrite the *MeanFlow Identity*; differentiating (9) gives (10), where we use the Fundamental Theorem of Calculus. To see sufficiency of the MeanFlow Identity, we refer the reader to section B.3 of [29]. Rearranging terms gives the measurement conditional MeanFlow Identity:

$$\mathbf{u}_{r,t}(\mathbf{x}_t|\mathbf{y}) = \mathbf{v}_t(\mathbf{x}_t|\mathbf{y}) - (t - r) \frac{d}{dt} \mathbf{u}_{r,t}(\mathbf{x}_t|\mathbf{y}). \quad (16)$$

Finally, we use the defined conditional flow in Eq. (7), which gives the instantaneous vector field $\mathbf{v}_t$,

$$\begin{aligned}
\mathbf{v}_t(\mathbf{x}_t|\mathbf{y}) &= \frac{d}{dt} \psi_t(\mathbf{x}|\mathbf{y}) = -(1 - \sigma_{\min}) \mathbf{P} \mathbf{x} + \mathbf{P} \mathbf{x}_1 \\
&= \mathbf{P}(\mathbf{x}_1 - (1 - \sigma_{\min}) \mathbf{x}).
\end{aligned}$$

Since $\mathbf{u}^\theta$ is a universal approximator, the global minimizer of the ℓ_2 loss satisfies (10), giving the result. $\square$