FAST-DIPS: ADJOINT-FREE ANALYTIC STEPS AND HARD-CONSTRAINED LIKELIHOOD CORRECTION FOR DIFFUSION-PRIOR INVERSE PROBLEMS

Anonymous authors
Paper under double-blind review

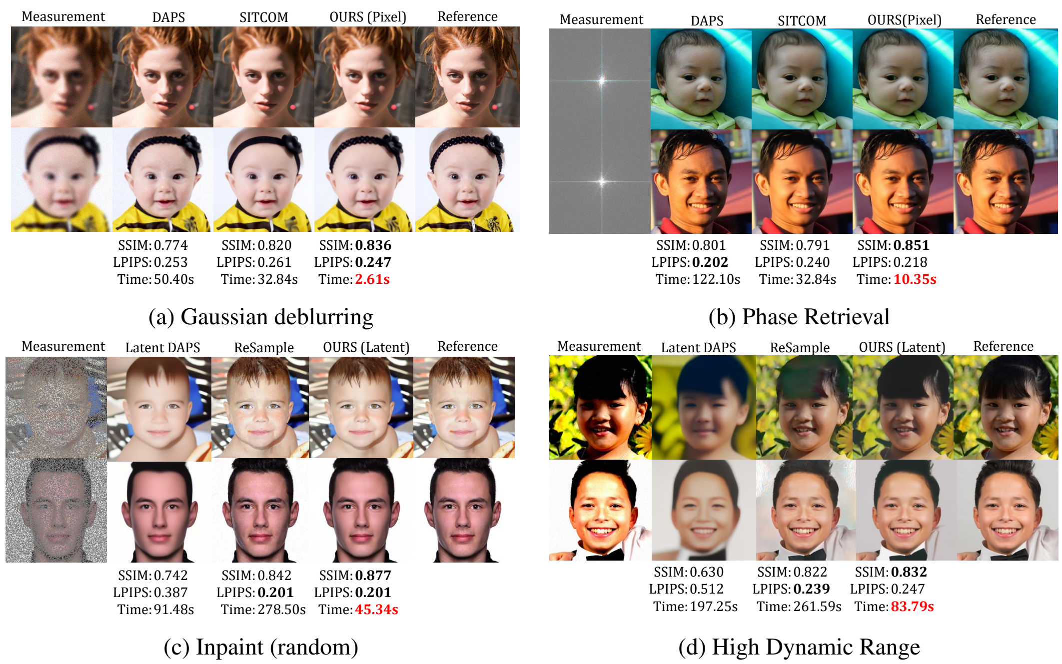

Figure 1: FFHQ results on four inverse problems: (a) Gaussian deblurring, (b) phase retrieval, (c) random inpainting, (d) HDR. Each panel shows the measurement, baselines (pixel: DAPS, SITCOM; latent: Latent DAPS, Resample), our FAST-DIPS output (pixel or latent as labeled), and the reference. SSIM/LPIPS and average per-image runtime (s) are overlaid; FAST-DIPS attains comparable or higher quality with markedly lower runtime.

ABSTRACT

FAST-DIPS is a training-free solver for diffusion-prior inverse problems, including nonlinear forward operators. At each noise level, a pretrained denoiser provides an anchor $\mathbf{x}_{0|t}$; we then perform a hard-constrained proximal correction in measurement space (AWGN) by solving $\min_{\mathbf{x}} \frac{1}{2\gamma_t}\|\mathbf{x}-\mathbf{x}_{0|t}\|^2$ s.t. $\|\mathcal{A}(\mathbf{x})-\mathbf{y}\| \leq \varepsilon$. The correction is implemented via an adjoint-free ADMM with a closed-form projection onto the Euclidean ball and a few steepest-descent updates whose step size is analytic and computable from one VJP and one JVP—or a forward-difference surrogate—followed by decoupled re-annealing. We show this step minimizes a local quadratic model (with backtracking-based descent), any ADMM fixed point satisfies KKT for the hard-constraint, and mode substitution yields a bounded time-marginal error. We also derive a latent variant ($\mathcal{A} \mapsto \mathcal{A} \circ \mathcal{D}$) and a one-parameter pixel$\rightarrow$latent hybrid schedule. FAST-DIPS delivers comparable or better PSNR/SSIM/LPIPS while being substantially faster, requiring only autodiff access to $\mathcal{A}$ and no hand-coded adjoints or inner MCMC.

1 INTRODUCTION

Inverse problems seek to recover an unknown signal $\mathbf{x}$ from partial and noisy measurements $\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{n}$. Such problems are ubiquitous in science and engineering, yet they are often ill-posed: distinct $\mathbf{x}$ can produce similar $\mathbf{y}$ due to the structure of the operator $\mathcal{A}$ and measurement noise $\mathbf{n}$. The Bayesian viewpoint constrains the solution via a prior and asks to sample from $p(\mathbf{x} \mid \mathbf{y}) \propto p(\mathbf{y} \mid \mathbf{x})\, p(\mathbf{x})$.

Diffusion models have emerged as a powerful class of learned priors for modeling complex data distributions, including natural images (Ho et al. (2020); Song & Ermon (2020); Song et al. (2021a;b); Dhariwal & Nichol (2021); Karras et al. (2022); Song et al. (2023); Lu & Song (2025)). Through reverse-time dynamics, they progressively transform simple noise into samples from the target distribution. This generative mechanism offers a natural framework for inverse problems, where the reverse-time SDE is guided by measurements to draw posterior.

Diffusion-based inverse problem solvers generally begin with an unconditional pretrained prior and impose data consistency at sampling time. Representative examples include task-specific diffusion solver (Saharia et al. (2023); Lugmayr et al. (2022); Liu et al. (2023)), linear-operator frameworks(Kawar et al. (2022); Wang et al. (2023)), and decoupled/posterior-aware updates (Chung et al. (2023a;b); Dou & Song (2024); Zhang et al. (2025)). Other lines formulate plug-and-play optimization with diffusion denoisers (Zhu et al. (2023); Rout et al. (2024); Wu et al. (2024); Xu & Chi (2024); Mardani et al. (2024); Wang et al. (2024); Zheng et al. (2025)), Monte-Carlo guidance (Cardoso et al. (2024)), or aim for faster sampling via preconditioning, parallelization, or schedule tailoring (Garber & Tirer (2024); Cao et al. (2024); Liu et al. (2024); Chung et al. (2024)). A central practical question is how data consistency is enforced. Many training-free designs rely on differentiation through $\mathcal{A}$, often in the form of explicit adjoints or pseudo-inverse, which can raise engineering barriers and restrict applicability to operators with readily available derivatives (Kawar et al. (2022); Wang et al. (2023); Rout et al. (2023); Liu et al. (2024); Pandey et al. (2024); Cao et al. (2024); Garber & Tirer (2024); Dou & Song (2024); Cardoso et al. (2024); Chung et al. (2024)). Methods that avoid hand-coded adjoints typically lean on inner iterative solvers or MCMC subloops, which increase wall-clock cost due to repeated score/denoiser calls (Zhu et al. (2023); Wu et al. (2024); Xu & Chi (2024); Mardani et al. (2024); Wang et al. (2024); Zhang et al. (2025)).

A complementary design axis is *latent* vs. *pixel* execution. Latent diffusion models reduce dimensionality and sampling cost, and many recent posterior samplers therefore operate in latent space (Rombach et al. (2022); Podell et al. (2024); Song et al. (2024); Rout et al. (2024); Zhang et al. (2025)). However, when fidelity is defined in pixel space, gradients $\nabla_{\mathbf{z}} \|\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{y}\|^2$ require repeated decoder-Jacobian evaluations, creating a throughput bottleneck. Conversely, pixel-space updates avoid the decoder but can be sensitive to how Jacobian–vector products (JVPs) are computed for highly nonlinear $\mathcal{A}$. These tradeoffs motivate methods that (i) enforce explicit measurement-space feasibility, (ii) avoid hand-coded adjoints while making minimal autodiff calls, (iii) minimize inner iterations, and (iv) leverage latent space where it helps most.

We propose **FAST-DIPS** (Fast And STable Diffusion-prior Inverse Problem Solver), a training-free framework that (i) keeps the transport across diffusion time steps decoupled, (ii) enforces a hard credible set in measurement space under an AWGN assumption (Euclidean norm), and (iii) performs the per-level correction via an adjoint-free ADMM with few-step descent update equipped with an analytic step size. Concretely, the denoiser provides a level-wise anchor; around it, we solve a hard-constrained proximal problem that projects the predicted measurement onto a ball (credible set) and updates the image by a single steepest-descent step with a step size computable from one vector–Jacobian product (VJP) and one JVP—or a forward-difference JVP fallback—followed by short backtracking. After correction we re-anneal by injecting the next-level noise, realizing the exact time-marginal recursion. We also develop a latent counterpart (replace $\mathcal{A}$ by $\mathcal{A} \circ \mathcal{D}$, where $\mathcal{D} : \mathbb{R}^k \to \mathbb{R}^{CHW}$ is a pretrained decoder; the matching encoder is $\mathcal{E}$) and a hybrid schedule that corrects in pixels early (cheap, robust) and latents late (manifold-faithful).

FAST-DIPS differs from PnP/RED-ADMM (Chan et al. (2017); Venkatakrishnan et al. (2013)): the denoiser is not used as a proximal map; instead, it supplies an anchor and ADMM enforces measurement feasibility around that anchor. Unlike quadratic data penalties that require tuning a tradeoff weight and can be brittle under noise miscalibration, we use a set-valued (indicator) likelihood in the measurement domain (AWGN), which exposes an interpretable budget. Unlike coupled DPS-style

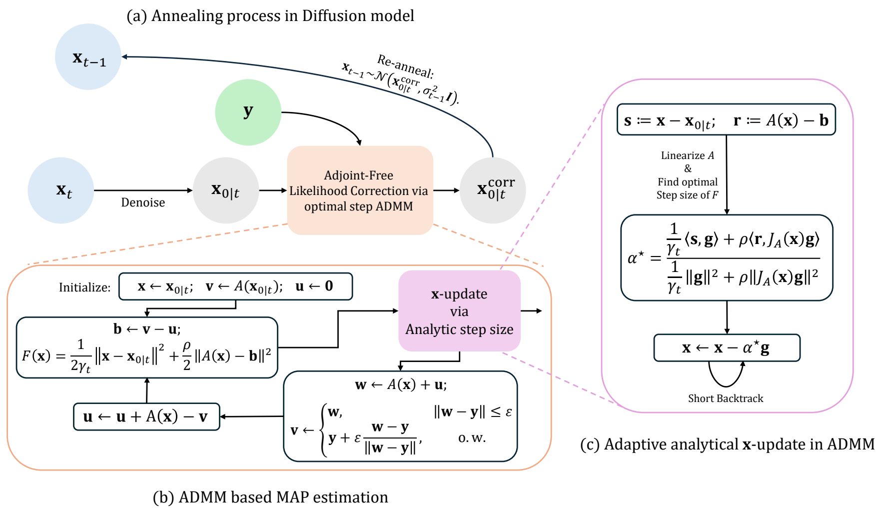

Figure 2: FAST-DIPS method sketch. At each noise level t we (1) take a denoiser anchor $\mathbf{x}_{0|t}$, (2) apply a hard-constrained correction by solving a proximal objective subject to a measurement-space credible set via few-step ADMM (closed-form projection and few-step descent with analytic α^* from one VJP + one JVP or a forward-difference surrogate), and (3) re-anneal to obtain $\mathbf{x}_{t-1}$.

guidance (Chung et al. (2023a)), we deliberately keep traversal decoupled and invoke the exact marginal transport after each correction. Relative to latent-only pipelines, our hybrid pixel→latent scheme trims decoder-Jacobian traffic early while preserving generative-manifold fidelity late. Importantly, while FAST-DIPS does assume (piecewise) differentiability to leverage automatic differentiation (Baydin et al. (2018)) for VJP (and a single JVP or its forward-difference surrogate), it does not require hand-crafted adjoints or closed-form Jacobians of $\mathcal{A}$, reducing engineering burden compared to many prior training-free designs.

Our contributions can be summarized as follows:

- **Adjoint-free hard-constrained correction.** A denoiser-anchored, measurement-space credible-set MAP with schedule-aware trust region; ADMM with closed-form projection and few analytic descent steps using one VJP + one JVP (or a forward-only probe), eliminating hand-crafted adjoints and inner MCMC.
- **Theory with practical guarantees.** The analytic step exactly minimizes a local quadratic model and, with backtracking, guarantees descent; ADMM fixed points satisfy KKT for the hard-constraint; decoupled re-annealing; mode substitution yields a bounded time-marginal error.
- **Latent & hybrid execution + empirical speed.** A latent counterpart via $A \circ D$ and a one-switch pixel→latent hybrid improve early-time throughput and late-time fidelity; across eight linear and nonlinear tasks, the method attains similar or better quality with $11.3\times$–$19.5\times$ lower runtime across pixel-space tasks on FFHQ and robust default hyperparameters.

Orthogonal to our contributions, fast samplers and preconditioning/parallelization can reduce the number of denoising steps (Zhao et al. (2024); Cao et al. (2024); Liu et al. (2024); Chung et al. (2024)). FAST-DIPS complements such advances by minimizing inner correction cost and adjoint engineering while preserving explicit measurement feasibility, so these techniques are composable with our approach.

2 Method

2.1 High-level overview and design choices

We briefly summarize the goals and main design choices of FAST-DIPS before introducing the detailed derivations.

Setting and practical issues. We use a pretrained diffusion model as a prior for inverse problems $\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{n}$, where $\mathcal{A}$ may be nonlinear. Existing training-free solvers often enforce data consistency via many gradient steps or inner Langevin/MCMC chains at each noise level, and control their influence through a soft data-fidelity weight. This can be computationally heavy (many evaluations of $\mathcal{A}$ and its gradient) and sensitive to step sizes and noise miscalibration.

Per-level update in FAST-DIPS. At each diffusion level t, FAST-DIPS performs a single correct–then–noise update with three ingredients:

1. **Local hard-constrained MAP.** The denoiser output $\mathbf{x}_{0|t}$ is treated as the center of a local Gaussian prior, and data consistency is enforced via a simple measurement-space constraint $\|\mathcal{A}(\mathbf{x}) - \mathbf{y}\| \leq \varepsilon$. This defines a constrained proximal problem whose solution $\mathbf{x}_{0|t}^{\mathrm{corr}}$ is the likelihood correction at level t.
2. **Adjoint-free ADMM with analytic step size.** We solve this constrained problem with a small, fixed number of deterministic ADMM iterations. The measurement variable is updated by a closed-form projection onto the constraint set, while the image variable is updated by steepest-descent step whose step size is chosen analytically by minimizing a local quadratic model. This optimal step design keeps the number of evaluations of $\mathcal{A}$ and its derivatives very small. No inner MCMC chain and no hand-coded adjoint operator are required, which leads to substantially fewer total function calls than methods based on long gradient/MCMC inner loops.
3. **Re-annealing.** After the correction, we re-anneal by adding Gaussian noise with the next diffusion variance, decoupling the measurement-aware update from the diffusion noise.

Theoretical support. In the remainder of the section we formalize this procedure and show that: (i) the analytic step size yields a provable decrease of a well-defined augmented objective under mild regularity (Proposition 3 together with Remark 2); (ii) any fixed point of the ADMM iterations satisfies the KKT conditions of the constrained proximal problem (Proposition 4); and (iii) replacing the full conditional by the local mode $\mathbf{x}_{0|t}^{\mathrm{corr}}$ in the re-annealing step induces a bounded approximation error on the diffusion time-marginals (Proposition 1). These results explain why FAST-DIPS can be both fast (few evaluations of $\mathcal{A}$ and its derivatives) and robust in practice.

Practical per-step recipe. In practice, one reverse diffusion step of FAST-DIPS at level t works as follows. (i) Denoiser proposal: given the current state $\mathbf{x}_t$ and noise level σ_t, we compute $\mathbf{x}_{0|t} = \mathbf{x}_{\mathrm{den}}(\mathbf{x}_t, \sigma_t)$ and initialize $\mathbf{x}^{(0)} \leftarrow \mathbf{x}_{0|t}$, $\mathbf{v}^{(0)} \leftarrow \mathcal{A}(\mathbf{x}^{(0)})$, $\mathbf{u}^{(0)} \leftarrow \mathbf{0}$. (ii) Fast ADMM correction: for a small fixed number of iterations K (typically 2–5), we update $\mathbf{x}^{(k)}$ by one steepest-descent step on a quadratic-regularized data term, with an analytic step size computed from a local quadratic model using one VJP and one JVP (or a single finite-difference probe) of $\mathcal{A}$; we then project $\mathcal{A}(\mathbf{x}^{(k+1)}) + \mathbf{u}^{(k)}$ onto the ℓ_2-ball $\{\|\mathbf{v} - \mathbf{y}\| \leq \varepsilon\}$ and update the dual variable $\mathbf{u}^{(k+1)}$. This yields the corrected estimate $\mathbf{x}_{0|t}^{\mathrm{corr}} = \mathbf{x}^{(K)}$. (iii) Re-annealing: we sample $\mathbf{x}_{t-1} = \mathbf{x}_{0|t}^{\mathrm{corr}} + \sigma_{t-1}\boldsymbol{\xi}$ with $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, I)$.

2.2 Problem Setup

Let $\mathbf{x}_0 \in \mathbb{R}^{C \times H \times W}$ denote the clean image stacked as a vector and

$$\mathbf{y} \;=\; \mathcal{A}(\mathbf{x}) + \mathbf{n}, \qquad \mathbf{n} \sim \mathcal{N}(\mathbf{0},\, \beta^2 I), \tag{1}$$

where $\mathcal{A} : \mathbb{R}^{CHW} \to \mathbb{R}^m$ is a (possibly nonlinear) forward operator. Throughout the paper we assume additive white Gaussian noise (AWGN) with variance β^2 and use the standard Euclidean norm in measurement space.

2.3 Probabilistic Motivation and the Per-Level Objective

The reverse process of the diffusion model, conditioned on $\mathbf{y}$, is described by the reverse-time SDE (Song et al. (2021b)):

$$d\mathbf{x}_t = -2\dot{\sigma}(t)\sigma(t)\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{y}; \sigma_t)\, dt + \sqrt{2\dot{\sigma}(t)\sigma(t)}\, d\mathbf{w}_t \tag{2}$$

At each diffusion level t we maintain a state $\mathbf{x}_t$ and wish to transform the time–marginal $p(\mathbf{x}_t \mid \mathbf{y})$ into a good approximation to $p(\mathbf{x}_{t-1} \mid \mathbf{y})$ by performing a local, measurement-aware likelihood correction around the denoiser's prediction. The derivation proceeds from the conditional factorization

$$p(\mathbf{x}_0 \mid \mathbf{x}_t, \mathbf{y}) \;\propto\; p(\mathbf{x}_0 \mid \mathbf{x}_t)\, p(\mathbf{y} \mid \mathbf{x}_0), \tag{3}$$

and two modeling choices: a local Laplace surrogate for $p(\mathbf{x}_0 \mid \mathbf{x}_t)$ and a set-valued likelihood in measurement space.

Local prior surrogate around the denoiser. Write

$$\mathbf{x}_{0|t} \;:=\; \mathbf{x}_{\text{den}}(\mathbf{x}_t, \sigma_t), \tag{4}$$

and approximate the intractable $p(\mathbf{x}_0 \mid \mathbf{x}_t)$ by a Gaussian centered at $\mathbf{x}_{0|t}$,

$$p(\mathbf{x}_0 \mid \mathbf{x}_t) \;\approx\; \tilde{p}_t(\mathbf{x}_0 \mid \mathbf{x}_t) \;\propto\; \exp\Big(-\tfrac{1}{2\gamma_t}\,\|\mathbf{x}_0 - \mathbf{x}_{0|t}\|^2\Big), \tag{5}$$

where $\gamma_t > 0$ plays the role of a local prior variance. We use the schedule-aware parameterization $\gamma_t = \sigma_t^2$ so that the proximal trust-region naturally tightens with annealing (Zhang et al. (2025)).

Conservative likelihood via a measurement-space credible set. Under AWGN, the Gaussian likelihood is

$$p(\mathbf{y} \mid \mathbf{x}_0, \beta) \;\propto\; \beta^{-m} \exp\Big(-\tfrac{1}{2\beta^2}\,\|\mathcal{A}(\mathbf{x}_0) - \mathbf{y}\|^2\Big), \tag{6}$$

which we replace by a set-valued surrogate that is robust to noise miscalibration while preserving a principled notion of data fidelity. If β is known, then for any confidence level $1-\delta$ the $(1-\delta)$-level set of Equation 6 is the Euclidean ball $\{\mathbf{v} : \|\mathbf{v} - \mathbf{y}\| \le \varepsilon\}$ with $\varepsilon = \beta\sqrt{\chi^2_{m,1-\delta}}$ (Casella & Berger (1990)); conditioning on this set replaces the likelihood by its indicator. If β is unknown, profiling it out gives $-\log p(\mathbf{y} \mid \mathbf{x}_0, \hat{\beta}(\mathbf{x}_0)) \propto m \log \|\mathcal{A}(\mathbf{x}_0) - \mathbf{y}\|$ (Casella & Berger (1990)), which is monotone in the residual norm; optimizing at a fixed confidence thus amounts to enforcing $\|\mathcal{A}(\mathbf{x}_0) - \mathbf{y}\| \le \varepsilon$ for a chosen budget $\varepsilon > 0$ (Engl et al. (1996)). Both viewpoints lead to the conservative surrogate

$$\tilde{\ell}_\varepsilon(\mathbf{y} \mid \mathbf{x}_0) \;\propto\; \mathbf{1}\{\|\mathcal{A}(\mathbf{x}_0) - \mathbf{y}\| \le \varepsilon\}\,. \tag{7}$$

Per-level surrogate conditional and MAP. Combining Equation 5 and Equation 7 with Equation 3 yields

$$\tilde{p}_t(\mathbf{x}_0 \mid \mathbf{x}_t, \mathbf{y}) \;\propto\; \exp\Big(-\tfrac{1}{2\gamma_t}\,\|\mathbf{x}_0 - \mathbf{x}_{0|t}\|^2\Big)\,\mathbf{1}\{\|\mathcal{A}(\mathbf{x}_0) - \mathbf{y}\| \le \varepsilon\}\,. \tag{8}$$

We take the mode of Equation 8 as the likelihood correction at level t, which solves

$$\mathbf{x}_{0|t}^{\text{corr}} \;\in\; \arg\min_{\mathbf{x} \in \mathbb{R}^{CHW}} \;\frac{1}{2\gamma_t}\,\|\mathbf{x} - \mathbf{x}_{0|t}\|^2 \;\;\text{s.t.}\;\; \|\mathcal{A}(\mathbf{x}) - \mathbf{y}\| \le \varepsilon. \tag{9}$$

Problem Equation 9 is a hard-constrained proximal objective: the first term is a schedule-aware trust region around the denoiser estimate, while the constraint enforces measurement feasibility within an uncertainty budget in the whitened space.

2.4 Decoupled re-annealing and connection to time–marginals

Let $\kappa_{t\to t-1}(\mathbf{x}_{t-1} \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \mathbf{x}_0, \sigma_{t-1}^2 I)$ denote the diffusion kernel that transports the clean image to the next diffusion state. The exact time–marginal recursion (Ho et al. (2020); Song et al. (2021b)) is

$$p(\mathbf{x}_{t-1} \mid \mathbf{y}) = \int \left[\int \kappa_{t\to t-1}(\mathbf{x}_{t-1} \mid \mathbf{x}_0)\, p(\mathbf{x}_0 \mid \mathbf{x}_t, \mathbf{y})\, d\mathbf{x}_0 \right] p(\mathbf{x}_t \mid \mathbf{y})\, d\mathbf{x}_t. \tag{10}$$

Thus, transforming $p(\mathbf{x}_t \mid \mathbf{y})$ to $p(\mathbf{x}_{t-1} \mid \mathbf{y})$ amounts to obtaining a representative $\mathbf{x}_0 \sim p(\mathbf{x}_0 \mid \mathbf{x}_t, \mathbf{y})$ and injecting Gaussian noise of variance σ_{t-1}^2. We approximate $p(\mathbf{x}_0 \mid \mathbf{x}_t, \mathbf{y})$ by $\tilde{p}_t$ in Equation 8 and substitute its mode, yielding the practical sampling rule

$$\mathbf{x}_{t-1} \;=\; \mathbf{x}_{0|t}^{\text{corr}} + \sigma_{t-1}\,\boldsymbol{\xi}, \qquad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, I). \tag{11}$$

Proposition 1 (Mode-substitution error under Laplace)**.** *Assume locally* $p(\mathbf{x}_0 \mid \mathbf{x}_t, \mathbf{y}) \approx \mathcal{N}(\boldsymbol{m}_t, \Sigma_t)$ *and let* $\mathbf{x}_t^{\rm corr}$ *solve Equation 9. Then the KL divergence between the time–marginals obtained by (i) injecting noise from* $\mathcal{N}(\boldsymbol{m}_t, \Sigma_t)$ *and (ii) injecting noise centered at* $\mathbf{x}_{0|t}^{\rm corr}$ *is bounded by*

$$\mathrm{KL}\Big(\mathcal{N}(\boldsymbol{m}_t, \Sigma_t + \sigma_{t-1}^2 I) \,\big\|\, \mathcal{N}(\mathbf{x}_{0|t}^{\rm corr}, \sigma_{t-1}^2 I)\Big) \;\leq\; \frac{\|\boldsymbol{m}_t - \mathbf{x}_{0|t}^{\rm corr}\|^2}{2\sigma_{t-1}^2} + \frac{\|\Sigma_t\|_F^2}{4\sigma_{t-1}^4}. \tag{12}$$

Consequences. *The bound is small (i) early, when* σ_{t-1}^2 *is large, and (ii) late, when* $\|\Sigma_t\|$ *is small; this justifies the decoupled rule Equation 11.*

2.5 Pixel-space ADMM solver with adjoint-free updates

We solve Equation 9 via variable splitting (Combettes & Pesquet (2011); Boyd et al. (2011)) in pixel space. Introduce an auxiliary $\mathbf{v} \approx \mathcal{A}(\mathbf{x})$ and the feasibility set $\mathcal{C} := \{\mathbf{v} : \|\mathbf{v} - \mathbf{y}\| \leq \varepsilon\}$. Consider

$$\min_{\mathbf{x},\mathbf{v}} \; \underbrace{\tfrac{1}{2\gamma_t}\|\mathbf{x} - \mathbf{x}_{0|t}\|^2}_{f(\mathbf{x})} + \underbrace{\iota_{\mathcal{C}}(\mathbf{v})}_{g(\mathbf{v})} \quad \text{s.t.} \quad \mathcal{A}(\mathbf{x}) - \mathbf{v} = \mathbf{0}, \tag{13}$$

where $\iota_{\mathcal{C}}$ is the indicator of $\mathcal{C}$. Using scaled ADMM with penalty $\rho > 0$ and scaled dual $\mathbf{u}$, we iterate

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \frac{1}{2\gamma_t}\|\mathbf{x} - \mathbf{x}_{0|t}\|^2 + \frac{\rho}{2}\|\mathcal{A}(\mathbf{x}) - \mathbf{v}^k + \mathbf{u}^k\|^2, \tag{14}$$

$$\mathbf{v}^{k+1} = \Pi_{\mathcal{C}}\Big(\mathcal{A}(\mathbf{x}^{k+1}) + \mathbf{u}^k\Big), \tag{15}$$

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \mathcal{A}(\mathbf{x}^{k+1}) - \mathbf{v}^{k+1}. \tag{16}$$

Let $\mathbf{b}^k := \mathbf{v}^k - \mathbf{u}^k$ for brevity.

Proposition 2 (Closed-form projection onto the credible set)**.** *Let* $\mathcal{C} = \{\mathbf{v} \in \mathbb{R}^m : \|\mathbf{v} - \mathbf{y}\| \leq \varepsilon\}$ *in the measurement space. Then the Euclidean projection* $\Pi_{\mathcal{C}}(\mathbf{w})$ *in Equation 15 is exactly the radial shrink (Parikh & Boyd (2014))*

$$\Pi_{\mathcal{C}}(\mathbf{w}) \;=\; \begin{cases} \mathbf{w}, & \|\mathbf{w} - \mathbf{y}\| \leq \varepsilon, \\ \mathbf{y} + \varepsilon \, \dfrac{\mathbf{w} - \mathbf{y}}{\|\mathbf{w} - \mathbf{y}\|}, & \textit{otherwise.} \end{cases} \tag{17}$$

Efficient x-update. Define the smooth objective for Equation 14

$$F(\mathbf{x}) \;=\; \frac{1}{2\gamma_t}\|\mathbf{x} - \mathbf{x}_{0|t}\|^2 + \frac{\rho}{2}\|\mathcal{A}(\mathbf{x}) - \mathbf{b}^k\|^2. \tag{18}$$

Its gradient is

$$\mathbf{g} \;=\; \nabla F(\mathbf{x}) \;=\; \frac{1}{\gamma_t}(\mathbf{x} - \mathbf{x}_{0|t}) \;+\; \rho\, J_{\mathcal{A}}(\mathbf{x})^\top\big(\mathcal{A}(\mathbf{x}) - \mathbf{b}^k\big), \qquad \mathbf{x} \leftarrow \mathbf{x} - \alpha\,\mathbf{g}, \tag{19}$$

where $J_{\mathcal{A}}(\mathbf{x})$ is the Jacobian of $\mathcal{A}$ at $\mathbf{x}$. Crucially, both the VJP $J_{\mathcal{A}}(\mathbf{x})^\top \mathbf{r}$ and the JVP $J_{\mathcal{A}}(\mathbf{x})\,\mathbf{g}$ can be obtained from autodiff (Baydin et al. (2018)).

Let $\mathbf{s} := \mathbf{x} - \mathbf{x}_{0|t}$ and $\mathbf{r} := \mathcal{A}(\mathbf{x}) - \mathbf{b}^k$. We linearize $\mathcal{A}$ along the descent direction:

$$\mathcal{A}(\mathbf{x} - \alpha \mathbf{g}) \;\approx\; \mathcal{A}(\mathbf{x}) - \alpha\, J_{\mathcal{A}}(\mathbf{x})\,\mathbf{g}. \tag{20}$$

For linear operators $\mathcal{A}$ the relation above holds exactly (since $J_{\mathcal{A}}(\mathbf{x}) \equiv \mathcal{A}$), whereas for non-linear $\mathcal{A}$ it is the first-order Taylor approximation with a higher-order remainder; nonetheless, Proposition 3 shows that even in this non-linear case the resulting analytic step $\alpha^\star$, together with backtracking, yields a descent step for the true objective F despite the residual term.

Substituting Equation 20 into $F(\mathbf{x} - \alpha \mathbf{g})$ yields a one-dimensional quadratic model (Nocedal & Wright (2006))

$$\tilde{F}(\alpha) = \frac{1}{2\gamma_t}\|\mathbf{s} - \alpha \mathbf{g}\|^2 + \frac{\rho}{2}\|\mathbf{r} - \alpha J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2, \tag{21}$$

whose exact minimizer is

$$\alpha^\star \;=\; \frac{\frac{1}{\gamma_t}\langle \mathbf{s},\, \mathbf{g}\rangle \;+\; \rho\,\langle \mathbf{r},\, J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\rangle}{\frac{1}{\gamma_t}\|\mathbf{g}\|^2 \;+\; \rho\,\|J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2} \tag{22}$$

with $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}$ obtained via a single JVP.

Proposition 3 (Local model-optimal step and descent)**.** *Under C^1 regularity of $\mathcal{A}$ near $\mathbf{x}$, $\alpha^\star$ in Equation 22 minimizes the quadratic model Equation 21. Moreover,*

$$F(\mathbf{x}-\alpha^\star\mathbf{g}) \;\le\; F(\mathbf{x}) \;-\; \frac{\left(\frac{1}{\gamma_t}\langle \mathbf{s},\mathbf{g}\rangle + \rho\langle \mathbf{r}, J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\rangle\right)^2}{2\left(\frac{1}{\gamma_t}\|\mathbf{g}\|^2 + \rho\|J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2\right)} \;+\; O(\|\mathbf{g}\|^3), \tag{23}$$

and the backtracking line search (Armijo (1966)) guarantees monotone decrease of F even when Equation 20 is only a local approximation.

Remark 1 (Linear $\mathcal{A}$ yields exact optimal line search)**.** *If $\mathcal{A}$ is linear, then Equation 20 is exact and Equation 22 gives the true optimal line-search step for F along $-\mathbf{g}$ (Nocedal & Wright (2006)), delivering the fastest progress among steepest-descent steps.*

Step Size via Finite-Difference Approximation. The analytic step size $\alpha^\star$ in Equation 22 provides a nearly optimal descent but requires a JVP, $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}$. In scenarios where an automatic differentiation engine providing JVPs is unavailable or impractical, we can estimate the JVP by a single forward probe (Nocedal & Wright (2006)):

$$J_{\mathcal{A}}(\mathbf{x})\mathbf{g} \;\approx\; \frac{\mathcal{A}(\mathbf{x}+\eta\mathbf{g})-\mathcal{A}(\mathbf{x})}{\eta} \;=:\; \frac{\Delta\mathcal{A}}{\eta}, \qquad \eta\in(10^{-4},10^{-2}] \tag{24}$$

which replaces one JVP by one extra forward evaluation of $\mathcal{A}$.

By substituting this approximation into the quadratic model's minimizer Equation 22, we derive a practical, "forward-only" step size that circumvents the need for an explicit JVP or an adjoint operator.

Remark 2 (Step size from finite-difference JVP)**.** *Replacing $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}$ in Equation 22 by $\Delta\mathcal{A}/\eta$ from Equation 24 yields the numerically stable single-forward-call step*

$$\alpha_{\mathrm{FD}} \;=\; \frac{\eta^2\,\frac{1}{\gamma_t}\langle \mathbf{s},\,\mathbf{g}\rangle \;+\; \eta\,\rho\,\langle \mathbf{r},\,\Delta\mathcal{A}\rangle}{\eta^2\,\frac{1}{\gamma_t}\|\mathbf{g}\|^2 \;+\; \rho\,\|\Delta\mathcal{A}\|^2} \qquad \textit{where} \quad \Delta\mathcal{A}=\mathcal{A}(\mathbf{x}+\eta\mathbf{g})-\mathcal{A}(\mathbf{x}). \tag{25}$$

which is algebraically equivalent to substituting $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\approx\Delta\mathcal{A}/\eta$ in Equation 22 (the scaling by η^2 avoids division by small η). Since $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}=\Delta\mathcal{A}/\eta+O(\eta)$, we have $\alpha_{\mathrm{FD}}=\alpha^\star+O(\eta)$; Armijo backtracking then preserves monotone decrease of the true F.

2.6 Optimality conditions and feasibility

Proposition 4 (Fixed points satisfy KKT for Equation 9)**.** *Let $(\mathbf{x}^*,\mathbf{v}^*,\mathbf{u}^*)$ be a fixed point of Equation 14–Equation 16. Then $\mathcal{A}(\mathbf{x}^*)=\mathbf{v}^*$, $\mathbf{v}^*\in\mathcal{C}$, and there exists $\lambda^*\geq 0$ such that*

$$\frac{1}{\gamma_t}(\mathbf{x}^*-\mathbf{x}_{0|t}) \;+\; \lambda^*\,J_{\mathcal{A}}(\mathbf{x}^*)^\top\boldsymbol{\nu}^* \;=\; 0, \qquad \lambda^*\big(\|\mathcal{A}(\mathbf{x}^*)-\mathbf{y}\|-\varepsilon\big) \;=\; 0, \tag{26}$$

where

$$\boldsymbol{\nu}^*\in\begin{cases}\left\{\dfrac{\mathcal{A}(\mathbf{x}^*)-\mathbf{y}}{\|\mathcal{A}(\mathbf{x}^*)-\mathbf{y}\|}\right\}, & \|\mathcal{A}(\mathbf{x}^*)-\mathbf{y}\|=\varepsilon,\\ \{\mathbf{0}\}, & \|\mathcal{A}(\mathbf{x}^*)-\mathbf{y}\|<\varepsilon.\end{cases}$$

Hence $\mathbf{x}^*$ satisfies the KKT conditions of Equation 9 (Bertsekas (1999)).

Remark 3 (Nonconvexity)**.** *With nonlinear $\mathcal{A}$, problem Equation 9 is generally nonconvex; we do not claim global convergence. Our guarantees are local: the $\mathbf{x}$-update descends F (Proposition 3 and Remark 2), and any fixed point satisfies KKT (Proposition 4). The outer re-annealing Equation 11, together with Proposition 1, explains robustness to residual modeling error.*

2.7 Latent FAST-DIPS

We extend the framework to latent space via Latent Diffusion Models (LDMs) by substituting the forward operator $\mathcal{A}$ with the composite operator $\mathcal{A}\circ\mathcal{D}$ (where $\mathcal{D}$ is the pretrained decoder). Under this change, the pixel-space objective, ADMM updates, and guarantees carry over, yielding adjoint-free optimization with autodiff JVPs. To balance cost and fidelity, we propose a hybrid schedule: early steps (large σ_t) apply cheaper pixel-space corrections, then switch to latent corrections once $\sigma_t \le \sigma_{\text{switch}}$ to better conform to the learned manifold. Derivations, analytic step sizes, and implementation details appear in Appendix A.1.

Task	Type	Method	FFHQ				ImageNet			
			PSNR (↑)	SSIM (↑)	LPIPS (↓)	Run-time (s)	PSNR (↑)	SSIM (↑)	LPIPS (↓)	Run-time (s)
Super resolution 4×	*Pixel*	DAPS	28.774	0.774	0.257	40.229	25.686	0.651	0.364	97.192
		SITCOM	29.555	**0.841**	**0.237**	21.591	**26.519**	**0.716**	**0.309**	65.657
		C-ΠGDM	27.794	0.807	0.209	1.404	23.645	0.631	0.313	4.085
		HRDIS	30.455	0.867	0.156	2.274	26.764	0.744	0.291	6.216
		Ours	**29.573**	**0.841**	0.244	2.726	26.367	0.714	0.334	6.266
	Latent	LatentDAPS	**29.184**	**0.825**	**0.273**	93.383	26.189	0.702	0.388	95.675
		PSLD	23.749	0.601	0.347	92.799	21.262	0.405	0.501	149.29
		ReSample	23.317	0.456	0.507	248.865	22.152	0.423	0.470	275.999
		Ours	28.634	0.797	0.283	45.304	**26.298**	**0.704**	**0.377**	46.516
Inpaint (box)	*Pixel*	DAPS	24.546	0.754	0.218	33.108	**21.399**	0.726	0.271	81.166
		SITCOM	**25.336**	**0.858**	**0.169**	24.994	20.638	**0.794**	**0.209**	73.986
		C-ΠGDM	18.294	0.731	0.358	1.277	17.514	0.676	0.360	3.683
		HRDIS	21.735	0.785	0.194	3.726	20.507	0.707	0.280	10.107
		Ours	24.605	0.850	0.190	2.937	21.381	0.777	0.278	6.347
	Latent	LatentDAPS	23.530	0.742	0.369	91.687	19.630	0.588	0.522	96.110
		PSLD	21.428	0.823	**0.126**	91.189	**21.084**	**0.803**	**0.186**	146.644
		ReSample	19.978	0.796	0.247	253.162	18.087	0.713	0.309	281.831
		Ours	**24.048**	**0.829**	0.247	45.276	19.349	0.716	0.389	45.989
Inpaint (random)	*Pixel*	DAPS	30.280	0.797	0.211	35.361	25.946	0.662	0.352	82.617
		SITCOM	**32.580**	**0.911**	**0.148**	35.499	26.201	0.702	0.351	106.182
		C-ΠGDM	25.888	0.728	0.283	1.281	23.701	0.595	0.352	3.613
		HRDIS	28.722	0.823	0.202	4.518	24.614	0.676	0.321	9.703
		Ours	31.022	0.879	0.202	2.908	**28.353**	**0.791**	**0.249**	5.857
	Latent	LatentDAPS	25.979	0.742	0.387	91.480	22.695	0.567	0.549	95.442
		PSLD	22.836	0.472	0.467	87.157	22.761	0.522	0.431	146.022
		ReSample	29.950	0.842	**0.201**	278.498	26.916	0.756	**0.255**	315.707
		Ours	**30.091**	**0.877**	**0.201**	45.335	**27.245**	**0.775**	0.288	46.454
Gaussian deblurring	*Pixel*	DAPS	28.895	0.775	0.253	50.400	25.946	0.662	0.352	94.605
		SITCOM	28.775	0.820	0.261	32.841	**26.201**	0.702	0.351	103.338
		C-ΠGDM	24.432	0.678	0.368	1.305	23.701	0.595	0.352	4.881
		HRDIS	27.674	0.791	0.259	2.569	24.575	0.633	0.419	5.960
		Ours	**29.406**	**0.836**	**0.247**	2.612	26.181	**0.705**	**0.344**	4.958
	Latent	LatentDAPS	25.742	0.732	0.384	93.313	22.818	0.561	0.543	98.340
		PSLD	16.807	0.227	0.569	94.823	16.608	0.212	0.566	148.738
		ReSample	26.345	0.661	0.329	292.612	23.530	0.497	0.439	333.822
		Ours	**28.006**	**0.793**	**0.312**	46.307	**25.356**	**0.661**	**0.424**	48.229
Motion deblurring	*Pixel*	DAPS	31.074	0.829	0.199	50.924	28.838	0.776	0.243	94.681
		SITCOM	31.172	0.872	0.203	36.684	28.875	**0.807**	0.247	103.338
		Ours	**31.736**	**0.878**	**0.171**	2.616	**29.037**	0.799	**0.236**	4.623
	Latent	LatentDAPS	26.649	0.757	0.361	93.400	23.557	0.592	0.513	97.988
		PSLD	19.237	0.288	0.518	90.682	18.327	0.288	0.544	148.151
		ReSample	28.744	0.754	**0.262**	302.828	24.845	0.579	0.404	316.985
		Ours	**29.285**	**0.822**	0.278	46.785	**26.627**	**0.709**	**0.386**	47.282
Phase retrieval	*Pixel*	DAPS	**30.253**	0.801	**0.202**	122.100	22.354	0.519	0.402	320.926
		SITCOM	28.512	0.791	0.240	37.425	18.704	0.393	0.519	99.103
		HRDIS	23.670	0.537	0.448	12.020	14.019	0.195	0.722	29.915
		Ours	29.253	**0.851**	0.218	10.354	19.738	0.490	0.479	16.629
	Latent	LatentDAPS	23.450	0.695	0.418	193.005	17.067	0.446	0.624	202.426
		ReSample	24.676	0.606	0.412	321.227	16.913	0.320	0.608	354.430
		Ours	**28.330**	**0.789**	**0.244**	87.167	**18.874**	**0.441**	**0.507**	85.520
Nonlinear deblur	*Pixel*	DAPS	28.907	0.780	0.222	763.863	27.537	0.734	0.266	1453.314
		SITCOM	**29.770**	**0.844**	**0.190**	43.040	**28.138**	**0.791**	**0.218**	113.165
		HRDIS	24.929	0.658	0.357	3.094	22.553	0.504	0.448	6.653
		Ours	27.818	0.803	0.280	57.903	25.607	0.695	0.373	62.350
	Latent	LatentDAPS	25.151	0.727	0.384	229.700	22.516	0.568	0.530	249.639
		ReSample	**28.748**	0.797	**0.236**	1276.326	26.047	0.697	**0.301**	1250.783
		Ours	28.746	**0.823**	0.260	110.567	**26.234**	**0.720**	0.350	113.537
High dynamic range	*Pixel*	DAPS	26.988	0.834	0.196	103.243	26.568	**0.819**	**0.198**	293.286
		SITCOM	**27.628**	0.808	0.214	38.150	**26.849**	0.796	0.207	109.946
		HRDIS	26.346	0.836	**0.178**	2.428	24.623	0.825	0.199	5.989
		Ours	26.275	**0.843**	0.218	7.212	24.522	0.775	0.290	13.367
	Latent	LatentDAPS	20.789	0.630	0.512	197.250	19.394	0.469	0.641	207.469
		ReSample	25.038	0.822	**0.239**	261.558	**24.950**	**0.783**	**0.257**	285.495
		Ours	**25.869**	**0.832**	0.247	83.790	24.415	0.773	0.291	85.685

Table 1: Quantitative evaluation on 100 FFHQ images and 100 ImageNet images for eight inverse problems (five linear and three nonlinear). The best and second-best results within each task type (Pixel and Latent) are indicated in **bold** and underlined, respectively. Method names shown in gray denote methods designed for noiseless settings

3 Experiments

3.1 Experimental Setup

Our experimental setup, including the suite of inverse problems and noise levels, largely follows that of DAPS (Zhang et al. (2025). We evaluate our method across eight tasks—five linear and three nonlinear—to demonstrate its versatility.

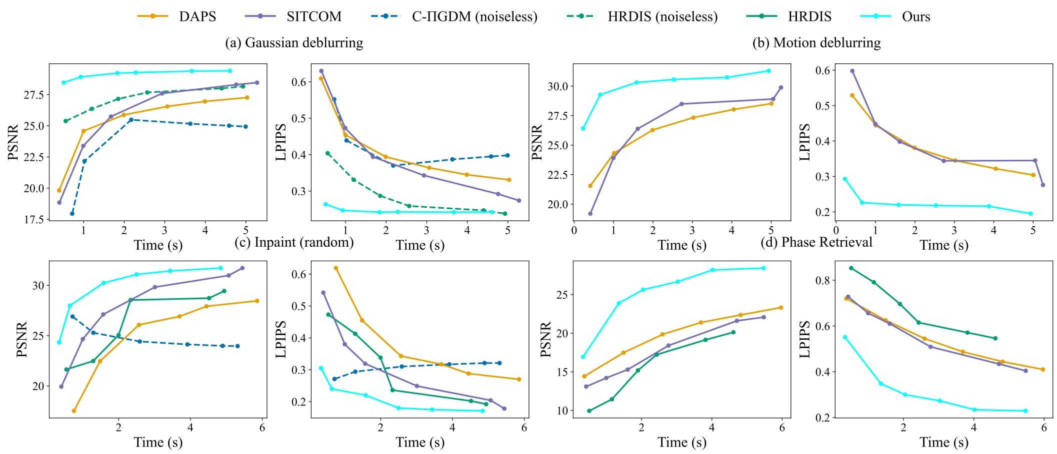

Figure 3: Quantitative evaluations comparing image quality and computational time for baseline methods. Each point is derived from an experiment on 100 FFHQ images. The y-axis value (PSNR or LPIPS) is the mean of the scores from the 100 resulting images. The x-axis value is the average per-image runtime, calculated by dividing the total processing time for all 100 images by 100. The plots show results for three linear tasks (a-c) and one nonlinear task (d).

Implementation details. For all experiments, we employ pretrained diffusion models in both pixel and latent domains. For the pixel-space setting, we use diffusion models trained on FFHQ (Chung et al. (2023a)) and ImageNet (Dhariwal & Nichol (2021)). For the latent-space setting, we use the unconditional LDM-VQ4 model (Rombach et al. (2022)) for both FFHQ and ImageNet. These models are used consistently across all baselines and our method to ensure a fair comparison. We adopt the time-step discretization and noise schedule from EDM (Karras et al. (2022)). Evaluation is performed on 100 images from FFHQ (256×256) and 100 images from ImageNet (256×256). Across all tasks, measurements are corrupted by additive Gaussian noise with a standard deviation of $\beta = 0.05$, and performance is reported using PSNR, SSIM, and LPIPS.

Baselines. We compare our method against a range of state-of-the-art baselines in both pixel and latent spaces. In pixel space, we include recent fast-sampling methods such as SITCOM (Alkhouri et al. (2025)), C-ΠGDM (Pandey et al. (2024)), and HRDIS (Dou et al. (2025)), alongside DAPS (Zhang et al. (2025)), which is recognized for its balance of performance and efficiency. For latent-space comparisons, we benchmark against prominent methods including PSLD (Rout et al. (2023)), ReSample (Song et al. (2024)), and Latent-DAPS. Details of the baseline methods are provided in Appendix A.7.

3.2 Main Results

Table 1 presents the quantitative results on the FFHQ and ImageNet datasets, where all baselines are run with their official default settings. In pixel space, our method achieves comparable or superior performance to the baselines across nearly all tasks, but with a significantly lower run-time. This acceleration is particularly evident in Gaussian and motion deblurring, where FAST-DIPS is about 19.4$\times$ faster than DAPS while also achieving higher scores. For the challenging nonlinear task of phase retrieval, we follow the common practice of selecting the best of four independent runs. In this setting, our method is approximately 11.8$\times$ faster than DAPS, while achieving higher PSNR and SSIM on FFHQ. Furthermore, our approach addresses key inefficiencies commonly found in latent-space methods. While most guided techniques suffer from long run-times due to the computational cost of backpropagating through the decoder, our hybrid pixel-latent schedule avoids this bottleneck. By performing corrections in pixel space during the early sampling stages and switching to latent-space correction later, our method effectively reduces sampling time while maintaining high-quality, manifold-faithful reconstructions.

Table 1 alone does not fully capture how different methods compare under the same run-time budget. To offer a more comprehensive evaluation, Figure 3 reports PSNR and LPIPS while considering the computational runtime. For this benchmark, we vary only the number of sampling steps/inner iterations per method, while all other hyperparameters were kept at their originally proposed optimal values to ensure a fair comparison. (Full details are provided in Appendix A.7). We evaluate three linear and one nonlinear task in total. Across all four tasks, our method consistently improves metrics in proportion to run time while maintaining a clear gap over competing baselines. The advantage is particularly pronounced in motion deblurring and phase retrieval, where the superiority highlighted earlier is equally evident under identical run-time budgets. In Gaussian deblurring, even compared to noiseless baselines, our method quickly attains strong PSNR and LPIPS in the early stage and sustains or further improves them as sampling proceeds. This robust performance was also mirrored in random inpainting. For this task, perceptual quality is paramount, and our method demonstrates its ability to generate natural-looking results by consistently maintaining a low LPIPS.

We include additional experiments on FAST-DIPS in Appendix A.8, covering the effectiveness of the $\mathbf{x}$-update step, hyperparameter robustness, the hybrid schedule trade-off, experiments with non-Gaussian noise and qualitative results in both pixel and latent spaces.

3.3 Ablation studies

We study two factors inside the per-level correction: whether we enforce feasibility by projection and how we choose the step size for the $\mathbf{x}$-update. The projection variant is our default FAST-DIPS (ADMM + proj.); the no-projection control is an unsplit penalized solver we call QDP (no splitting, no proj.), which minimizes the same quadratic objective as the ADMM $\mathbf{x}$-subproblem. To compare fairly, we equalize compute by counting first-order autodiff work: each $\mathbf{x}$-gradient step uses one forward of A, one VJP, and one JVP (or a single forward probe for FD); projection and dual updates are negligible. With K ADMM iterations and S gradient steps per iteration, FAST-DIPS spends $K \times S$ such triplets at each diffusion level, so we give QDP exactly $K \times S$ gradient steps per level. For step size we compare a tuned constant α, the analytic model-optimal $\alpha^\star$ (one VJP + one JVP), and a forward-only finite-difference surrogate α_{FD}. Full protocol and numbers are provided in Appendix A.4, Table 3.

On a representative linear pixel task (Gaussian blur), α_{FD} reaches virtually the same quality as $\alpha^\star$ at lower cost; on the nonlinear latent HDR task the optimization is sensitive to a fixed step and the JVP-based $\alpha^\star$ is the robust choice, whereas α_{FD} tends to underperform through the decoder–forward stack. Enforcing feasibility by projection consistently improves quality relative to the unsplit penalty path under the matched budget; the extra cost in latent space is dominated by backprop through the decoder rather than the projection. A practical recipe is therefore to use α_{FD} in pixel space and $\alpha^\star$ in latent space within FAST-DIPS.

4 Conclusion

Our proposed method, FAST-DIPS, targets practical challenges in training-free, diffusion-based inverse problems. It is broadly applicable: by using VJP and JVP from automatic differentiation instead of a hand-crafted adjoint, it can handle a wide range of linear and nonlinear forward models without requiring SVDs or pseudo-inverses.

For the guidance step, we replace generic optimizers (e.g., Adam with tuned learning rates) by a single gradient update with an analytic step size from a local quadratic model. This deterministic update removes step-size hyperparameters and improves efficiency and stability.

Empirically, FAST-DIPS works well for both noisy and noiseless problems and shows a predictable trade-off between computation and quality: more correction steps consistently improve reconstructions. The framework also does not rely on carefully chosen initial samples. Limitations and future directions are discussed in Appendix A.9.

5 REPRODUCIBILITY STATEMENT

Our experimental setup (datasets, pretrained models, forward operators, noise levels, metrics, and hardware) is specified in Section 3.1. In brief, we use publicly available pixel- and latent-space diffusion priors on the FFHQ-256 validation set, the EDM discretization, additive Gaussian measurement noise with $\beta = 0.05$, and evaluate PSNR/SSIM/LPIPS on 100 images. Most experiments are performed using a single NVIDIA RTX 4090 GPU, whereas the experiments reported in Tables 5, 7, 9 were conducted on a single RTX 6000 Ada GPU. For phase retrieval, we follow the common "best-of-4" protocol. Baselines are run from the authors' official repositories with their recommended defaults; Appendix A.7 lists the packages we used and task-specific settings. If the paper is accepted, we will release a public repository with scripts and configs.

REFERENCES

Ismail Alkhouri, Shijun Liang, Cheng-Han Huang, Jimmy Dai, Qing Qu, Saiprasad Ravishankar, and Rongrong Wang. SITCOM: Step-wise triple-consistent diffusion sampling for inverse problems. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2025.

Larry Armijo. Minimization of functions having lipschitz continuous first partial derivatives. *Pacific Journal of Mathematics*, 16(1):1–3, 1966.

Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: A survey. *Journal of Machine Learning Research*, 18:153:1–153:43, 2018.

Dimitri P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 2 edition, 1999.

Stephen P. Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

Jiezhang Cao, Yue Shi, Kai Zhang, Yulun Zhang, Radu Timofte, and Luc Van Gool. Deep equilibrium diffusion restoration with parallel sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2824–2834, 2024.

Gabriel V. Cardoso, Yazid Janati El Idrissi, Sylvain Le Corff, and Eric Moulines. Monte carlo guided denoising diffusion models for bayesian linear inverse problems. In *International Conference on Learning Representations (ICLR)*, 2024.

George Casella and Roger L. Berger. *Statistical Inference*. Duxbury Press, 1990.

Stanley H. Chan, Xiran Wang, and Omar A. Elgendy. Plug-and-play ADMM for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1): 84–98, 2017.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations (ICLR)*, 2023a.

Hyungjin Chung, Jeongsol Kim, and Jong Chul Ye. Direct diffusion bridge using data consistency for inverse problems. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023b.

Hyungjin Chung, Suhyeon Lee, and Jong Chul Ye. Decomposed diffusion sampler for accelerating large-scale inverse problems. In *International Conference on Learning Representations (ICLR)*, 2024.

Patrick L. Combettes and Jean-Christophe Pesquet. Proximal splitting methods in signal processing. In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pp. 185–212. Springer, 2011.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

Hongkun Dou, Zeyu Li, Jinyang Du, Lijun Yang, Wen Yao, and Yue Deng. Hybrid regularization improves diffusion-based inverse problem solving. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

Zehao Dou and Yang Song. Diffusion posterior sampling for linear inverse problem solving: A filtering perspective. In *International Conference on Learning Representations (ICLR)*, 2024.

Heinz W. Engl, Martin Hanke, and Andreas Neubauer. *Regularization of Inverse Problems*. Kluwer Academic Publishers, 1996.

Tomer Garber and Tom Tirer. Image restoration by denoising diffusion models with iteratively preconditioned guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25245–25254, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020.

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 (NeurIPS)*, volume 35, 2022.

Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022.

Gongye Liu, Haoze Sun, Jiayi Li, Fei Yin, and Yujiu Yang. Accelerating diffusion models for inverse problems through shortcut sampling. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1101–1109, 2024.

Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos A. Theodorou, Weili Nie, and Anima Anandkumar. I^2SB: Image-to-image schrödinger bridge. *arXiv preprint arXiv:2302.05872*, 2023.

Cheng Lu and Yang Song. Simplifying, stabilizing and scaling continuous-time consistency models. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. RePaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11451–11461, 2022.

Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024.

Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2 edition, 2006.

Kushagra Pandey, Ruihan Yang, and Stephan Mandt. Fast samplers for inverse problems in iterative refinement models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Neal Parikh and Stephen P. Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1 (3):127–239, 2014.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

Christoph Reich, Biplob Debnath, Deep Patel, and Srimat Chakradhar. Differentiable JPEG: The Devil is in the Details. In *Winter Conference on Applications of Computer Vision*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, 2022.

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 *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

Litu Rout, Yujia Chen, Abhishek Kumar, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Beyond first-order Tweedie: Solving inverse problems using latent diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9472–9481, 2024.

Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4713–4726, 2023.

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 *International Conference on Learning Representations (ICLR)*, 2024.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*, 2021a.

Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020.

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 (ICLR)*, 2021b.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202, pp. 32211–32252. PMLR, 2023.

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 (GlobalSIP)*, pp. 945–948. IEEE, 2013.

Hengkang Wang, Xu Zhang, Taihui Li, Yuxiang Wan, Tiancong Chen, and Ju Sun. DMPlug: A plug-in method for solving inverse problems with diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. In *International Conference on Learning Representations (ICLR)*, 2023.

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 *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Xingyu Xu and Yuejie Chi. Provably robust score-based diffusion posterior sampling for plug-and-play image reconstruction. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Bingliang Zhang, Wenda Chu, Chenlin Meng, Anima Anandkumar, Julius Berner, and Yang Song. Improving diffusion inverse problem solving with decoupled noise annealing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Jiankun Zhao, Bowen Song, and Liyue Shen. CoSiGN: Few-step guidance of consistency model to solve general inverse problems. In *European Conference on Computer Vision (ECCV)*, pp. 108–126, 2024.

Yang Zheng, Wen Li, and Zhaoqiang Liu. Integrating intermediate layer optimization and projected gradient descent for solving inverse problems with diffusion models. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2025.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2023.

A Appendix

A.1 Latent-space FAST-DIPS and a hybrid pixel–latent schedule

The pixel-space method in 2.5 corrects the denoiser's proposal directly in image space. In many diffusion systems, however, the prior is trained in a lower-dimensional *latent* space. Let $E : \mathbb{R}^{CHW} \to \mathbb{R}^k$ and $\mathcal{D} : \mathbb{R}^k \to \mathbb{R}^{CHW}$ denote a pretrained encoder–decoder with $\mathbf{z}_0 = E(\mathbf{x}_0)$ and $\mathbf{x}_0 = \mathcal{D}(\mathbf{z}_0)$. Measurements are still acquired in pixel space via Equation 1. A latent denoiser $\mathbf{z}_{\text{den}}(\mathbf{z}_t, \sigma_t)$ is available from the diffusion prior. We now derive a latent analogue of the per-level objective and show that all pixel-space results transfer verbatim under the substitution $\mathcal{A} \mapsto \mathcal{A}\circ\mathcal{D}$ and $\mathbf{x} \leftrightarrow \mathbf{z}$.

Per-level surrogate in latent space. At level t, the denoiser proposes $\mathbf{z}_{0|t} := \mathbf{z}_{\text{den}}(\mathbf{z}_t, \sigma_t)$. As in §2.3, we approximate $p(\mathbf{z}_0 \,|\, \mathbf{z}_t)$ by a local Gaussian centered at $\mathbf{z}_{0|t}$ with variance parameter $\gamma_z > 0$ (we use $\gamma_z = \sigma_t^2$ for schedule-awareness), and we employ the same credible-set likelihood surrogate in the whitened measurement space, now expressed through the decoder:

$$\tilde{p}_t(\mathbf{z}_0 \mid \mathbf{z}_t, \mathbf{y}) \;\propto\; \exp\Big(-\tfrac{1}{2\gamma_z} \|\mathbf{z}_0 - \mathbf{z}_{0|t}\|^2 \Big)\, \mathbf{1}\{\|\mathcal{A}(\mathcal{D}(\mathbf{z}_0)) - \mathbf{y}\| \le \varepsilon_z\}\,. \tag{27}$$

Taking the mode yields the latent per-level MAP:

$$\mathbf{z}_t^{\text{corr}} \;\in\; \arg\min_{\mathbf{z}\in\mathbb{R}^k} \; \frac{1}{2\gamma_z} \|\mathbf{z} - \mathbf{z}_{0|t}\|^2 \;\; \text{s.t.} \;\; \|\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{y}\| \le \varepsilon_z. \tag{28}$$

Re-annealing then follows the same transport rule as Equation 11:

$$\mathbf{z}_{t-1} \;=\; \mathbf{z}_t^{\text{corr}} + \sigma_{t-1}\,\boldsymbol{\xi}, \qquad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, I), \qquad \mathbf{x}_{t-1} = \mathcal{D}(\mathbf{z}_{t-1}). \tag{29}$$

ADMM in latent space and adjoint-free updates. Introduce $\mathbf{v} \approx \mathcal{A}(\mathcal{D}(\mathbf{z}))$ and the same feasibility set $\mathcal{C} := \{\mathbf{v} : \|\mathbf{v} - \mathbf{y}\| \le \varepsilon_z\}$. The scaled ADMM iterations mirror Equation 14–Equation 16:

$$\mathbf{z}^{k+1} = \arg\min_{\mathbf{z}} \; \frac{1}{2\gamma_z} \|\mathbf{z} - \mathbf{z}_{0|t}\|^2 + \frac{\rho_z}{2} \|\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{v}^k + \mathbf{u}^k\|^2, \tag{30}$$

$$\mathbf{v}^{k+1} = \Pi_{\mathcal{C}}\Big(\mathcal{A}(\mathcal{D}(\mathbf{z}^{k+1})) + \mathbf{u}^k \Big), \tag{31}$$

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \mathcal{A}(\mathcal{D}(\mathbf{z}^{k+1})) - \mathbf{v}^{k+1}. \tag{32}$$

The projection $\Pi_{\mathcal{C}}$ is identical to Equation 17 because feasibility is enforced *in measurement space*. For the $\mathbf{z}$-update, define

$$F_z(\mathbf{z}) \;=\; \frac{1}{2\gamma_z} \|\mathbf{z} - \mathbf{z}_{0|t}\|^2 + \frac{\rho_z}{2} \|\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b}^k\|^2, \qquad \mathbf{b}^k := \mathbf{v}^k - \mathbf{u}^k, \tag{33}$$

whose gradient is

$$\mathbf{g}_z \;=\; \nabla F_z(\mathbf{z}) \;=\; \frac{1}{\gamma_z} (\mathbf{z} - \mathbf{z}_{0|t}) \;+\; \rho_z\, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})^\top \big(\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b}^k \big), \qquad \mathbf{z} \leftarrow \mathbf{z} - \alpha\, \mathbf{g}_z. \tag{34}$$

As in pixel space, both the VJP $J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})^\top \mathbf{r}$ and the JVP $J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z$ are obtained directly from autodiff (backprop through $\mathcal{D}$ and $\mathcal{A}$; forward-mode or a single finite-difference for the JVP if needed), so the update remains *adjoint-free*.

Analytic step size in latent space. Let $\mathbf{s}_z := \mathbf{z} - \mathbf{z}_{0|t}$ and $\mathbf{r} := \mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b}^k$. Linearizing $\mathcal{A}\circ\mathcal{D}$ along $-\mathbf{g}_z$ gives $\mathcal{A}(\mathcal{D}(\mathbf{z} - \alpha\mathbf{g}_z)) \approx \mathcal{A}(\mathcal{D}(\mathbf{z})) - \alpha\, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\, \mathbf{g}_z$. The scalar quadratic model

$$\tilde{F}_z(\alpha) = \frac{1}{2\gamma_z} \|\mathbf{s}_z - \alpha \mathbf{g}_z\|^2 + \frac{\rho_z}{2} \|\mathbf{r} - \alpha J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z\|^2 \tag{35}$$

is minimized at

$$\alpha_z^* \;=\; \frac{\frac{1}{\gamma_z} \langle \mathbf{s}_z,\, \mathbf{g}_z \rangle \;+\; \rho_z\, \langle \mathbf{r},\, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z \rangle}{\frac{1}{\gamma_z} \|\mathbf{g}_z\|^2 \;+\; \rho_z\, \|J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z\|^2} \tag{36}$$

followed by clamping $\alpha \leftarrow \max(0, \alpha_z^*)$ and backtracking to ensure descent of F_z.

Proposition 5 (Local model-optimal step and descent in latent space)**.** *Under C^1 regularity of $\mathcal{A}\circ\mathcal{D}$ near $\mathbf{z}$, the step Equation 36 minimizes the quadratic model $\tilde{F}_z(\alpha)$ and*

$$F_z(\mathbf{z}-\alpha_z^*\mathbf{g}_z) \;\le\; F_z(\mathbf{z}) \;-\; \frac{\big(\frac{1}{\gamma_z}\langle \mathbf{s}_z,\mathbf{g}_z\rangle + \rho_z\langle \mathbf{r}, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z\rangle\big)^2}{2\big(\frac{1}{\gamma_z}\|\mathbf{g}_z\|^2 + \rho_z\|J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z\|^2\big)} \;+\; O(\|\mathbf{g}_z\|^3), \tag{37}$$

with monotone decrease ensured by backtracking.

Proposition 6 (KKT at latent ADMM fixed points)**.** *If $(\mathbf{z}^*,\mathbf{v}^*,\mathbf{u}^*)$ is a fixed point of Equation 30–Equation 32, then $\mathcal{A}(\mathcal{D}(\mathbf{z}^*)) = \mathbf{v}^* \in \mathcal{C}$ and there exists $\lambda^* \ge 0$ such that*

$$\frac{1}{\gamma_z}(\mathbf{z}^*-\mathbf{z}_{0|t}) \;+\; \lambda^*\, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z}^*)^\top \boldsymbol{\nu}^* \;=\; \mathbf{0}, \qquad \lambda^*\big(\|\mathcal{A}(\mathcal{D}(\mathbf{z}^*))-\mathbf{y}\|-\varepsilon_z\big)=0, \tag{38}$$

with $\boldsymbol{\nu}^* = (\mathcal{A}(\mathcal{D}(\mathbf{z}^*))-\mathbf{y})/\|\mathcal{A}(\mathcal{D}(\mathbf{z}^*))-\mathbf{y}\|$ when the constraint is active and $\boldsymbol{\nu}^*=\mathbf{0}$ otherwise.

Remark 4 (Transfer of pixel-space results)**.** *All propositions in §2.4–§2.6 transfer to the latent case by replacing $\mathcal{A}$ with $\mathcal{A}\circ\mathcal{D}$ and $\mathbf{x}$ with $\mathbf{z}$: the mode-substitution KL bound remains unchanged because feasibility and annealing live in measurement space; the projection stays exact; and the analytic step and KKT statements follow by the same quadratic-model and fixed-point arguments. In particular, the latent method is also adjoint-free in practice because both VJP and JVP are provided by autodiff across $\mathcal{D}$ and $\mathcal{A}$.*

Why (and when) prefer latent updates. Late in the schedule, σ_t is small, the denoiser's latent prediction $\mathbf{z}_{0|t}$ lies near the generative manifold, and optimizing in $\mathbf{z}$ respects that geometry by construction. Early in the schedule, however, correcting in pixel space is often cheaper (no backprop through $\mathcal{D}$) and sufficiently robust because injected noise dominates the time–marginal. This observation motivates a *hybrid* schedule.

Hybrid pixel–latent schedule. We adopt a single switching parameter σ_{switch}: for $\sigma_t > \sigma_{\text{switch}}$ we correct in pixel space using Equation 9–Equation 16, then re-encode$\mathbf{z} \leftarrow E(\mathbf{x})$ before annealing in latent space; once $\sigma_t \le \sigma_{\text{switch}}$, we correct directly in latent space using Equation 28–Equation 32. This keeps early iterations light and late iterations manifold-faithful.

Complexity and switching. A latent $\mathbf{z}$-gradient step costs one pass through $\mathcal{D}$ and $\mathcal{A}$ to form $\mathbf{r}$, one VJP through $\mathcal{A}\circ\mathcal{D}$ to form $J_{\mathcal{A}\circ\mathcal{D}}^\top\mathbf{r}$, and one JVP to form $J_{\mathcal{A}\circ\mathcal{D}}\mathbf{g}_z$; we found this JVP-based step is effective for nonlinear-deblur in latent space. In pixel space, for strongly nonlinear $\mathcal{A}$ we recommend the FD variant Equation 24+Equation 25, which swaps the JVP for a single extra forward call and was both faster and more stable in our nonlinear-deblur experiments. The switch σ_{switch} trades early-time efficiency for late-time fidelity; a stable default is to place it where the SNR of the denoiser's prediction visibly improves (e.g., where γ_t becomes comparable to the scale of $\|\mathbf{x}-\mathbf{x}_{0|t}\|$ in Equation 18).

Remark 5 (Consistency of pixel $\to$ encode with latent correction)**.** *If E and $\mathcal{D}$ are approximately inverses near the data manifold (i.e., $\mathcal{D}(E(\mathbf{x}))\approx\mathbf{x}$ and $E(\mathcal{D}(\mathbf{z}))\approx\mathbf{z}$) and are locally Lipschitz, then a pixel correction followed by $\mathbf{z} \leftarrow E(\mathbf{x})$ produces a latent iterate within $O(\|\mathcal{D}\circ E - \mathrm{Id}\|)$ of the one obtained by one latent correction step with the same residual budget. Thus the hybrid scheme is a coherent approximation of the pure latent method early in the schedule.*

A.2 ALGORITHMS

Algorithm 1 FAST-DIPS in Pixel Space

Require: measurement $\mathbf{y}$; schedule $\{\sigma_t\}$; denoiser $\mathbf{x}_{\mathrm{den}}(\cdot, \sigma_t)$; forward $\mathcal{A}$; parameters ρ, $\{\gamma_t\}$, K, S, η

Ensure: reconstructed image $\mathbf{x}_0$

1: Sample $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \sigma_T^2 I)$
2: **for** $t = T$ **down to** 1 **do**
3:     *predict* $\mathbf{x}_{0|t} \leftarrow \mathbf{x}_{\mathrm{den}}(\mathbf{x}_t, \sigma_t)$
4:     Initialize $\mathbf{x} \leftarrow \mathbf{x}_{0|t}$; $\quad \mathbf{v} \leftarrow \mathcal{A}(\mathbf{x})$; $\quad \mathbf{u} \leftarrow \mathbf{0}$
5:     **for** $k = 1$ **to** K **do**
6:         $\mathbf{b} \leftarrow \mathbf{v} - \mathbf{u}$; $\quad F(\mathbf{x}) \leftarrow \frac{1}{2\gamma_t}\|\mathbf{x} - \mathbf{x}_{0|t}\|^2 + \frac{\rho}{2}\|\mathcal{A}(\mathbf{x}) - \mathbf{b}\|^2$
7:         **for** $s = 1$ **to** S **do** ▷ x-update: gradient step + backtracking
8:             $\mathbf{r} \leftarrow \mathcal{A}(\mathbf{x}) - \mathbf{b}$; $\mathbf{s} \leftarrow \mathbf{x} - \mathbf{x}_{0|t}$
9:             $\mathbf{g}_{\text{data}} \leftarrow \nabla_{\mathbf{x}}\left(\frac{1}{2}\|\mathcal{A}(\mathbf{x}) - \mathbf{b}\|^2\right)$ ▷ via automatic differentiation
10:             $\mathbf{g} \leftarrow \frac{1}{\gamma_t}\mathbf{s} + \rho\, \mathbf{g}_{\text{data}}$; $\Delta\mathcal{A} \leftarrow \mathcal{A}(\mathbf{x} + \eta\mathbf{g}) - \mathcal{A}(\mathbf{x})$
11:             $\alpha \leftarrow \dfrac{\eta^2 \frac{1}{\gamma_t}\langle \mathbf{s}, \mathbf{g}\rangle + \eta\,\rho\,\langle \mathbf{r}, \Delta\mathcal{A}\rangle}{\eta^2 \frac{1}{\gamma_t}\|\mathbf{g}\|^2 + \rho\,\|\Delta\mathcal{A}\|^2}$
12:             Backtrack on α until $F(\mathbf{x} - \alpha\mathbf{g}) < F(\mathbf{x})$; set $\mathbf{x} \leftarrow \mathbf{x} - \alpha\mathbf{g}$
13:         **end for**
14:         $\mathbf{w} \leftarrow \mathcal{A}(\mathbf{x}) + \mathbf{u}$; $\quad \mathbf{v} \leftarrow \Pi_{\|\cdot - \mathbf{y}\| \le \epsilon}(\mathbf{w})$
15:         $\mathbf{u} \leftarrow \mathbf{u} + \mathcal{A}(\mathbf{x}) - \mathbf{v}$
16:     **end for**
17:     Sample $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, I)$ and set $\mathbf{x}_{t-1} \leftarrow \mathbf{x} + \sigma_{t-1}\,\boldsymbol{\xi}$
18: **end for**
19: **return** $\mathbf{x}_0$

Algorithm 2 FAST-DIPS in Latent Space

Require: measurement $\mathbf{y}$; schedule $\{\sigma_t\}$; latent denoiser $\mathbf{z}_{\text{den}}(\cdot, \sigma_t)$; encoder $\mathcal{E}$; decoder $\mathcal{D}$; forward $\mathcal{A}$; parameters $\rho_x, \gamma_x, K_x, S_x, \varepsilon_x,\ \rho_z, \gamma_z, K_z, S_z, \varepsilon_z,\ \sigma_{\text{switch}}$
Ensure: reconstructed image $\mathbf{x}_0$

Sample $\mathbf{z}_T \sim \mathcal{N}(\mathbf{0}, \sigma_T^2 I)$
for $t = T$ **down to** 1 **do**
  predict (latent) $\mathbf{z}_{0|t} \leftarrow \mathbf{z}_{\text{den}}(\mathbf{z}_t, \sigma_t)$
  if $\sigma_t > \sigma_{\text{switch}}$ **then** ▷ early: pixel correction
    $\mathbf{x}_{0|t} \leftarrow \mathcal{D}(\mathbf{z}_{0|t})$; $\mathbf{x} \leftarrow \mathbf{x}_{0|t}$; $\mathbf{v} \leftarrow \mathcal{A}(\mathbf{x})$; $\mathbf{u} \leftarrow \mathbf{0}$
    for $k = 1$ **to** K_x **do**
      $\mathbf{b} \leftarrow \mathbf{v} - \mathbf{u}$; $F_x(\mathbf{x}) \leftarrow \frac{1}{2\gamma_x}\|\mathbf{x} - \mathbf{x}_{0|t}\|^2 + \frac{\rho_x}{2}\|\mathcal{A}(\mathbf{x}) - \mathbf{b}\|^2$
      for $s = 1$ **to** S_x **do** ▷ $\mathbf{x}$-update with analytic step
        $\mathbf{g} \leftarrow \frac{1}{\gamma_x}(\mathbf{x} - \mathbf{x}_{0|t}) + \rho_x J_{\mathcal{A}}(\mathbf{x})^\top(\mathcal{A}(\mathbf{x}) - \mathbf{b})$
        Form $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}$ (JVP) and set α by Equation 22;
        Backtrack on α until $F_x(\mathbf{x} - \alpha\mathbf{g}) < F(\mathbf{x})$; set $\mathbf{x} \leftarrow \mathbf{x} - \alpha\mathbf{g}$
      end for
      $\mathbf{w} \leftarrow \mathcal{A}(\mathbf{x}) + \mathbf{u}$; $\mathbf{v} \leftarrow \Pi_{\|\cdot - \mathbf{y}\| \le \varepsilon_x}(\mathbf{w})$; $\mathbf{u} \leftarrow \mathbf{u} + \mathcal{A}(\mathbf{x}) - \mathbf{v}$
    end for
    re-encode $\mathbf{z} \leftarrow \mathcal{E}(\mathbf{x})$
  else ▷ late: latent correction
    $\mathbf{z} \leftarrow \mathbf{z}_{0|t}$; $\mathbf{v} \leftarrow \mathcal{A}(\mathcal{D}(\mathbf{z}))$; $\mathbf{u} \leftarrow \mathbf{0}$
    for $k = 1$ **to** K_z **do**
      $\mathbf{b} \leftarrow \mathbf{v} - \mathbf{u}$; $F_z(\mathbf{z}) \leftarrow \frac{1}{2\gamma_z}\|\mathbf{z} - \mathbf{z}_{0|t}\|^2 + \frac{\rho_z}{2}\|\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b}\|^2$
      for $s = 1$ **to** S_z **do** ▷ $\mathbf{z}$-update with analytic step
        $\mathbf{g}_z \leftarrow \frac{1}{\gamma_z}(\mathbf{z} - \mathbf{z}_{0|t}) + \rho_z J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})^\top(\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b})$
        Form $J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z$ (JVP) and set α by Equation 36;
        Backtrack on α until $F_z(\mathbf{z} - \alpha\mathbf{g}_z) < F_z(\mathbf{z})$; set $\mathbf{z} \leftarrow \mathbf{z} - \alpha\mathbf{g}_z$
      end for
      $\mathbf{w} \leftarrow \mathcal{A}(\mathcal{D}(\mathbf{z})) + \mathbf{u}$; $\mathbf{v} \leftarrow \Pi_{\|\cdot - \mathbf{y}\| \le \varepsilon_z}(\mathbf{w})$; $\mathbf{u} \leftarrow \mathbf{u} + \mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{v}$
    end for
  end if
  re-anneal $\mathbf{z}_{t-1} \leftarrow \mathbf{z} + \sigma_{t-1}\boldsymbol{\xi}$, $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, I)$
end for
return $\mathbf{x}_0 \leftarrow \mathcal{D}(\mathbf{z}_0)$

A.3 Theory and Proofs

This appendix first summarizes the proposed FAST-DIPS procedure and its modeling assumptions (App. A.3.1). We then restate the key propositions/remarks from the main text and provide detailed proofs (App. A.3.2–A.3.3). Finally, we give step-by-step derivations of the analytic step sizes used in the pixel and latent updates and explain how they can be computed with autodiff VJP/JVP or a single forward-difference probe (App. A.3.4).

A.3.1 Overview and assumptions

Method in one paragraph. At diffusion level t, the pretrained denoiser returns an anchor $\mathbf{x}_{0|t} = \mathbf{x}_{\text{den}}(\mathbf{x}_t, \sigma_t)$. We then solve a *hard-constrained proximal* problem around $\mathbf{x}_{0|t}$,

$$\min_{\mathbf{x} \in \mathbb{R}^{CHW}} \frac{1}{2\gamma_t}\|\mathbf{x} - \mathbf{x}_{0|t}\|^2 \quad \text{s.t.} \quad \|\mathcal{A}(\mathbf{x}) - \mathbf{y}\| \le \varepsilon, \tag{39}$$

in the standard (Euclidean) measurement space. We solve Equation 39 by scaled ADMM with variables $(\mathbf{x}, \mathbf{v}, \mathbf{u})$:

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \frac{1}{2\gamma_t}\|\mathbf{x} - \mathbf{x}_{0|t}\|^2 + \frac{\rho}{2}\|\mathcal{A}(\mathbf{x}) - \mathbf{v}^k + \mathbf{u}^k\|^2, \tag{40}$$

$$\mathbf{v}^{k+1} = \Pi_{\mathcal{C}}\big(\mathcal{A}(\mathbf{x}^{k+1}) + \mathbf{u}^k\big), \quad \mathcal{C} = \{\mathbf{v} : \|\mathbf{v} - \mathbf{y}\| \le \varepsilon\}, \tag{41}$$

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \mathcal{A}(\mathbf{x}^{k+1}) - \mathbf{v}^{k+1}. \tag{42}$$

The $\mathbf{v}$-update is a closed-form projection onto a ball; the $\mathbf{x}$-update is one (or a few) *adjoint-free* gradient steps with an *analytic, model-optimal* step size, where the needed directional Jacobian term $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}$ is obtained either by autodiff JVP or by a *single forward-difference probe*. After correction, we *re-anneal* by sampling

$$\mathbf{x}_{t-1} = \mathbf{x}_t^{\mathrm{corr}} + \sigma_{t-1}\,\boldsymbol{\xi}, \qquad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, I), \tag{43}$$

which implements the decoupled time–marginal transport.

Standing assumptions.

- **A1** (Noise model and metric) We assume additive white Gaussian noise (AWGN) with covariance $\beta^2 I$ and work in the standard Euclidean metric in measurement space; the feasibility set is the ball $\{\mathbf{v} : \|\mathbf{v} - \mathbf{y}\| \le \varepsilon\}$.
- **A2** (Regularity) $\mathcal{A}$ is C^1 in a neighborhood of the iterates, and $J_{\mathcal{A}}$ is locally Lipschitz.
- **A3** (Feasibility) The credible-set radius ε is chosen so that the ground-truth measurement is feasible: $\|\mathcal{A}(\mathbf{x}_0) - \mathbf{y}\| \le \varepsilon$.

A.3.2 Pixel-space propositions and proofs

We restate the pixel-space results from the main text and provide detailed proofs.

Proposition 2 (Closed-form projection onto the credible set)**.** *Let $\mathcal{C} = \{\mathbf{v} \in \mathbb{R}^m : \|\mathbf{v} - \mathbf{y}\| \le \varepsilon\}$ in the measurement space. Then the Euclidean projection $\Pi_{\mathcal{C}}(\mathbf{w})$ in Equation 15 is exactly the radial shrink (Parikh & Boyd (2014))*

$$\Pi_{\mathcal{C}}(\mathbf{w}) \;=\; \begin{cases} \mathbf{w}, & \|\mathbf{w} - \mathbf{y}\| \le \varepsilon, \\ \mathbf{y} + \varepsilon\,\dfrac{\mathbf{w} - \mathbf{y}}{\|\mathbf{w} - \mathbf{y}\|}, & \textit{otherwise.} \end{cases} \tag{17}$$

Proof of Proposition 2. We solve $\min_{\mathbf{v}} \frac{1}{2}\|\mathbf{v} - \mathbf{w}\|^2$ s.t. $\|\mathbf{v} - \mathbf{y}\| \le \varepsilon$. The objective is 1-strongly convex and the feasible set is closed and convex; hence there is a unique minimizer.

KKT derivation. The Lagrangian is

$$\mathcal{L}(\mathbf{v}, \lambda) = \tfrac{1}{2}\|\mathbf{v} - \mathbf{w}\|^2 + \lambda\big(\|\mathbf{v} - \mathbf{y}\| - \varepsilon\big), \qquad \lambda \ge 0.$$

Stationarity gives

$$\mathbf{0} = \nabla_{\mathbf{v}}\mathcal{L}(\mathbf{v}, \lambda) = (\mathbf{v} - \mathbf{w}) + \lambda\,\frac{\mathbf{v} - \mathbf{y}}{\|\mathbf{v} - \mathbf{y}\|} \quad \text{if } \mathbf{v} \ne \mathbf{y}.$$

There are two cases.

(i) Interior case. If the constraint is inactive at the optimum, then $\lambda = 0$ by complementary slackness and stationarity gives $\mathbf{v} = \mathbf{w}$. Feasibility requires $\|\mathbf{w} - \mathbf{y}\| \le \varepsilon$, i.e., $\mathbf{w} \in \mathcal{C}$.

(ii) Boundary case. Otherwise $\|\mathbf{v} - \mathbf{y}\| = \varepsilon$ and $\lambda > 0$. Stationarity implies $\mathbf{v} - \mathbf{w}$ is colinear with $\mathbf{v} - \mathbf{y}$; hence the optimizer lies on the ray from $\mathbf{y}$ to $\mathbf{w}$. Write $\mathbf{v} = \mathbf{y} + \tau(\mathbf{w} - \mathbf{y})$ with $\tau \ge 0$. Enforcing $\|\mathbf{v} - \mathbf{y}\| = \varepsilon$ yields $\tau = \varepsilon/\|\mathbf{w} - \mathbf{y}\|$. Substituting gives

$$\mathbf{v} = \mathbf{y} + \varepsilon\,\frac{\mathbf{w} - \mathbf{y}}{\|\mathbf{w} - \mathbf{y}\|}.$$

This is exactly the radial projection formula in Equation 17. Uniqueness follows from strong convexity. ☐

Proposition 3 (Local model-optimal step and descent)**.** *Under C^1 regularity of $\mathcal{A}$ near $\mathbf{x}$, $\alpha^\star$ in Equation 22 minimizes the quadratic model Equation 21. Moreover,*

$$F(\mathbf{x}-\alpha^\star\mathbf{g}) \;\le\; F(\mathbf{x}) \;-\; \frac{\big(\frac{1}{\gamma_t}\langle \mathbf{s},\mathbf{g}\rangle + \rho\langle \mathbf{r}, J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\rangle\big)^2}{2\big(\frac{1}{\gamma_t}\|\mathbf{g}\|^2 + \rho\|J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2\big)} \;+\; O(\|\mathbf{g}\|^3), \tag{23}$$

and the backtracking line search (Armijo (1966)) guarantees monotone decrease of F even when Equation 20 is only a local approximation.

Proof of Proposition 3. Write $F(\mathbf{x}) = \frac{1}{2\gamma_t}\|\mathbf{s}\|^2 + \frac{\rho}{2}\|\mathbf{r}\|^2$ with $\mathbf{s} = \mathbf{x} - \mathbf{x}_{0|t}$ and $\mathbf{r} = \mathcal{A}(\mathbf{x}) - \mathbf{b}$. The gradient is

$$\mathbf{g} = \nabla F(\mathbf{x}) = \tfrac{1}{\gamma_t}\mathbf{s} + \rho\, J_{\mathcal{A}}(\mathbf{x})^\top \mathbf{r}.$$

Consider the steepest-descent trial $\mathbf{x}(\alpha) = \mathbf{x} - \alpha\mathbf{g}$. A first-order Taylor expansion along $-\mathbf{g}$ gives

$$\mathcal{A}\big(\mathbf{x}(\alpha)\big) = \mathcal{A}(\mathbf{x}) - \alpha J_{\mathcal{A}}(\mathbf{x})\mathbf{g} + \mathbf{e}(\alpha), \qquad \|\mathbf{e}(\alpha)\| \le \tfrac{L_{\mathcal{A}}}{2}\alpha^2\|\mathbf{g}\|^2,$$

for some local Lipschitz constant $L_{\mathcal{A}}$ of $J_{\mathcal{A}}$ (from **A2**). Plugging this into $F(\mathbf{x}(\alpha))$ yields

$$F(\mathbf{x}(\alpha)) = \underbrace{\tfrac{1}{2\gamma_t}\|\mathbf{s}-\alpha\mathbf{g}\|^2 + \tfrac{\rho}{2}\|\mathbf{r} - \alpha J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2}_{:=\tilde{F}(\alpha)} + \rho\,\langle \mathbf{r} - \alpha J_{\mathcal{A}}(\mathbf{x})\mathbf{g},\, \mathbf{e}(\alpha)\rangle + \tfrac{\rho}{2}\|\mathbf{e}(\alpha)\|^2.$$

The model $\tilde{F}$ is a convex quadratic in α with derivative

$$\tilde{F}'(\alpha) = -\tfrac{1}{\gamma_t}\langle \mathbf{s},\mathbf{g}\rangle - \rho\,\langle \mathbf{r}, J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\rangle + \alpha\Big(\tfrac{1}{\gamma_t}\|\mathbf{g}\|^2 + \rho\|J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2\Big),$$

and curvature $\tilde{F}''(\alpha) = \frac{1}{\gamma_t}\|\mathbf{g}\|^2 + \rho\|J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2 \ge 0$, with equality only at stationary points where $\mathbf{g} = \mathbf{0}$ and $J_{\mathcal{A}}(\mathbf{x})\mathbf{g} = \mathbf{0}$. Setting $\tilde{F}'(\alpha) = 0$ yields the model minimizer $\alpha^\star$ in Equation 22.

Descent of the true F. Using the expansion above and Cauchy–Schwarz with the bound on $\|\mathbf{e}(\alpha)\|$, we obtain

$$F(\mathbf{x}-\alpha\mathbf{g}) \le \tilde{F}(\alpha) + \rho\|\mathbf{r} - \alpha J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|\;\tfrac{L_{\mathcal{A}}}{2}\alpha^2\|\mathbf{g}\|^2 + \tfrac{\rho}{2}\Big(\tfrac{L_{\mathcal{A}}}{2}\alpha^2\|\mathbf{g}\|^2\Big)^2.$$

At $\alpha = \alpha^\star$, $\tilde{F}(\alpha^\star) = \min_\alpha \tilde{F}(\alpha)$ and the improvement over $\tilde{F}(0) = F(\mathbf{x})$ is

$$\tilde{F}(0) - \tilde{F}(\alpha^\star) = \frac{\big(\frac{1}{\gamma_t}\langle \mathbf{s},\mathbf{g}\rangle + \rho\langle \mathbf{r}, J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\rangle\big)^2}{2\big(\frac{1}{\gamma_t}\|\mathbf{g}\|^2 + \rho\|J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2\big)}.$$

The remainder terms are $O(\alpha^{*2}\|\mathbf{g}\|^2)$ and $O(\alpha^{*4}\|\mathbf{g}\|^4)$; shrinking α by a constant factor (standard Armijo backtracking) ensures these are dominated by the quadratic-model decrease, yielding strict descent of F. □

Remark 2 (Step size from finite-difference JVP)**.** *Replacing $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}$ in Equation 22 by $\Delta\mathcal{A}/\eta$ from Equation 24 yields the numerically stable single-forward-call step*

$$\alpha_{\mathrm{FD}} \;=\; \frac{\eta^2\,\frac{1}{\gamma_t}\langle \mathbf{s},\,\mathbf{g}\rangle \;+\; \eta\,\rho\,\langle \mathbf{r},\,\Delta\mathcal{A}\rangle}{\eta^2\,\frac{1}{\gamma_t}\|\mathbf{g}\|^2 \;+\; \rho\,\|\Delta\mathcal{A}\|^2} \qquad \textit{where} \quad \Delta\mathcal{A} = \mathcal{A}(\mathbf{x}+\eta\mathbf{g}) - \mathcal{A}(\mathbf{x}). \tag{25}$$

which is algebraically equivalent to substituting $J_{\mathcal{A}}(\mathbf{x})\mathbf{g} \approx \Delta\mathcal{A}/\eta$ in Equation 22 (the scaling by η^2 avoids division by small η). Since $J_{\mathcal{A}}(\mathbf{x})\mathbf{g} = \Delta\mathcal{A}/\eta + O(\eta)$, we have $\alpha_{\mathrm{FD}} = \alpha^\star + O(\eta)$; Armijo backtracking then preserves monotone decrease of the true F.

Remark 1 (Linear $\mathcal{A}$ yields exact optimal line search)**.** *If $\mathcal{A}$ is linear, then Equation 20 is exact and Equation 22 gives the true optimal line-search step for F along $-\mathbf{g}$ (Nocedal & Wright (2006)), delivering the fastest progress among steepest-descent steps.*

Justification. If $\mathcal{A}(\mathbf{x}) = H\mathbf{x}$, then $J_{\mathcal{A}}(\mathbf{x}) = H$ and the linearization is exact: $\mathcal{A}(\mathbf{x} - \alpha\mathbf{g}) = \mathcal{A}(\mathbf{x}) - \alpha H\mathbf{g}$. Hence $\tilde{F}$ coincides with $F(\mathbf{x} - \alpha\mathbf{g})$ along the line, and the model minimizer in Equation 22 is the exact optimal line-search step.

Proposition 4 (Fixed points satisfy KKT for Equation 9)**.** *Let $(\mathbf{x}^*, \mathbf{v}^*, \mathbf{u}^*)$ be a fixed point of Equation 14–Equation 16. Then $\mathcal{A}(\mathbf{x}^*) = \mathbf{v}^*$, $\mathbf{v}^* \in \mathcal{C}$, and there exists $\lambda^* \geq 0$ such that*

$$\frac{1}{\gamma_t}(\mathbf{x}^* - \mathbf{x}_{0|t}) \; + \; \lambda^* J_{\mathcal{A}}(\mathbf{x}^*)^\top \boldsymbol{\nu}^* \; = \; 0, \qquad \lambda^*\big(\|\mathcal{A}(\mathbf{x}^*) - \mathbf{y}\| - \varepsilon\big) \; = \; 0, \tag{26}$$

where

$$\boldsymbol{\nu}^* \in \begin{cases} \left\{ \dfrac{\mathcal{A}(\mathbf{x}^*) - \mathbf{y}}{\|\mathcal{A}(\mathbf{x}^*) - \mathbf{y}\|} \right\}, & \|\mathcal{A}(\mathbf{x}^*) - \mathbf{y}\| = \varepsilon, \\ \{\mathbf{0}\}, & \|\mathcal{A}(\mathbf{x}^*) - \mathbf{y}\| < \varepsilon. \end{cases}$$

Hence $\mathbf{x}^*$ satisfies the KKT conditions of Equation 9 (Bertsekas (1999)).

Proof of Proposition 4. At a fixed point $(\mathbf{x}^*, \mathbf{v}^*, \mathbf{u}^*)$, the $\mathbf{u}$-update satisfies $\mathbf{u}^* = \mathbf{u}^* + \mathcal{A}(\mathbf{x}^*) - \mathbf{v}^*$, hence primal feasibility $\mathcal{A}(\mathbf{x}^*) - \mathbf{v}^* = \mathbf{0}$. The $\mathbf{v}$-update is the metric projection onto $\mathcal{C}$:

$$\mathbf{v}^* = \Pi_{\mathcal{C}}(\mathcal{A}(\mathbf{x}^*) + \mathbf{u}^*),$$

so $\mathbf{v}^* \in \mathcal{C}$ and the optimality condition of the projection reads

$$\mathbf{0} \in \partial\iota_{\mathcal{C}}(\mathbf{v}^*) + \rho\big(\mathbf{v}^* - (\mathcal{A}(\mathbf{x}^*) + \mathbf{u}^*)\big) = \partial\iota_{\mathcal{C}}(\mathbf{v}^*) - \rho\,\mathbf{u}^*,$$

i.e., $\rho\,\mathbf{u}^* \in \partial\iota_{\mathcal{C}}(\mathbf{v}^*) = N_{\mathcal{C}}(\mathbf{v}^*)$, the normal cone of $\mathcal{C}$ at $\mathbf{v}^*$. For the $\mathbf{x}$-subproblem, first-order optimality gives

$$\mathbf{0} = \tfrac{1}{\gamma_t}(\mathbf{x}^* - \mathbf{x}_{0|t}) + \rho\, J_{\mathcal{A}}(\mathbf{x}^*)^\top\big(\mathcal{A}(\mathbf{x}^*) - \mathbf{v}^* + \mathbf{u}^*\big) = \tfrac{1}{\gamma_t}(\mathbf{x}^* - \mathbf{x}_{0|t}) + \rho\, J_{\mathcal{A}}(\mathbf{x}^*)^\top\mathbf{u}^*,$$

using primal feasibility. The normal cone for the ball $\mathcal{C} = \{\mathbf{v} : \|\mathbf{v} - \mathbf{y}\| \leq \varepsilon\}$ is

$$N_{\mathcal{C}}(\mathbf{v}^*) = \begin{cases} \{\lambda\boldsymbol{\nu}^* : \lambda \geq 0\}, & \|\mathbf{v}^* - \mathbf{y}\| = \varepsilon, \\ \{\mathbf{0}\}, & \|\mathbf{v}^* - \mathbf{y}\| < \varepsilon, \end{cases} \quad \text{with} \quad \boldsymbol{\nu}^* = \frac{\mathbf{v}^* - \mathbf{y}}{\|\mathbf{v}^* - \mathbf{y}\|}.$$

Thus $\rho\,\mathbf{u}^* = \lambda^*\boldsymbol{\nu}^*$ for some $\lambda^* \geq 0$ when the constraint is active and $\mathbf{u}^* = \mathbf{0}$ otherwise. Substituting into the $\mathbf{x}$-optimality condition yields

$$\tfrac{1}{\gamma_t}(\mathbf{x}^* - \mathbf{x}_{0|t}) + \lambda^* J_{\mathcal{A}}(\mathbf{x}^*)^\top\boldsymbol{\nu}^* = \mathbf{0}.$$

Complementarity $\lambda^*(\|\mathcal{A}(\mathbf{x}^*) - \mathbf{y}\| - \varepsilon) = 0$ follows by construction of the normal cone. Hence $(\mathbf{x}^*, \lambda^*)$ satisfies the KKT conditions of Equation 39. ☐

Proposition 1 (Mode-substitution error under Laplace)**.** *Assume locally $p(\mathbf{x}_0 \mid \mathbf{x}_t, \mathbf{y}) \approx \mathcal{N}(\boldsymbol{m}_t, \Sigma_t)$ and let $\mathbf{x}_t^{\mathrm{corr}}$ solve Equation 9. Then the KL divergence between the time–marginals obtained by (i) injecting noise from $\mathcal{N}(\boldsymbol{m}_t, \Sigma_t)$ and (ii) injecting noise centered at $\mathbf{x}_{0|t}^{\mathrm{corr}}$ is bounded by*

$$\mathrm{KL}\Big(\mathcal{N}(\boldsymbol{m}_t, \Sigma_t + \sigma_{t-1}^2 I) \,\big\|\, \mathcal{N}(\mathbf{x}_{0|t}^{\mathrm{corr}}, \sigma_{t-1}^2 I)\Big) \;\leq\; \frac{\|\boldsymbol{m}_t - \mathbf{x}_{0|t}^{\mathrm{corr}}\|^2}{2\sigma_{t-1}^2} + \frac{\|\Sigma_t\|_F^2}{4\sigma_{t-1}^4}. \tag{12}$$

Consequences. *The bound is small (i) early, when σ_{t-1}^2 is large, and (ii) late, when $\|\Sigma_t\|$ is small; this justifies the decoupled rule Equation 11.*

Proof of Proposition 1. Let $P = \mathcal{N}(\boldsymbol{m}_t, \Sigma_t + \sigma^2 I)$ and $Q = \mathcal{N}(\mathbf{x}_t^{\mathrm{corr}}, \sigma^2 I)$ in $\mathbb{R}^d$. The Gaussian KL formula gives

$$\mathrm{KL}(P\|Q) = \tfrac{1}{2}\left(\mathrm{tr}(\Sigma_Q^{-1}\Sigma_P) + (\boldsymbol{\mu}_Q - \boldsymbol{\mu}_P)^\top\Sigma_Q^{-1}(\boldsymbol{\mu}_Q - \boldsymbol{\mu}_P) - d + \log\frac{\det\Sigma_Q}{\det\Sigma_P}\right).$$

With $\Sigma_Q = \sigma^2 I$, $\Sigma_P = \sigma^2 I + \Sigma_t$, $\boldsymbol{\mu}_Q - \boldsymbol{\mu}_P = \mathbf{x}_t^{\mathrm{corr}} - \boldsymbol{m}_t$, we get

$$\mathrm{KL}(P\|Q) = \frac{\|\mathbf{x}_t^{\mathrm{corr}} - \boldsymbol{m}_t\|^2}{2\sigma^2} + \frac{1}{2}\big(\mathrm{tr}(I + \tfrac{1}{\sigma^2}\Sigma_t) - d - \log\det(I + \tfrac{1}{\sigma^2}\Sigma_t)\big).$$

Diagonalize $\Sigma_t = U\Lambda U^\top$ with eigenvalues $\lambda_i \geq 0$. Then

$$\mathrm{KL}(P\|Q) = \frac{\|\mathbf{x}_t^{\mathrm{corr}} - \boldsymbol{m}_t\|^2}{2\sigma^2} + \frac{1}{2}\sum_{i=1}^{d}\left(\frac{\lambda_i}{\sigma^2} - \log\Big(1 + \frac{\lambda_i}{\sigma^2}\Big)\right).$$

Use $x - \log(1+x) \leq x^2/2$ for $x \geq 0$ termwise to obtain

$$\mathrm{KL}(P\|Q) \leq \frac{\|\mathbf{x}_t^{\mathrm{corr}} - \boldsymbol{m}_t\|^2}{2\sigma^2} + \frac{1}{4}\sum_{i=1}^{d}\frac{\lambda_i^2}{\sigma^4} = \frac{\|\mathbf{x}_t^{\mathrm{corr}} - \boldsymbol{m}_t\|^2}{2\sigma^2} + \frac{\|\Sigma_t\|_F^2}{4\sigma^4}.$$

Tightness regimes. The second term vanishes as $\sigma^2 \to \infty$ (early in the schedule) and as $\|\Sigma_t\|_F \to 0$ (late in the schedule); the first term quantifies bias between the mode $\mathbf{x}_t^{\mathrm{corr}}$ and the posterior mean $\boldsymbol{m}_t$. ☐

A.3.3 Latent-space counterparts and proofs

Why the substitution $\mathcal{A} \mapsto \mathcal{A}\circ\mathcal{D}$ is valid. If $\mathcal{A}$ and the decoder $\mathcal{D}$ are C^1, then so is the composite $\mathcal{A}\circ\mathcal{D}$. All arguments that relied on VJP/JVP of $\mathcal{A}$ and local linearization transfer verbatim to $\mathcal{A}\circ\mathcal{D}$ via the chain rule; the projection remains in *measurement* space and is unchanged.

Proposition 5 (Local model-optimal step and descent in latent space)**.** *Under C^1 regularity of $\mathcal{A}\circ\mathcal{D}$ near $\mathbf{z}$, the step Equation 36 minimizes the quadratic model $\tilde{F}_z(\alpha)$ and*

$$F_z(\mathbf{z} - \alpha_z^*\mathbf{g}_z) \;\leq\; F_z(\mathbf{z}) \;-\; \frac{\big(\frac{1}{\gamma_z}\langle \mathbf{s}_z, \mathbf{g}_z\rangle + \rho_z\langle \mathbf{r}, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z\rangle\big)^2}{2\big(\frac{1}{\gamma_z}\|\mathbf{g}_z\|^2 + \rho_z\|J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z\|^2\big)} \;+\; O(\|\mathbf{g}_z\|^3), \tag{37}$$

with monotone decrease ensured by backtracking.

Proof of Proposition 5. Define $F_z(\mathbf{z}) = \frac{1}{2\gamma_z}\|\mathbf{z} - \mathbf{z}_{0|t}\|^2 + \frac{\rho_z}{2}\|\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b}\|^2$ and $\mathbf{g}_z = \frac{1}{\gamma_z}(\mathbf{z} - \mathbf{z}_{0|t}) + \rho_z\, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})^\top(\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b})$. Linearize $\mathcal{A}(\mathcal{D}(\mathbf{z} - \alpha\mathbf{g}_z)) = \mathcal{A}(\mathcal{D}(\mathbf{z})) - \alpha J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z + \mathbf{e}_z(\alpha)$ with $\|\mathbf{e}_z(\alpha)\| \leq \frac{L_{\mathcal{A}\circ\mathcal{D}}2}{\alpha}^2\|\mathbf{g}_z\|^2$. Repeat the pixel-space proof with $\mathcal{A}$ replaced by $\mathcal{A}\circ\mathcal{D}$ to obtain the model minimizer Equation 36 and the same Armijo descent guarantee. ☐

Proposition 6 (KKT at latent ADMM fixed points)**.** *If $(\mathbf{z}^*, \mathbf{v}^*, \mathbf{u}^*)$ is a fixed point of Equation 30–Equation 32, then $\mathcal{A}(\mathcal{D}(\mathbf{z}^*)) = \mathbf{v}^* \in \mathcal{C}$ and there exists $\lambda^* \geq 0$ such that*

$$\frac{1}{\gamma_z}(\mathbf{z}^* - \mathbf{z}_{0|t}) \;+\; \lambda^*\, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z}^*)^\top\boldsymbol{\nu}^* \;=\; \mathbf{0}, \qquad \lambda^*\big(\|\mathcal{A}(\mathcal{D}(\mathbf{z}^*)) - \mathbf{y}\| - \varepsilon_z\big) = 0, \tag{38}$$

with $\boldsymbol{\nu}^* = (\mathcal{A}(\mathcal{D}(\mathbf{z}^*)) - \mathbf{y})/\|\mathcal{A}(\mathcal{D}(\mathbf{z}^*)) - \mathbf{y}\|$ when the constraint is active and $\boldsymbol{\nu}^* = \mathbf{0}$ otherwise.

Proof of Proposition 6. Identical to the pixel-space KKT proof, replacing $\mathcal{A}$ by $\mathcal{A}\circ\mathcal{D}$ and $\mathbf{x}$ by $\mathbf{z}$. The projection onto $\mathcal{C}$ is unchanged; the normal cone and complementarity conditions are therefore the same, yielding the stated KKT system. ☐

Remark 6 (Mode-substitution transport in latent space)**.** *Replacing $p(\mathbf{z}_0 \mid \mathbf{z}_t, \mathbf{y})$ by its mode and re-annealing with $\mathbf{z}_{t-1} = \mathbf{z}_t^{\mathrm{corr}} + \sigma_{t-1}\boldsymbol{\xi}$ induces the same KL structure as Prop. 1 after decoding because noise injection and credibility act in* measurement *space; only the mean is mapped by $\mathcal{D}$.*

A.3.4 Derivation of analytic step sizes and autodiff computation

Pixel space: detailed derivation. Recall

$$F(\mathbf{x}) = \frac{1}{2\gamma_t}\|\mathbf{x} - \mathbf{x}_{0|t}\|^2 + \frac{\rho}{2}\|\mathcal{A}(\mathbf{x}) - \mathbf{b}\|^2, \quad \mathbf{s} = \mathbf{x} - \mathbf{x}_{0|t}, \quad \mathbf{r} = \mathcal{A}(\mathbf{x}) - \mathbf{b}.$$

Then $\mathbf{g} = \frac{1}{\gamma_t}\mathbf{s} + \rho\, J_{\mathcal{A}}(\mathbf{x})^\top\mathbf{r}$. For the trial $\mathbf{x}(\alpha) = \mathbf{x} - \alpha\mathbf{g}$,

$$\mathcal{A}(\mathbf{x}(\alpha)) \approx \mathcal{A}(\mathbf{x}) - \alpha J_{\mathcal{A}}(\mathbf{x})\mathbf{g}$$

gives the scalar quadratic model

$$\tilde{F}(\alpha) = \frac{1}{2\gamma_t}\|\mathbf{s} - \alpha\mathbf{g}\|^2 + \frac{\rho}{2}\|\mathbf{r} - \alpha J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2,$$

whose derivative is

$$\tilde{F}'(\alpha) = -\tfrac{1}{\gamma_t}\langle \mathbf{s}, \mathbf{g}\rangle - \rho\langle \mathbf{r}, J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\rangle + \alpha\Big(\tfrac{1}{\gamma_t}\|\mathbf{g}\|^2 + \rho\|J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2\Big).$$

Setting $\tilde{F}'(\alpha) = 0$ yields $\alpha^\star$ in Equation 22. The curvature $\tilde{F}''(\alpha) = \frac{1}{\gamma_t}\|\mathbf{g}\|^2 + \rho\|J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2 \geq 0$ shows uniqueness unless $\mathbf{g} = \mathbf{0}$.

Autodiff computation recipe (pixel):

1. Evaluate $\mathcal{A}(\mathbf{x})$ to get $\mathbf{r} = \mathcal{A}(\mathbf{x})) - \mathbf{b}$.
2. Compute the VJP $J_{\mathcal{A}}(\mathbf{x})^\top \mathbf{r}$ (reverse-mode autodiff) and form $\mathbf{g}$.
3. Obtain the directional Jacobian $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}$ either
   - by forward-mode autodiff (preferred when available), or
   - by few forward-difference probe

     $$J_{\mathcal{A}}(\mathbf{x})\mathbf{g} \approx \frac{\Delta\mathcal{A}}{\eta}, \qquad \Delta\mathcal{A} := \mathcal{A}(\mathbf{x} + \eta\mathbf{g}) - \mathcal{A}(\mathbf{x}),\;\; \eta \in (10^{-4}, 10^{-2}],$$

     in which case it is numerically convenient to assemble the FD-stabilized closed form Equation 25 (equivalent to substituting $\Delta\mathcal{A}/\eta$ into Equation 22 but avoiding division by small η).
4. Assemble the numerator/denominator, clamp $\alpha \leftarrow \max(0, \alpha^\star)$, and perform Armijo backtracking.

Latent space: detailed derivation. With

$$F_z(\mathbf{z}) = \frac{1}{2\gamma_z}\|\mathbf{z} - \mathbf{z}_{0|t}\|^2 + \frac{\rho_z}{2}\|\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b}\|^2, \quad \mathbf{g}_z = \frac{1}{\gamma_z}(\mathbf{z} - \mathbf{z}_{0|t}) + \rho_z\, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})^\top(\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b}),$$

linearize $\mathcal{A}\circ\mathcal{D}$ to obtain

$$\tilde{F}_z(\alpha) = \frac{1}{2\gamma_z}\|\mathbf{z} - \mathbf{z}_{0|t} - \alpha\mathbf{g}_z\|^2 + \frac{\rho_z}{2}\|\mathcal{A}(\mathcal{D}(\mathbf{z})) - \mathbf{b} - \alpha\, J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z\|^2,$$

whose minimizer is Equation 36. The VJP/JVP are computed end-to-end through $\mathcal{D}$ and $\mathcal{A}$ by autodiff; a single finite-difference through the composition is a valid JVP fallback:

$$J_{\mathcal{A}\circ\mathcal{D}}(\mathbf{z})\mathbf{g}_z \approx \frac{\mathcal{A}(\mathcal{D}(\mathbf{z} + \delta\mathbf{g}_z)) - \mathcal{A}(\mathcal{D}(\mathbf{z}))}{\delta}.$$

Complex-valued measurements. When measurements are complex, we work with real–imaginary stacking (dimension $2m$) and the Euclidean norm; all expressions remain valid verbatim, with $J_{\mathcal{A}}$ denoting the real Jacobian.

Backtracking and safeguards. We use the Armijo condition

$$F(\mathbf{x} - \alpha\mathbf{g}) \leq F(\mathbf{x}) - c\,\alpha\|\mathbf{g}\|^2, \qquad c \in (0, 1),$$

reducing $\alpha \leftarrow \tau\alpha$ (e.g., $\tau = \frac{1}{2}$) until acceptance. If $\alpha^\star \leq 0$, initialize with

$$\alpha_0 = \frac{\|\mathbf{g}\|^2}{\|\mathbf{g}\|^2/\gamma_t + \|J_{\mathcal{A}}(\mathbf{x})\mathbf{g}\|^2/\rho}$$

and backtrack. Identical safeguards apply in latent space with $(\gamma_z, \rho_z, \mathbf{g}_z)$.

A.3.5 ADDITIONAL REMARKS

Remark 7 (Trust-region scaling along the schedule)**.** *Setting $\gamma_t = \sigma_t^2$ ties the proximal radius to the diffusion noise: large exploratory moves are allowed early (large σ_t), while the anchor tightens late, mirroring increasing prior certainty.*

Remark 8 (Feasibility and whitening in implementation)**.** *Under the AWGN setting adopted throughout, the measurement-space credible set is a Euclidean ball and the projection is the closed-form radial shrink of Equation 17; all ADMM updates are therefore standard and closed-form.*

Remark 9 (Empirical choice: FD in pixel, JVP in latent)**.** *The latent formulation includes a decoder–forward stack, making it more complex than in pixel space. Accordingly, in pixel space we use the forward-difference variant Equation 25, which replaces one JVP with a single extra forward call to $\mathcal{A}$ and solves the subproblem faster and more efficiently. By contrast, in latent space we rely on the autodiff JVP for greater stability. In both cases, Armijo backtracking guarantees descent of F.*

A.4 ABLATION STUDIES

Goal and tasks. We assess the impact of measurement-space feasibility via projection and the choice of step size inside the $\mathbf{x}$-update. Experiments use 10 FFHQ images on two representatives: Gaussian blur in pixel space and HDR in latent space, with PSNR/SSIM/LPIPS and average per-image runtime.

Baseline and objective. To isolate projection, we evaluate an unsplit penalized baseline that optimizes the *same* quadratic objective as the x-subproblem inside ADMM, but *without* variable splitting or projection:

$$\min_{\mathbf{x}\in\mathbb{R}^{CHW}} \frac{1}{2\gamma_t}\|\mathbf{x}-\mathbf{x}_{0|t}\|^2 + \frac{1}{2\beta^2}\|\mathcal{A}(\mathbf{x})-\mathbf{y}\|^2,$$

which we refer to as *QDP (no splitting, no proj.)*. In all runs we match the ADMM instantiation by setting $\gamma_t = \sigma_t^2$ identically to FAST-DIPS and choosing the data-penalty weight so that $\frac{\rho}{2} = \frac{1}{2\beta^2}$.

Compute-matched fairness. Each $\mathbf{x}$-gradient step entails one forward pass of A, one VJP, and one JVP (or a single forward probe for FD); projection and dual updates are negligible. With K ADMM iterations and S gradient steps per iteration, *FAST-DIPS (ADMM + proj.)* spends $K \times S$ such triplets per level, so *QDP* is allotted $K \times S$ gradient steps per level to match compute. Step-size mechanisms are kept identical between solvers: constant α, analytic/JVP $\alpha^\star$, and finite-difference α_{FD}.

Findings. In pixel space, α_{FD} is competitive with $\alpha^\star$ at lower cost; in latent space, $\alpha^\star$ provides the stability needed for the nonlinear decoder–forward composition, while α_{FD} lags. Under the matched budget, enforcing feasibility via projection improves quality over the unsplit penalty path; latent runtimes primarily reflect decoder backprop.

A.5 HYPERPARAMETERS OVERVIEW.

Algorithm	Parameter	Linear task Super Resolution 4×	Inpaint (Box)	Inpaint (Random)	Gaussian deblurring	Motion deblurring	Non Linear task Phase retrieval	Nonlinear deblurring	High dynamic range
FAST-DIPS	T	75	75	75	50	50	150	150	150
	K	3	3	3	3	3	2	2	2
	S	1	1	1	2	2	5	5	5
	ρ	200	200	200	200	200	200	200	5
	ε	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
Latent FAST-DIPS	T	50	50	50	50	50	25	25	25
	(K_x, K_z)	(5,5)	(5,5)	(5,5)	(5,5)	(5,5)	(10,10)	(10,10)	(10,10)
	(S_x, S_z)	(3,3)	(3,3)	(3,3)	(3,3)	(3,3)	(3,3)	(3,3)	(3,3)
	(ρ_x, ρ_z)	(200,200)	(200,200)	(200,200)	(200,200)	(200,200)	(200,200)	(200,200)	(200,200)
	σ_{switch}	1	1	1	1	1	5	5	5
	ε	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Table 2: The hyperparameters of experiments in paper for all tasks.

Throughout our experiments, hyperparameter settings are summarized in Table 2. In the annealing process, we set $\sigma_{\text{max}} = 100$ in pixel space and 10 in latent space, with $\sigma_{\text{min}} = 0.1$ in both, to enhance robustness to measurement noise.

A.6 Experimental Details.

Validation set information For reproducibility, we explicitly specify the indices of the samples used for validation. **FFHQ (**256×256**).** We use 100 images corresponding to dataset indices `00000–00099`. **ImageNet (**256×256**).** We use 100 images from the ImageNet validation set corresponding to indices `49000–49099`.

A.7 Baseline Implementation Details.

All baselines were experiments using the authors' public repositories:

- **DAPS/LatentDAPS**: github.com/zhangbingliang2019/DAPS
- **SITCOM**: github.com/sjames40/SITCOM
- **HRDIS**: github.com/deng-ai-lab/HRDIS
- **C-ΠGDM**: github.com/mandt-lab/c-pigdm
- **PSLD**: github.com/LituRout/PSLD
- **ReSample**: github.com/soominkwon/resample

We followed each method's original paper and default repository settings. Additionally, for phase retrieval we applied a best-of-four protocol uniformly across all compared baselines.

Measurements noise setting. Because the SVD operator caused instability when injecting noise in super-resolution and Gaussian deblurring, HRDIS is evaluated with noise on all other tasks, while C-ΠGDM is evaluated only in the noiseless setting for all tasks.

Details of Figure 3. For the runtime–quality trade-off in Figure 3, we varied only the number of solver steps/iterations per method, keeping all other hyperparameters at their recommended defaults:

- **DAPS** The number of ODE steps was fixed at 4, while the number of annealing steps was swept over $\{2, 5, 10, 15, 20, 25\}$.
- **SITCOM** We swept pairs of diffusion steps N and inner iterations K over $(N, K) \in \{(3, 2), (5, 3), (5, 5), (5, 10), (5, 15), (5, 20)\}$.
- **HRDIS** We varied the number of diffusion steps over $\{10, 15, 50, 80, 100, 130\}$.
- **C-ΠGDM** We varied the number of diffusion steps over $\{20, 50, 75, 100, 150, 200\}$.

Automatic Differentiation Primitives. To implement the adjoint-free analytic updates without manually deriving gradients for the forward operator $\mathcal{A}$, we leverage the automatic differentiation capabilities of PyTorch. Specifically, the Vector-Jacobian Product (VJP) term $J_{\mathcal{A}}(\mathbf{x})^\top \mathbf{r}$, which is necessary for computing the gradient of the data-consistency term, is obtained via standard reverse-mode differentiation using `torch.autograd.grad`. For the Jacobian-Vector Product (JVP) term $J_{\mathcal{A}}(\mathbf{x})\mathbf{g}$, we utilize the functional transformation API, specifically `torch.func.jvp`. This allows us to efficiently compute the directional derivative required for the local quadratic model in a fully differentiable manner.

A.8 Additional Experiments

Effectiveness of K ADMM iterations and S gradient steps We performed an ablation study to evaluate the influence of both the number of ADMM iterations (K) and the number of gradient steps (S) in the x-update stage. We tested $K \in \{2, 3, 5\}$ and $S \in \{1, 3, 5\}$ across both linear and nonlinear tasks. As shown in Table 5, increasing K consistently improves performance for both linear tasks (e.g., super-resolution) and nonlinear tasks (e.g., nonlinear deblurring), though naturally at the cost of higher computation. This suggests that the refinement offered by additional ADMM iterations is broadly beneficial regardless of task difficulty.

In contrast, the effect of increasing the number of gradient steps S is task-dependent. For linear tasks, the performance gain is marginal relative to the additional runtime. However, for nonlinear

Solver	Step Size Method	Gaussian Blur (Pixel) PSNR	SSIM	LPIPS	Run-time (s)
QDP (no splitting, no proj.)	constant $\alpha = 10^{-4}$	22.854	0.665	0.429	1.893
	constant $\alpha = 10^{-3}$	28.028	0.796	0.314	1.867
	constant $\alpha = 10^{-2}$	2.687	0.162	0.779	1.955
	JVP	29.480	0.830	0.271	2.356
	FD	29.577	0.832	0.268	2.018
FAST-DIPS (ADMM + proj.)	constant $\alpha = 10^{-4}$	24.829	0.714	0.391	1.988
	constant $\alpha = 10^{-3}$	28.647	0.811	0.296	2.029
	constant $\alpha = 10^{-2}$	3.851	0.151	0.772	1.993
	JVP	29.762	0.829	0.273	2.502
	FD (Ours)	29.632	0.819	0.287	2.053

Solver	Step Size Method	High Dynamic Range (Latent) PSNR	SSIM	LPIPS	Run-time (s)
QDP (no splitting, no proj.)	constant $\alpha = 10^{-5}$	22.113	0.671	0.459	21.827
	constant $\alpha = 10^{-4}$	23.486	0.769	0.356	22.078
	constant $\alpha = 10^{-3}$	16.296	0.614	0.555	22.044
	JVP	24.356	0.757	0.357	60.963
	FD	23.196	0.750	0.364	31.158
FAST-DIPS (ADMM + proj.)	constant $\alpha = 10^{-6}$	21.522	0.641	0.496	25.011
	constant $\alpha = 10^{-5}$	25.021	0.768	0.339	25.110
	constant $\alpha = 10^{-4}$	23.328	0.797	0.298	25.159
	JVP	25.530	0.811	0.273	63.952
	FD	21.041	0.736	0.355	34.197

Table 3: Ablation of step-size selection inside two per-level solvers. Left: Gaussian blur (pixel). Right: HDR (latent). We compare constant α, analytic/JVP α^*, and forward-only α_{FD} within QDP (no splitting, no proj.) and FAST-DIPS (ADMM + proj.). For fairness, compute is matched by allocating $K \times S$ gradient steps per level to QDP when FAST-DIPS uses K ADMM iterations with S gradient steps each; projection/dual updates are negligible. All results are evaluated on FFHQ256 10 samples.

Task	PSNR	SSIM	LPIPS
Gaussian deblurring	28.730	0.814	0.273
Random Inpainting	30.806	0.878	0.192
Nonlinear deblurring	27.016	0.781	0.266

Table 4: Quantitative results under Poisson measurement noise ($\lambda_{\text{poisson}} = 1$). FAST-DIPS remains accurate and perceptually faithful across tasks.

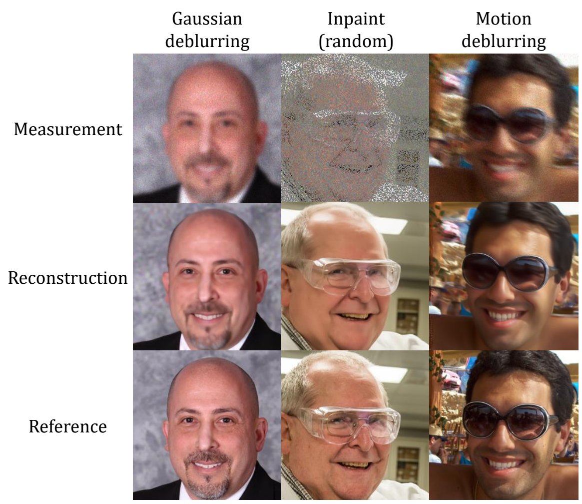

Figure 4: Qualitative reconstructions under Poisson measurement noise ($\lambda_{\text{poisson}} = 1$): FAST-DIPS preserves edges and textures across tasks.

tasks, the reconstruction metrics steadily improve as S increases, indicating that additional gradient refinement helps the solver locate more accurate correction points in challenging settings. Overall, both K and S present a clear quality–cost trade-off, with K providing general improvements and S offering additional benefits especially for complex nonlinear problems.

Hyperparameter Robustness We investigate the robustness of our method to its main hyperparameters. Table 6 shows the results for the super-resolution task when sweeping the ADMM penalty ρ, and the credible set radius ϵ. The performance remains remarkably stable across a wide range of values for each parameter. This highlights a key advantage of FAST-DIPS: it is not sensitive to fine-tuning and delivers strong results with default settings, enhancing its practicality and ease of use.

Hybrid Schedule Trade-off In our hybrid pixel-latent framework, the σ_{switch} parameter determines the point at which the correction process transitions from pixel space to latent space. Table 7 illustrates the resulting trade-off between performance and run-time. Performing the initial correction steps in pixel space ($\sigma_{\text{switch}} > 0$) provides a fast and effective rough update, significantly reducing the overall computation time. The subsequent switch to latent-space updates allows for more stable, fine-grained corrections that respect the generative manifold. This hybrid strategy proves highly effective, and an intermediate σ_{switch} value offers an optimal balance between speed and reconstruction fidelity.

K	S	Super Resolution 4× PSNR	SSIM	LPIPS	Run-time (s)	Nonlinear Blur PSNR	SSIM	LPIPS	Run-time (s)
2	1	29.627	0.837	0.260	1.444	25.101	0.711	0.367	8.711
	3	29.675	0.838	0.259	1.778	27.275	0.780	0.302	17.284
	5	29.678	0.838	0.259	2.113	27.914	0.800	0.285	26.766
3	1	29.740	0.839	0.256	1.569	25.491	0.725	0.347	10.937
	3	29.734	0.839	0.256	2.092	27.079	0.785	0.287	26.888
	5	29.734	0.839	0.256	2.608	27.546	0.801	0.274	38.314
5	1	29.736	0.838	0.252	1.794	25.501	0.734	0.323	15.707
	3	29.737	0.838	0.252	2.641	26.706	0.777	0.267	39.492
	5	29.737	0.838	0.252	3.418	27.006	0.786	0.265	63.044

Table 5: The trade-off between quality and cost in the x-update step. For complex nonlinear tasks such as nonlinear deblurring, increasing the number of gradient steps improves reconstruction quality but also increases computational cost. All experiments were conducted on 10 samples using an RTX 6000 Ada GPU.

ρ	PSNR	SSIM	LPIPS
10	27.546	0.783	0.339
100	29.614	0.836	0.267
200	29.740	0.839	0.256
500	29.565	0.828	0.260
1000	29.363	0.816	0.276

ε	PSNR	SSIM	LPIPS
0	29.739	0.839	0.255
0.01	29.739	0.839	0.255
0.05	29.740	0.839	0.256
0.1	29.740	0.839	0.256
1	29.726	0.839	0.258

Table 6: Sensitivity analysis of the main hyperparameters for Super resolution 4×, evaluated on 10 FFHQ images. The table shows the performance while sweeping the ADMM penalty ρ and the credible set radius ε. The results demonstrate that our method is robust, with performance remaining remarkably stable across a wide range of values, which reduces the need for extensive hyperparameter tuning.

σ_{switch}	PSNR	SSIM	LPIPS	Run-time (s)
< 0.0	24.283	0.553	0.469	3.082
0.2	27.185	0.681	0.374	11.727
1	28.809	0.793	0.302	38.014
5	28.828	0.791	0.306	73.138
>10.0	28.819	0.791	0.307	90.646

Table 7: Performance of the hybrid pixel–latent schedule with varying σ_{switch} values for 4× super-resolution on 10 FFHQ images. The schedule performs correction in pixel space when $\sigma_t > \sigma_{\text{switch}}$ and in latent space otherwise. The results, measured on an RTX 6000 Ada GPU, show that a balanced approach ($\sigma_{\text{switch}} = 0.6$) is more effective than a purely pixel-space (< 0.0) or purely latent-space (> 10.0) correction strategy.

Experiments with non-Gaussian noise. Figure 4 and Table 4 evaluate FAST-DIPS under Poisson measurement noise with rate $\lambda_{\text{poisson}} = 1$, showing that our method remains accurate and perceptually faithful beyond the additive white Gaussian noise (AWGN) setting. The robustness arises from replacing a parametric likelihood with a set-valued surrogate: at each diffusion level, we solve a denoiser-anchored, hard-constrained proximal problem that enforces feasibility within a measurement-space credible set (Euclidean ball) in a whitened domain, which is inherently tolerant to noise miscalibration and largely insensitive to the exact noise law when residuals are appropriately whitened. Our analytic step-size rules yield stable optimization across tasks, supporting practical insensitivity to corruption type.

Extension to Non-Differentiable Operators To address the applicability of FAST-DIPS to non-differentiable degradations, we evaluated our framework on JPEG restoration. While the standard JPEG compression pipeline involves a non-differentiable quantization step, it can be effectively handled using a differentiable surrogate (Reich et al. (2024)).

We applied FAST-DIPS using this differentiable surrogate to guide the restoration process under measurement noise $\beta = 0.05$. Qualitative results are presented in Figure 5, demonstrating that our method effectively suppresses blocking artifacts and restores high-frequency details. Quantitative metrics in Table 8 further confirm competitive reconstruction performance. These results suggest

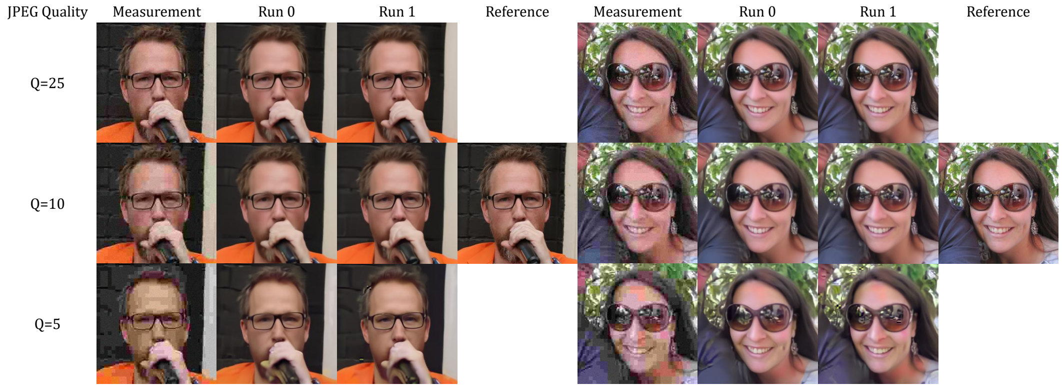

Figure 5: Qualitative results for JPEG restoration on FFHQ using FAST-DIPS with a differentiable surrogate operator. We display the measurement, reconstruction, and the ground-truth reference across three compression levels: JPEG Quality 5, 10, and 25.

JPEG Quality	PSNR	SSIM	LPIPS
25	31.175	0.869	0.229
10	29.343	0.834	0.267
5	26.749	0.788	0.338

Table 8: Quantitative evaluation of JPEG restoration on FFHQ across JPEG quality factors 5, 10, and 25.

Method	PSNR	SSIM	LPIPS	Runtime (s)
Latent-DAPS	28.308	0.809	0.428	580.664
Ours(Latent)	**31.438**	**0.852**	**0.356**	**247.399**

Table 9: Quantitative evaluation of high-resolution (512×512) Gaussian deblurring on FFHQ using 10 samples, conducted on an RTX 6000 Ada GPU.

that FAST-DIPS remains highly effective for formally non-differentiable problems, provided a differentiable proxy of the forward operator is available.

High-resolution image data We further conducted high-resolution experiments on the FFHQ dataset at 512×512 resolution in the latent setting, going beyond the standard 256×256 regime. As shown in Table 9, our method improves PSNR, SSIM, and LPIPS compared to Latent DAPS—the most recent state-of-the-art latent diffusion–based inverse problem solver—while also achieving approximately $2.3\times$ faster runtime. These results suggest that our hybrid approach can handle high-resolution inputs effectively and benefit from efficient computations in the latent space.

Qualitative Results Figures 6-22 provide additional qualitative samples for a comprehensive set of eight problems on FFHQ and ImageNet dataset. These results visually demonstrate the high-quality and consistent reconstructions achieved by both the pixel-space (FAST-DIPS) and latent-space (Latent FAST-DIPS) versions of our method.

A.9 Future work and Limitations

Our proposed method, FAST-DIPS, provides a robust framework for solving inverse problems, and its hyperparameter stability opens up several promising directions for future work. The framework is defined by a few key hyperparameters ($\rho, \varepsilon, \sigma_{\text{switch}}$), and as shown in additional experimental section Table 6, 7, it exhibits robustness across a wide range of their values, enhancing its practical usability. Among these, the ADMM penalty parameter ρ can be considered the most influential. While our experiments show stable performance with a fixed value, integrating adaptive penalty selection strategies could further improve convergence and robustness. Similarly, exploring an optimal or adaptive schedule for the hybrid switching point σ_{switch} remains another interesting avenue for research.

Despite these strengths and opportunities, we also acknowledge a primary limitation of the current framework: its dependency on differentiable forward operators. FAST-DIPS is "adjoint-free" in the sense that it does not require a hand-coded adjoint operator. However, its efficiency heavily relies on automatic differentiation to compute VJP and JVP needed for the analytic step size α^*. This implicitly assumes that the forward operator $\mathcal{A}$ is (at least piecewise) differentiable. For problems involving non-differentiable operators or black-box simulators where gradients are unavailable, our current approach cannot be directly applied. Future work could explore extensions using zeroth-order optimization techniques or proximal gradient methods that can handle non-differentiable terms.

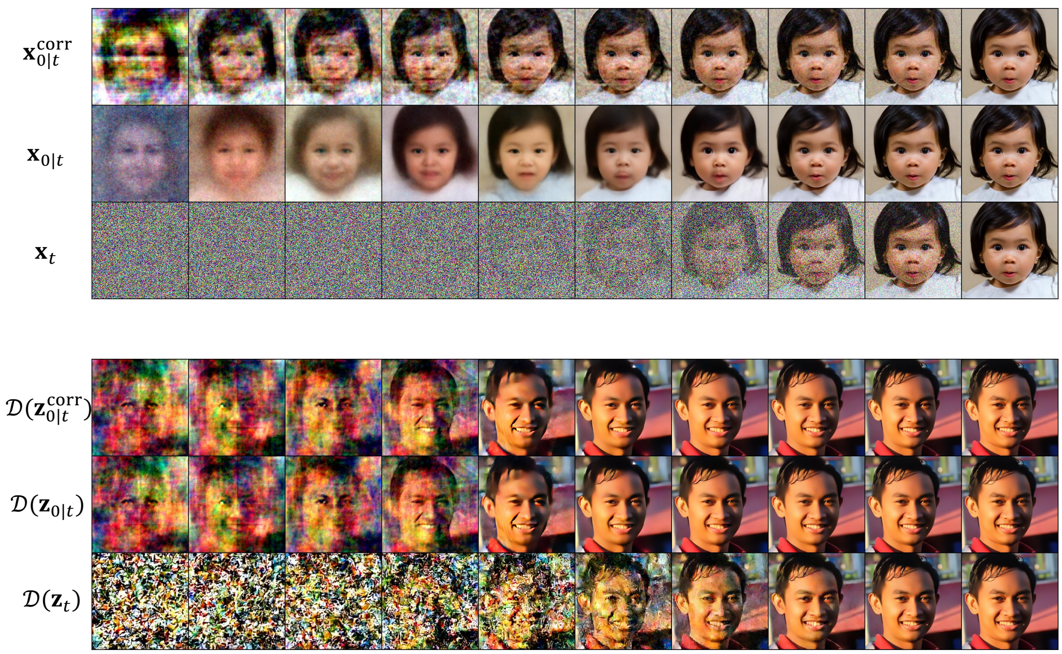

Figure 6: Phase Retrieval trajectory under FAST-DIPS and Latent FAST-DIPS with intermediate iterates along the diffusion schedule.

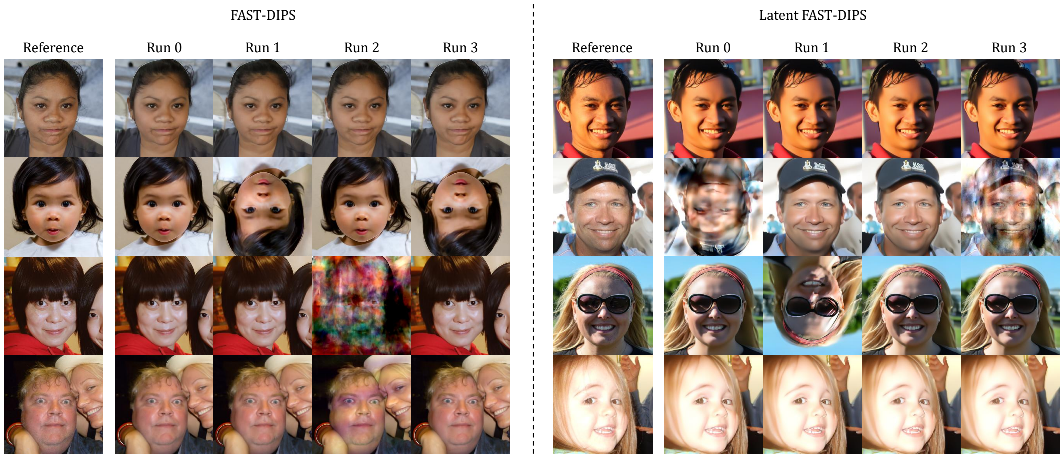

Figure 7: Additional qualitative results for **Phase Retrieval**. We show Measurement, Reconstruction, and Reference for both FAST-DIPS and Latent FAST-DIPS across four runs (Run 0–3).

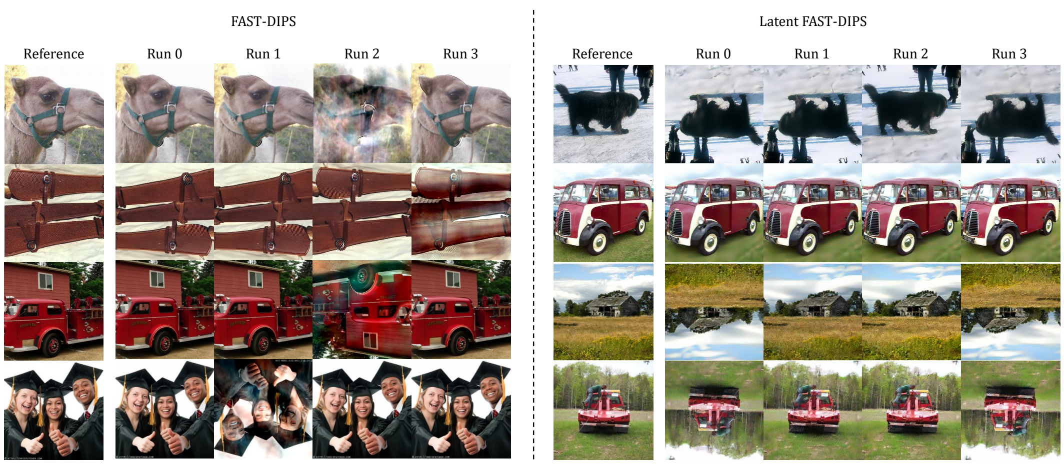

Figure 8: Qualitative results for **Phase Retrieval** on ImageNet dataset. We show Measurement, Reconstruction, and Reference for both FAST-DIPS and Latent FAST-DIPS across four runs (Run 0–3).

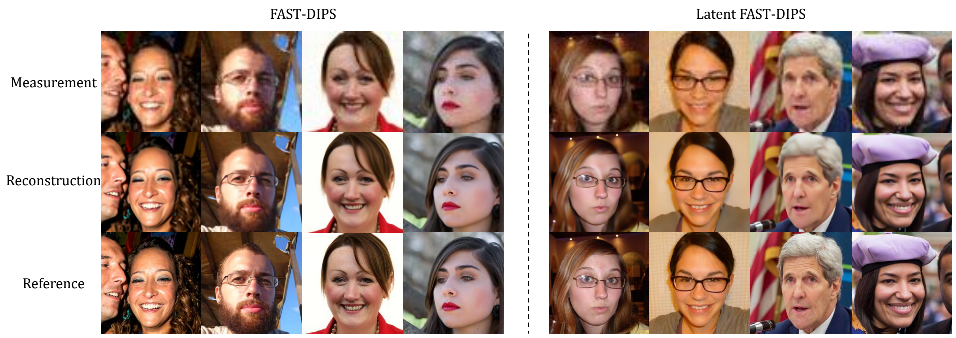

Figure 9: Additional qualitative results for **Super-Resolution** $\times 4$. Measurement, Reconstruction, and Reference are shown for FAST-DIPS and Latent FAST-DIPS.

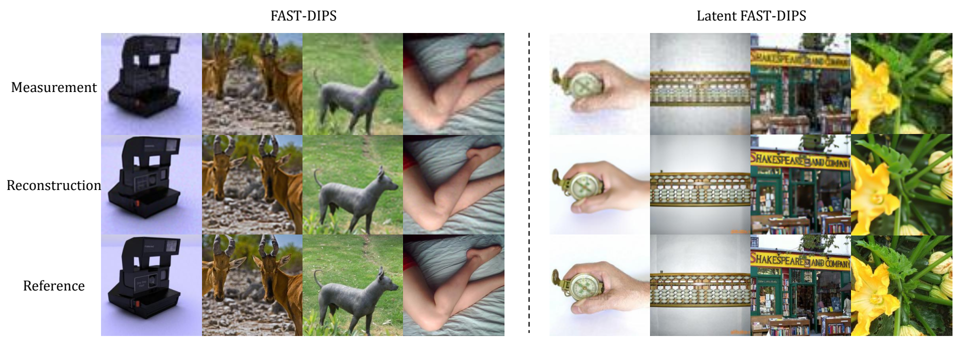

Figure 10: Qualitative results for **Super-Resolution** $\times 4$ on ImageNet dataset. Measurement, Reconstruction, and Reference are shown for FAST-DIPS and Latent FAST-DIPS.

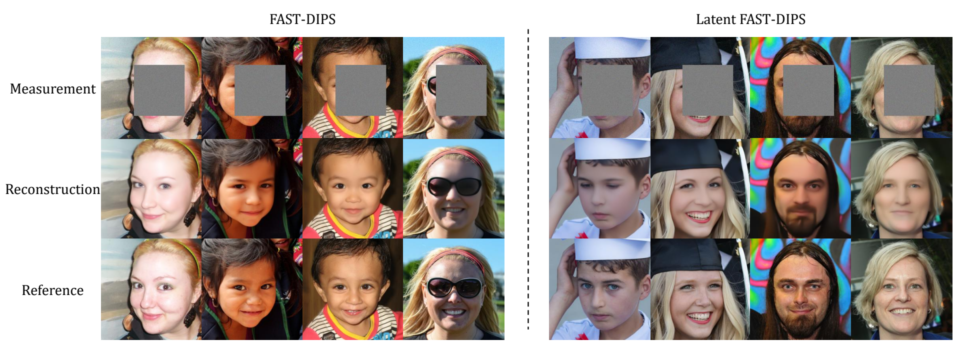

Figure 11: Additional qualitative results for **Inpaint(box)**. We display Measurement, Reconstruction, and Reference for FAST-DIPS and Latent FAST-DIPS.

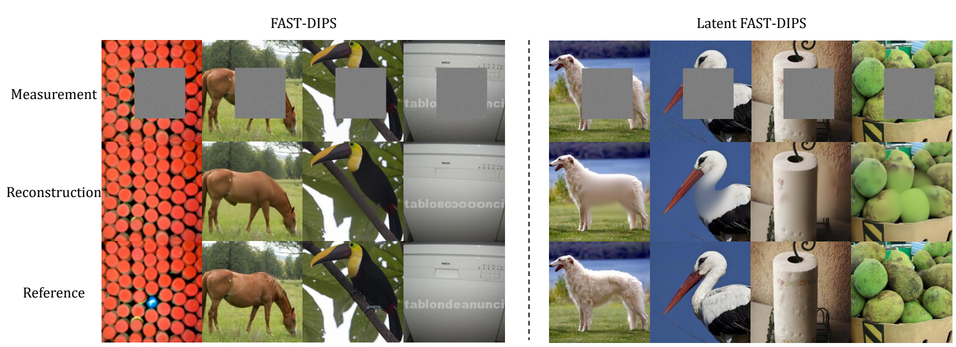

Figure 12: Qualitative results for **Inpaint(box)** on ImageNet dataset. We display Measurement, Reconstruction, and Reference for FAST-DIPS and Latent FAST-DIPS.

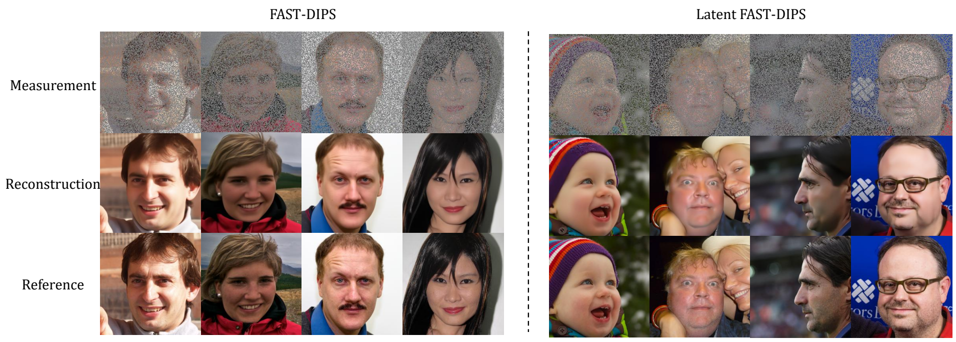

Figure 13: Additional qualitative results for **Inpaint(random)**. Measurement, Reconstruction, and Reference with FAST-DIPS and Latent FAST-DIPS.

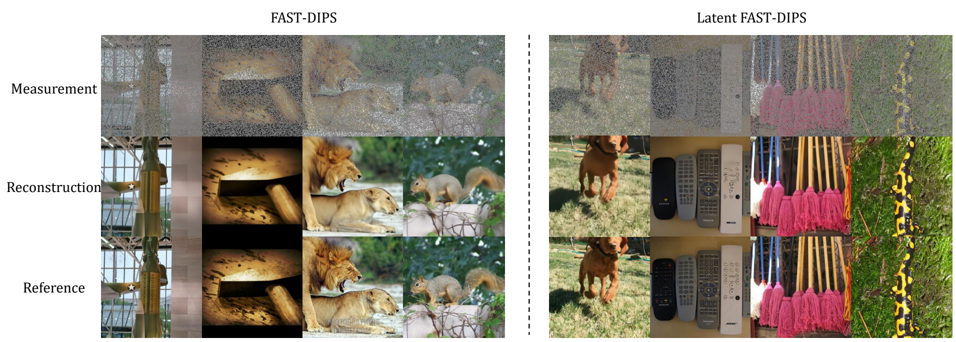

Figure 14: Qualitative results for **Inpaint(random)** on ImageNet dataset. Measurement, Reconstruction, and Reference with FAST-DIPS and Latent FAST-DIPS.

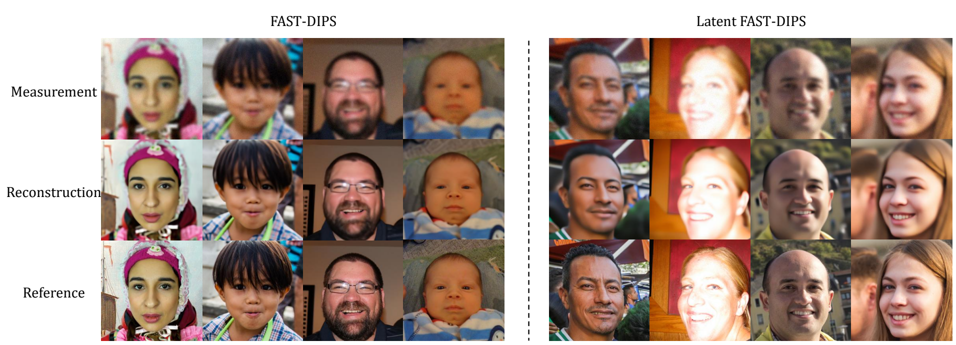

Figure 15: Additional qualitative results for **Gaussian deblurring**. We show Measurement, Reconstruction, and Reference for FAST-DIPS and Latent FAST-DIPS.

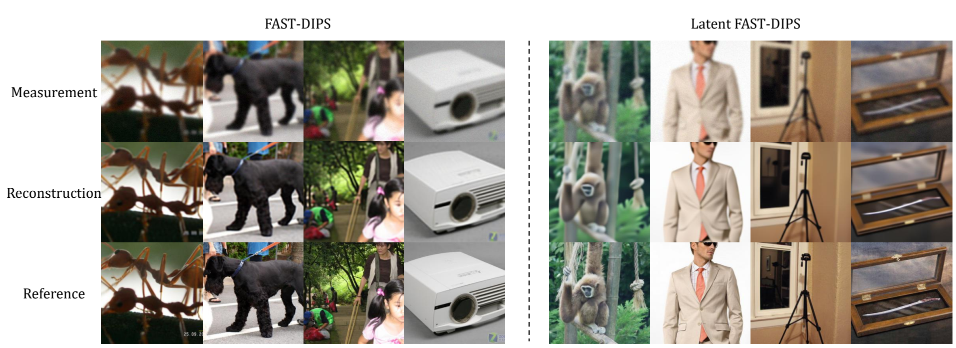

Figure 16: Qualitative results for **Gaussian deblurring** on ImageNet dataset. We show Measurement, Reconstruction, and Reference for FAST-DIPS and Latent FAST-DIPS.

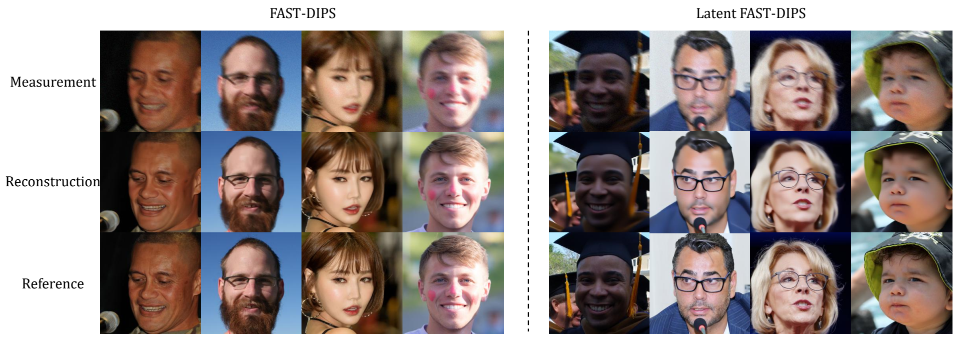

Figure 17: Additional qualitative results for **Motion deblurring**. Measurement, Reconstruction, and Reference are provided for FAST-DIPS and Latent FAST-DIPS.

Figure 18: Qualitative results for **Motion deblurring** on ImageNet dataset. Measurement, Reconstruction, and Reference are provided for FAST-DIPS and Latent FAST-DIPS.

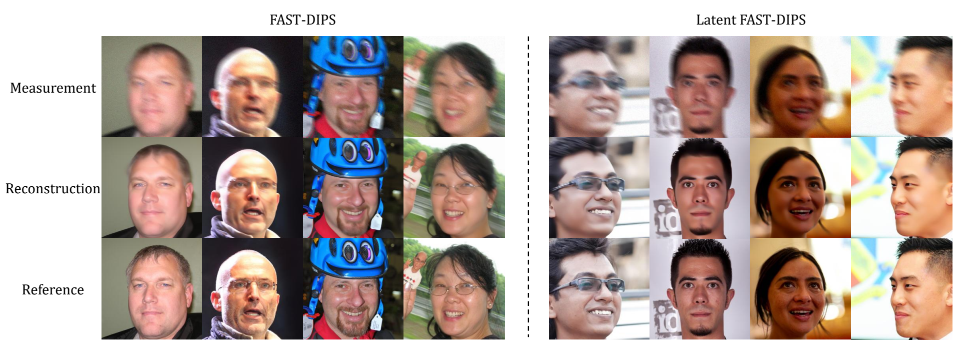

Figure 19: Additional qualitative results for **Nonlinear deblurring**. We present Measurement, Reconstruction, and Reference for FAST-DIPS and Latent FAST-DIPS.

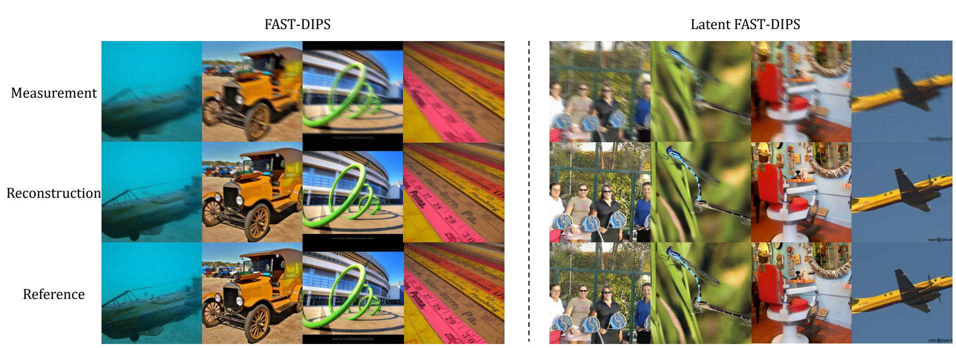

Figure 20: Qualitative results for **Nonlinear deblurring** on ImageNet dataset. We present Measurement, Reconstruction, and Reference for FAST-DIPS and Latent FAST-DIPS.

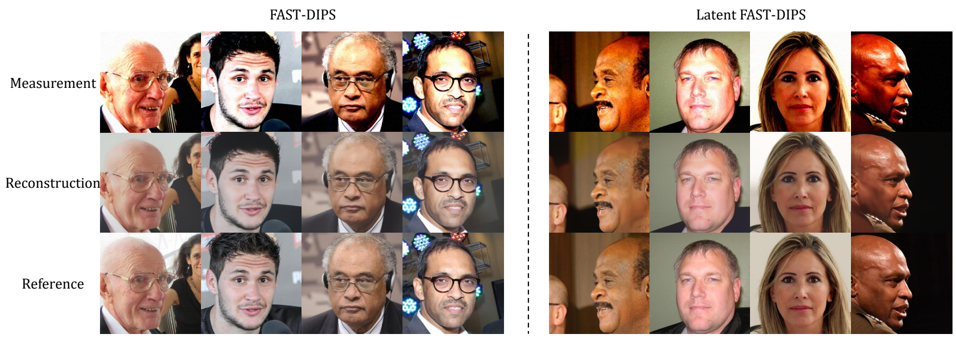

Figure 21: Additional qualitative results for **High Dynamic Range**. Measurement, Reconstruction, and Reference for FAST-DIPS and Latent FAST-DIPS.

Figure 22: Qualitative results for **High Dynamic Range** on ImageNet dataset. Measurement, Reconstruction, and Reference for FAST-DIPS and Latent FAST-DIPS.