GETTING YOUR GUIDANCE WEIGHTS RIGHT IN DIFFUSION AND FLOW-MATCHING POSTERIOR SAMPLING

Anonymous authors
Paper under double-blind review

ABSTRACT

Training-free posterior sampling methods, also known as Plug-and-Play methods, leverage pretrained unconditional diffusion or flow-matching models to solve inverse problems. Most existing approaches rely on guidance weights to balance, at each time step, prior information from the unconditional score or velocity network with measurement consistency, yet the tuning of these weights is often not discussed and is largely left to heuristics. We introduce a simple and principled offline strategy for automatically tuning these guidance weights. Our key observation is that, at each time step, the conditional denoising score-matching objective for diffusion models, or the conditional flow-matching objective for flow-matching models, is a least-squares objective. Therefore, when the conditional prediction is expressed as a weighted sum of the unconditional network output and a measurement-guidance term, optimizing over these weights reduces to a two-dimensional linear least-squares problem. The resulting time-dependent guidance weights can be optimized offline for a given measurement operator, noise level and sampler at the cost of a single minibatch of sampling trajectories, without retraining or fine-tuning the pretrained generative model. Instantiated with the standard Tweedie-based measurement-consistency term, our approach improves posterior sampling and achieves state-of-the-art reconstruction performance across diffusion- and flow-matching-based methods. Moreover, the optimized guidance weights enable diffusion samplers to reduce the number of sampling steps from 1000 to 50 with no significant degradation in reconstruction quality. Code will be made available.

1 INTRODUCTION

Diffusion models (Ho et al., 2020; Karras et al., 2022; Song & Ermon, 2019; Song et al., 2020; Song & Ermon, 2020) and flow matching models (Lipman et al., 2022; Liu, 2022; Tong et al., 2023; Albergo et al., 2025; Liu et al., 2022) have emerged as dominant families of iterative generative models, achieving remarkable success in synthesizing high-fidelity samples across a wide range of data modalities. However, training these models remains computationally expensive and can require substantial computational resources and time. In inverse problems, where the exact measurement operator and noise characteristics often depend on the acquisition setting specified at inference time, retraining or fine-tuning a generative model for each new acquisition setting is often impractical. Practitioners instead commonly rely on training-free methods (Chung et al., 2022a; Song et al., 2023; Boys et al., 2023; Kim et al., 2025; Martin et al., 2025), also referred to as Plug-and-Play methods. These methods leverage unconditionally pretrained generative models to perform approximate posterior sampling without requiring additional training for different measurement operators or noise levels.

Most existing training-free methods (Chung et al., 2022a; Song et al., 2023; Boys et al., 2023; Kim et al., 2025; Martin et al., 2025) incorporate a measurement-consistency step at each time step to modify the unconditional sampling direction, thereby steering the sampling trajectory toward samples consistent with the measurement. Repeating this procedure throughout the sampling process yields a sample from the approximate posterior distribution. Such methods usually rely on guidance weights to balance, at each time step, the prior information provided by the unconditional score or velocity network against measurement consistency. However, the tuning of these weights is often not discussed and is largely left to heuristics.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

In practice, the measurement setup may frequently change, for instance by modifying the number or locations of sensors, or their internal acquisition settings (Donoho, 2006). Such modifications require the guidance weights to be re-tuned, introducing an additional and time-consuming hyperparameter optimization step. Consequently, the need to tune guidance weights represents a key practical bottleneck of existing training-free posterior sampling methods.

This motivates the development of a strategy to automatically and efficiently determine appropriate guidance weights for a given measurement setup. Importantly, such a strategy must introduce negligible computational overhead. From a practitioner's perspective, guidance selection should take only a few seconds, ruling out iterative optimizers (*e.g.* gradient descent, etc.) that require repeated ODE/SDE integrations. In this paper, we introduce a simple and principled offline calibration strategy for automatically tuning time-dependent guidance weights. Our contributions are as follows:

- We make the key observation that, at each time step, the conditional denoising score-matching objective for diffusion models and the conditional flow-matching objective for flow-matching models are least-squares objectives. Therefore, when the conditional prediction is expressed as a weighted sum of the unconditional network output and a measurement-guidance term, optimizing these weights reduces to a two-dimensional linear least-squares problem. This yields a highly efficient, simulation-free strategy for calibrating time-dependent guidance weights.
- While theoretically grounded, the simulation-free calibration strategy does not guarantee improved posterior sampling trajectories, since neither the measurement-consistency term nor the unconditional network is necessarily optimal. To address this limitation, we propose a greedy SIMulation-based CAlibration strategy (SIMCA), which sequentially calibrates the guidance weight at each time step while keeping the weights calibrated at previous time steps fixed. SIMCA requires only a single minibatch of sampling trajectories, making it practical for tuning time-dependent guidance weights in a given measurement setup.
- We empirically demonstrate that SIMCA, instantiated with the standard Tweedie-based measurement-consistency term, matches or surpasses state-of-the-art training-free methods across multiple inverse problems and datasets, using both diffusion and flow-matching backbones.
- We further show that SIMCA enables diffusion samplers to reduce the number of sampling steps from 1000 to 50, a $\times 20$ reduction in inference cost, without significant degradation in reconstruction quality.

2 Background on training-free posterior sampling

Unconditional diffusion models and (Gaussian) flow-matching. Let p_0 denote the data distribution on $\mathbb{R}^d$. Diffusion (Ho et al., 2020; Karras et al., 2022; Song & Ermon, 2019; Song et al., 2020; Song & Ermon, 2020) and (Gaussian) flow matching[1] Lipman et al. (2022); Liu (2022); Albergo et al. (2025) models both generate samples from p_0 by transporting Gaussian noise along a prescribed family of marginals, obtained by interpolating data and noise,

$$\mathbf{x}_t = \alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}, \qquad \mathbf{x}_0 \sim p_0, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \qquad t \in [0,1], \tag{1}$$

so that $p_t(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\alpha_t \mathbf{x}_0, \sigma_t^2 \mathbf{I})$. Throughout, $t = 0$ denotes data and $t = 1$ noise. The schedule (α_t, σ_t) is chosen with $\alpha_0 \approx 1$, $\sigma_0 \approx 0$, $\alpha_1 \approx 0$ and $\sigma_1 \approx 1$, so that $p_1 \approx \mathcal{N}(\mathbf{0}, \mathbf{I})$ and sampling can be initialized from pure noise. The variance-preserving diffusion schedule $\alpha_t^2 + \sigma_t^2 = 1$ and the optimal-transport interpolant $\alpha_t = 1 - t$, $\sigma_t = t$ are two instances (Song et al., 2020; Lipman et al., 2022).

The two families differ in the vector field they learn. Diffusion models regress the score $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) \approx \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)$ by (often weighted) Denoising Score Matching (DSM) (Vincent, 2011; Song & Ermon, 2019; Song et al., 2020)

$$\mathcal{L}_{\text{DSM}}(\boldsymbol{\theta}) = \mathbb{E}\left[\left\| \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) + \frac{\boldsymbol{\epsilon}}{\sigma_t} \right\|^2 \right], \tag{2}$$

[1] For flow-matching, we focus here on the independent coupling case, as it allows us to establish the equivalence between diffusion models and Gaussian flow matching (Gao et al., 2024). In our experiments, however, we evaluate velocity networks trained using both independent and OT couplings.

where $\|\cdot\|$ is the L2-norm, while flow matching models regress the velocity $\mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ by Conditional Flow Matching (CFM) (Lipman et al., 2022; Liu, 2022)

$$\mathcal{L}_{\text{CFM}}(\boldsymbol{\theta}) = \mathbb{E}\left[\|\mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \dot{\alpha}_t \mathbf{x}_0 - \dot{\sigma}_t \boldsymbol{\epsilon}\|^2\right], \tag{3}$$

where $\dot{\alpha}_t$ and $\dot{\sigma}_t$ denote the time derivatives. In the rest of the paper, we employ unified notations as our calibration strategy can be applied in both cases:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}\left[\|\mathbf{u}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \boldsymbol{\psi}_t(\mathbf{x}_0, \boldsymbol{\epsilon})\|^2\right], \tag{4}$$

where

$$\boldsymbol{\psi}_t(\mathbf{x}_0, \boldsymbol{\epsilon}) = \begin{cases} -\boldsymbol{\epsilon}/\sigma_t & \text{(diffusion case)}, \\ \dot{\alpha}_t \mathbf{x}_0 + \dot{\sigma}_t \boldsymbol{\epsilon} & \text{(flow-matching case)}. \end{cases} \tag{5}$$

Sampling integrates the corresponding dynamics backward on a grid $1 = t_K > \cdots > t_0 = 0$, where the number of steps K trades sample quality against computational cost. In this paper, we consider both deterministic and stochastic samplers, namely DDPM (Ho et al., 2020), Euler-Maruyama on the reverse SDE, and the Euler step of the flow ODE:

$$\mathbf{x}_{t_{k-1}} = \Phi\left(\mathbf{x}_{t_k}, \mathbf{u}_{\boldsymbol{\theta}}(\mathbf{x}_{t_k}, t_k), t_k\right). \tag{6}$$

Inverse problems and Training-free posterior sampling. We seek to recover an unknown signal $\mathbf{x}_0$ from a degraded observation

$$\mathbf{y} = \mathcal{A}\mathbf{x}_0 + \mathbf{n}, \qquad \mathbf{n} \sim \mathcal{N}(\mathbf{0}, \sigma_m^2 \mathbf{I}), \tag{7}$$

where $\mathcal{A}$ is a known linear measurement operator, the setting we consider throughout.

Posterior sampling amounts to running the same dynamics driven by a learned conditional field $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{y}, t)$ instead of the unconditional one $\mathbf{u}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$. Training-free methods aim to approximate that conditional field $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{y}, t)$ leveraging an unconditionally pretrained network $\mathbf{u}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$, along with a measurement-consistency mechanism, without requiring additional training for different measurement operators $\mathcal{A}$ or noise levels σ_m (Song et al., 2020; Chung et al., 2022a; Song et al., 2023; Boys et al., 2023).

Measurement-consistency mechanism. Many different measurement-consistency mechanisms have been proposed for both diffusion and flow-matching models. Existing strategies include Tweedie-based approximations (Chung et al., 2022a; Song et al., 2023; Boys et al., 2023), spectral methods exploiting the singular value decomposition of $\mathcal{A}$ (Kawar et al., 2022a;b), sequential Monte Carlo (Cardoso et al., 2024; Dou & Song, 2024; Kelvinius et al., 2025), conversion of the velocity field to a score followed by diffusion-style guidance (Kim et al., 2025; Song et al., 2026), and schemes that interleave generative steps with optimization (Song et al., 2021; Graikos et al., 2022; Mardani et al., 2024) or projection (Wang et al., 2022; Chung et al., 2022b; Song et al., 2023) steps. In this paper, we focus on a specific measurement-consistency mechanism that approximates the conditional field with a weighted sum of the unconditional network output and a measurement-guidance term $\mathbf{f}(\mathbf{x}_t, \mathbf{y}, t)$:

$$\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{y}, t) \;\approx\; w_{u,t}\, \mathbf{u}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) + w_{f,t}\, \mathbf{f}(\mathbf{x}_t, \mathbf{y}, t), \tag{8}$$

where $\mathbf{w}_t = [w_{u,t}, w_{f,t}]^\top$ are the scalar guidance weights. A well-known example of an additive measurement-consistency term is the Tweedie-based measurement-consistency term introduced in (Chung et al., 2022a), which we employ in our experiments:

$$\mathbf{f}(\mathbf{x}_t, \mathbf{y}, t) \propto \nabla_{\mathbf{x}_t} \|\mathbf{y} - \mathcal{A}\hat{\mathbf{x}}_0(\mathbf{x}_t)\|^2. \tag{9}$$

where $\hat{\mathbf{x}}_0(\mathbf{x}_t)$ is the Tweedie estimate (Efron, 2011). This paper focuses on the tuning of these guidance weights, that are often not discussed and largely left to heuristics.

3 Related Work

Guidance weights in training-free posterior sampling. Additive training-free samplers weight their measurement-consistency term either heuristically, as the residual-normalized step of DPS (Chung et al., 2022a) and the hand-set schedules of flow-based samplers (Kim et al., 2025; Martin et al., 2025), or through a Gaussian model of the denoising posterior, as in ΠGDM (Song et al., 2023), TMPD (Boys et al., 2023), their refinements (Peng et al., 2024), and their flow counterpart OT-ODE (Pokle et al., 2024). The latter derive the weight from an assumed posterior covariance rather than tuning it, and thereby improve both the measurement term and its effective scale; our ablations suggest that this second effect accounts for much of their benefit (Section 5.3). What none of them does is fit the weight to a criterion measurable from data, and all fix the weight of the pretrained field to one.

Adaptive guidance. Time-varying weights have been studied mainly for classifier-free guidance (Malarz et al., 2025), where the weight trades sample quality against conditioning fidelity and, having no reference value, is set heuristically. In training-free inverse problems the guided field approximates a well-defined conditional field, so its approximation error can be measured against the tractable target of Eq. 5 and minimized in closed form rather than tuned. SAIP (Wang & Meng, 2026) also adapts the balance between the two terms, but through a single coupled parameter, estimated online and stated against the intractable posterior score. Methodologically, our closed-form solve from Monte Carlo moments follows Analytic-DPM (Bao et al., 2022), applied there to the unconditional reverse variance.

4 Method

Given a pretrained unconditional score or velocity network $\mathbf{u}_{\boldsymbol{\theta}}$ and a user-specified measurement-consistency term $\mathbf{f}$, we seek to automatically and efficiently calibrate time-dependent guidance weights $\mathbf{w} = \{\mathbf{w}_{t_k}\}_{k=1\dots K}$ for a given measurement operator $\mathcal{A}$ and noise level σ_m. Importantly, the proposed strategy must introduce negligible computational overhead, since from a practitioner's perspective, calibration should take only a few seconds, ruling out iterative optimizers (e.g. gradient descent, etc.). In the remainder of this section, we omit the time subscripts from the distributions to lighten the notation; for example $p_0(\mathbf{x}_0) = p(\mathbf{x}_0)$.

4.1 Simulation-free calibration strategy

Our key observation is that replacing $\mathbf{u}_{\boldsymbol{\theta}}$ with Eq. 8 in Eq. 4 yields a linear least-squares problem w.r.t. the guidance weights $\mathbf{w}_{t_k} = [w_{u,t_k}, w_{f,t_k}]^\top$:

$$\mathcal{L}_{\text{free},t_k}(\mathbf{w}_{t_k}) = \mathbb{E}_{p(\mathbf{x}_{t_k}|\mathbf{x}_0)p(\mathbf{y}|\mathbf{x}_0)p(\mathbf{x}_0)}\left[\left\|\mathbf{e}(\mathbf{x}_{t_k}, \mathbf{w}_{t_k}, \mathbf{x}_0, \mathbf{y}, t_k)\right\|^2\right] \tag{10}$$

where

$$\mathbf{e}\left(\mathbf{x_t}, \mathbf{w}_t, \mathbf{x}_0, \mathbf{y}, t\right) = w_{u,t}\,\mathbf{u}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) + w_{f,t}\,\mathbf{f}(\mathbf{x}_t, \mathbf{y}, t) - \boldsymbol{\psi}_t\left(\mathbf{x}_0, \frac{\mathbf{x}_t - \alpha_t \mathbf{x}_0}{\sigma_t}\right) \tag{11}$$

is the regression error, $p(\mathbf{x}_{t_k}|\mathbf{x}_0)$ corresponds to the forward process Eq. 1 and $p(\mathbf{y}|\mathbf{x}_0)$ is the measurement likelihood Eq. 7.

This observation enables a simulation-free calibration strategy that does not require sampling model trajectories, in which the time-dependent guidance weights are optimized independently at each of the K time steps. Since only two scalar parameters, w_{u,t_k} and w_{f,t_k}, need to be estimated at each time step, the expectation over $p(\mathbf{x}_0)$ can be approximated in practice using a single small minibatch, making the overall calibration procedure highly efficient.

However, there is a caveat to the simulation-free calibration strategy described above. When optimizing Eq. 10 over a sufficiently rich function class, such as a neural network, the minimizer corresponds to the true posterior score in the diffusion case (Batzolis et al., 2021). Similar results can be obtained for the flow-matching case. In contrast, our calibration is restricted to only two parameters, and neither the measurement-consistency term nor the unconditional network is necessarily optimal. Therefore, optimizing the guidance weights independently at each time step in

Algorithm 1 Calibration of the guidance weights $\{\mathbf{w}_{t_k}\}_{k=1\dots K}$. Black lines are common to both variants; the simulation-based variant adds the blue lines and skips the orange one.

Require: $\mathbf{u}_{\boldsymbol{\theta}}$, $\mathbf{f}$, minibatch $\{(\mathbf{x}_{0,i}, \mathbf{y}_i)\}_{i=1}^N$, schedule $\{(t_k, \alpha_{t_k}, \sigma_{t_k})\}_{k=1\dots K}$, sampler Φ

$\mathbf{x}_{t_K}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for $i = 1, ..., N$ ▷ Initialize sampling trajectories
for $k = K, ..., 1$ **do** ▷ for each timestep
  $t \leftarrow t_k$; $\mathtt{H} \leftarrow 0$; $\mathbf{b} \leftarrow 0$
  for $i = 1, \dots, N$ **do** ▷ for each sample in the minibatch
    $\boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ $\mathbf{x}_t^{(i)} \leftarrow \alpha_t \mathbf{x}_{0,i} + \sigma_t \boldsymbol{\epsilon}_i$
    $\boldsymbol{\epsilon}_i \leftarrow (\mathbf{x}_t^{(i)} - \alpha_t \mathbf{x}_{0,i})/\sigma_t$
    $\boldsymbol{\psi}_i \leftarrow \boldsymbol{\psi}_t(\mathbf{x}_{0,i}, \boldsymbol{\epsilon}_i)$ $\mathbf{u}_i \leftarrow \mathbf{u}_{\boldsymbol{\theta}}(\mathbf{x}_t^{(i)}, t)$ $\mathbf{f}_i \leftarrow \mathbf{f}(\mathbf{x}_t^{(i)}, \mathbf{y}_i, t)$
    $\mathtt{H} \leftarrow \mathtt{H} + \begin{bmatrix} \mathbf{u}_i^\top \mathbf{u}_i & \mathbf{u}_i^\top \mathbf{f}_i \\ \mathbf{f}_i^\top \mathbf{u}_i & \mathbf{f}_i^\top \mathbf{f}_i \end{bmatrix}$ $\mathbf{b} \leftarrow \mathbf{b} + \begin{bmatrix} \mathbf{u}_i^\top \boldsymbol{\psi}_i \\ \mathbf{f}_i^\top \boldsymbol{\psi}_i \end{bmatrix}$
  end for
  $\mathbf{w}_t^* \leftarrow$ solve $\mathtt{H}\mathbf{w} = \mathbf{b}$ ▷ 2x2 linear system
  $\mathbf{x}_{t_{k+1}}^{(i)} \leftarrow \Phi\big(\mathbf{x}_t^{(i)},\, w_{u,t}^* \mathbf{u}_i + w_{f,t}^* \mathbf{f}_i,\, t\big)$ for $i = 1, ..., N$ ▷ propagation of sampled trajectories
end for
return $\{\mathbf{w}_{t_k}^*\}_{k=1\dots K}$

Eq. 10 does not guarantee that the resulting sampling trajectories will be close to the true posterior trajectories. In the following section, we therefore turn to simulation-based calibration.

4.2 Simulation-based calibration strategy

We argue that, since neither the measurement-consistency term nor the unconditional network is necessarily optimal, an efficient calibration strategy should be simulation-based. Indeed, during sampling, at a given timestep t_k, a sample $\mathbf{x}_{t_k}$ is necessarily drawn from an approximate posterior $\widetilde{p}(\mathbf{x}_{t_k}|\mathbf{y})$, rather than from the true posterior $p(\mathbf{x}_{t_k}|\mathbf{y})$. This discrepancy should therefore be taken into account when calibrating the guidance weights.

To do so, we propose to replace the forward-process density $p(\mathbf{x}_{t_k}|\mathbf{x}_0)$, in Eq. 10, with the corresponding approximate posterior $\widetilde{p}(\mathbf{x}_{t_k}|\mathbf{y})$. Importantly, $\widetilde{p}(\mathbf{x}_{t_k}|\mathbf{y})$ depends on the guidance weights $\{\mathbf{w}_{t_i}\}_{i=k+1\dots K}$.

We therefore obtain the following simulation-based objective:

$$\mathcal{L}_{\text{sim},t_k}(\mathbf{w}_{t_k}, \mathbf{w}_{t_{k+1}}, ..., \mathbf{w}_{t_K}) = \mathbb{E}_{\widetilde{p}(\mathbf{x}_{t_k}|\mathbf{y})p(\mathbf{y}|\mathbf{x}_0)p(\mathbf{x}_0)}\left[\|\mathbf{e}(\mathbf{x}_{t_k}, \mathbf{w}_{t_k}, \mathbf{x}_0, \mathbf{y}, t_k)\|^2\right], \tag{12}$$

which is a linear least-squares objective in $\mathbf{w}_{t_k}$, but depends non-linearly on the guidance weights on the guidance weights $\{\mathbf{w}_{t_i}\}_{i=k+1\dots K}$. To obtain an efficient calibration algorithm, we adopt a greedy strategy in which only the current guidance weights $\mathbf{w}_{t_k}$ are optimized, while the weights calibrated at previous time steps are kept fixed:

$$\mathbf{w}_{t_k}^* = \arg\min_{\mathbf{w}_{t_k}} \mathcal{L}_{\text{sim},t_k}(\mathbf{w}_{t_k}, \mathbf{w}_{t_{k+1}}^*, ..., \mathbf{w}_{t_K}^*). \tag{13}$$

Since Eq. 12 is a linear least-squares problem in $\mathbf{w}_{t_k}$, optimization of Eq. 13 reduces to a 2×2 linear system. The proposed SIMulation-based CAlibration strategy (SIMCA) is detailed in Alg. 1. Importantly, the resulting calibration also depends on the choice of the sampler Φ (see Eq. 6).

4.3 Practical considerations

Cost. SIMCA requires only a single minibatch of sampling trajectories, making it practical for tuning time-dependent guidance weights for a given measurement setup. Once calibrated, the guidance weights are fixed and introduce no inference-time overhead.

Recalibration. The guidance weights depend on the measurement operator $\mathcal{A}$, the noise level σ_m, the measurement-consistency term $\mathbf{f}$, the pretrained unconditional network $\mathbf{u}_{\boldsymbol{\theta}}$, the sampler Φ, and

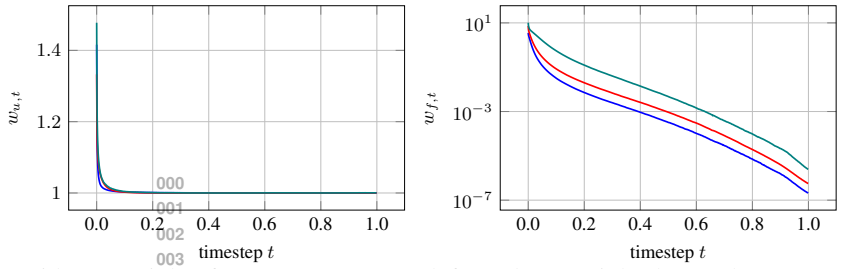


Figure 1: Guidance weights from SIMCA: $w_{u,t}$ (left) and $w_{f,t}$ (right, log scale) on FFHQ at $K = 1000$, for box inpainting, random inpainting and super-resolution.

the number of timesteps. They must therefore be recomputed whenever any of these components changes. Since each recalibration requires only one offline pass, practitioners can efficiently adapt the guidance weights to different measurement setups.

Compatibility. Our calibration procedure applies to any method whose guidance takes the additive form in Eq. equation 8. It requires no retraining, architectural changes, or modifications to the sampler.

5 Experiments

5.1 Experimental Setup

Backbones and protocols. To fairly evaluate SIMCA, we follow the evaluation protocols of DAPS (Zhang et al., 2025) for diffusion and Flower (Pourya et al., 2026) for flow matching. See Appendix F for further details.

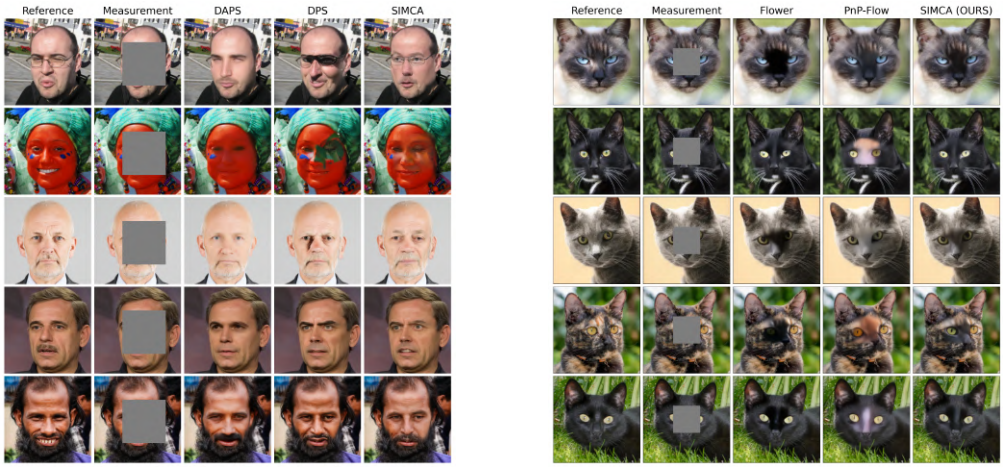
Baselines. For diffusion we compare against DPS (Chung et al., 2022a), DDRM (Kawar et al., 2022a), DDNM (Wang et al., 2022), DiffPIR (Zhu et al., 2023), FPS-SMC (Dou & Song, 2024), and DAPS (Zhang et al., 2025). For flow matching we compare against PnP-GS (Hurault et al., 2022), DiffPIR (Zhu et al., 2023), OT-ODE (Pokle et al., 2024), D-Flow (Ben-Hamu et al., 2024), Flow-Priors (Zhang et al., 2024), PnP-Flow (Martin et al., 2025), and Flower (Pourya et al., 2026). All baseline metrics are those published by the respective benchmarks, obtained under the same operators, noise levels, and test images as ours, with hyperparameters grid-searched to maximize PSNR.

Calibration protocol. Guidance weights are calibrated offline on images disjoint from the test images: for flow matching, we use the same calibration split as in (Pourya et al., 2026); for diffusion, we use images from the training sets of FFHQ (Karras et al., 2019) and ImageNet (Deng et al., 2009). All main tables report SIMCA, the simulation-free variant being deferred to Section 5.3. Since the weights are tied to the sampler and its discretization, each configuration is calibrated separately and its step count reported: $K \in \{1000, 500, 50\}$ for diffusion, with a finer sweep in Appendix E, and the renoising sampler of Flower (with the same number of steps) or a plain Euler integrator for flow matching.

5.2 Main results

Calibrated guidance weights. Fig. 1 shows the guidance weights obtained on FFHQ for three measurement operators. The schedules are therefore neither constant nor reproducible by a single hand-designed curve shared across operators, which is what the remainder of this section quantifies.

Diffusion backbones. Table 1 reports results on FFHQ and ImageNet. SIMCA either matches or outperforms the state-of-the-art methods on almost every metric, task, and dataset: on random inpainting, LPIPS drops from 0.203 to 0.068 and FID from 69.20 to 18.31 on FFHQ, and from 0.297 to 0.109 and 87.53 to 32.08 on ImageNet.



(a) FFHQ, diffusion backbone.

(b) AFHQ-Cat, flow-matching backbone.

Figure 2: Qualitative comparison on the box inpainting task. SIMCA produces completions coherent with the visible context, whereas the baselines oversmooth the fill, hallucinate details, leak colours, or produce dark or discoloured fills.

The most informative comparison is against DAPS (Zhang et al., 2025), the strongest and most elaborate baseline. SIMCA surpasses it on random inpainting by a wide margin on both datasets (31% lower LPIPS and 43% lower FID on FFHQ), on all four metrics on FFHQ super-resolution, and on PSNR, SSIM, and LPIPS on FFHQ motion deblurring. On box inpainting the results are mixed: SSIM and LPIPS improve over every baseline, while PSNR remains below DDNM and FPS-SMC. This is the signature of a sampler that explores the posterior rather than collapsing onto its mean, and box inpainting is the task with the broadest posterior. Figure 2 shows this directly: SIMCA completes the structure visible outside the mask, where DAPS oversmooths and DPS hallucinates.

Effect of the step count. The three SIMCA rows in each block of Table 1 differ only in the number of sampler iterations. Surprisingly, halving the budget from 1000 to 500 steps improves almost every metric on every task. Reducing further to 50 improves PSNR and SSIM again, reaching the best values in the table on four of the five FFHQ tasks, while LPIPS and FID degrade moderately: coarser discretization accumulates less stochasticity, moving the sampler toward the posterior mean, which distortion metrics reward and perceptual metrics penalize. A $20\times$ reduction in inference cost is therefore available at a bounded perceptual cost, and a $2\times$ reduction at none.

Flow matching backbones. Tables 2 and 3 report results on CelebA and AFHQ-Cat, where the pretrained field is a velocity rather than a score. SIMCA is applied unchanged, which is itself a strong result as it bridges the gap between training-free diffusion and flow-matching methods.

On CelebA, SIMCA attains the best value on fourteen of the fifteen columns, improving on the strongest baseline, Flower (Pourya et al., 2026), on every task, with PSNR gains of 1.7 dB on random inpainting and 1.2 dB on box inpainting. On AFHQ-Cat the gains are more nuanced: SIMCA leads on both inpainting tasks across all three metrics (Figure 2) and on super-resolution PSNR and SSIM, while PnP-GS, Flower and OT-ODE perform best on denoising and deblurring. We refer the reader to Appendix D for a deeper analysis.

Where SIMCA does not help. On ImageNet Gaussian deblurring, SIMCA trails the strongest baselines on every metric, reaching 25.28 dB against 28.06 for DDNM, and the same pattern appears on AFHQ-Cat deblurring and denoising. The improvement over DPS is the largest in the table, from 20.31 to 25.28 dB, so the residual gap reflects a Gaussian likelihood that is genuinely limiting on these operators.

Table 1: Reconstruction quality on FFHQ and ImageNet. Within each task, best values are in **Bold** and second best are underlined. SIMCA-K denotes calibration and sampling with K steps.

Task	Method	FFHQ				ImageNet			
		PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓
SR×4	DAPS	29.07	0.818	0.177	51.44	25.89	0.694	0.276	**83.57**
	DPS	25.86	0.753	0.269	81.07	21.13	0.489	0.361	106.32
	DDRM	26.58	0.782	0.282	79.25	22.62	0.521	0.324	103.85
	DDNM	28.03	0.795	0.197	64.62	23.96	0.604	0.475	98.62
	FPS-SMC	28.42	0.813	0.204	49.25	24.82	0.703	0.313	97.51
	DiffPIR	26.64	-	0.260	65.77	23.18	-	0.371	106.32
	SIMCA-1k	29.12	0.821	0.145	46.39	26.14	0.712	0.241	97.48
	SIMCA-500	29.26	0.826	**0.143**	**43.14**	26.17	0.715	**0.239**	93.57
	SIMCA-50	**30.00**	**0.853**	0.144	49.02	**26.45**	**0.726**	0.244	107.25
Box Inpainting	DAPS	24.07	0.814	0.133	43.10	21.43	0.725	0.214	109.85
	DPS	22.51	0.792	0.209	61.27	18.94	0.722	0.257	126.52
	DDRM	22.26	0.801	0.207	78.62	18.63	0.733	0.254	116.37
	DDNM	24.47	0.837	0.235	46.59	21.64	0.748	0.319	**103.97**
	FPS-SMC	**24.86**	0.823	0.146	48.34	**22.16**	0.726	0.208	111.58
	SIMCA-1k	22.90	0.846	0.106	43.68	19.56	0.786	**0.185**	135.29
	SIMCA-500	23.28	0.850	**0.105**	**39.77**	19.63	**0.789**	0.186	140.73
	SIMCA-50	23.42	**0.855**	0.110	47.12	19.86	0.778	0.210	146.87
Random Inpainting	DAPS	31.12	0.844	0.098	32.17	28.44	0.775	0.135	54.25
	DPS	25.46	0.823	0.203	69.20	23.52	0.745	0.297	87.53
	DDNM	29.91	0.817	0.121	44.37	**31.16**	0.841	0.191	63.84
	FPS-SMC	28.21	0.823	0.261	61.23	24.52	0.701	0.316	79.12
	SIMCA-1k	32.91	0.916	0.069	19.16	29.22	0.851	**0.109**	**32.08**
	SIMCA-500	33.07	0.920	**0.068**	**18.31**	29.30	**0.853**	0.110	33.69
	SIMCA-50	**33.76**	**0.930**	0.072	24.88	29.46	0.852	0.132	46.08
Gaussian Deblurring	DAPS	**29.19**	**0.817**	0.165	**53.33**	26.15	0.684	**0.253**	**75.68**
	DPS	25.87	0.764	0.219	79.75	20.31	0.598	0.397	116.42
	DDRM	24.93	0.732	0.239	92.43	21.26	0.564	0.443	146.89
	DDNM	28.20	0.804	0.216	57.83	**28.06**	**0.703**	0.278	81.43
	FPS-SMC	26.54	0.773	0.253	67.45	23.91	0.601	0.387	91.72
	DiffPIR	27.36	-	0.236	59.65	22.80	-	0.355	93.36
	SIMCA-1k	28.23	0.794	0.163	57.07	25.28	0.668	0.281	121.85
	SIMCA-500	28.29	0.798	**0.161**	53.60	25.19	0.672	0.272	120.68
	SIMCA-50	28.33	0.812	0.165	58.54	25.02	0.663	0.294	141.21
Motion Deblurring	DAPS	29.66	0.847	0.157	**39.49**	**27.86**	0.766	**0.196**	**61.83**
	DPS	24.52	0.801	0.246	65.23	18.96	0.629	0.423	137.81
	FPS-SMC	27.39	0.826	0.227	48.32	24.52	0.647	0.326	87.43
	DiffPIR	26.57	-	0.255	65.78	24.01	-	0.366	94.63
	SIMCA-1k	30.55	0.850	0.122	43.22	27.67	0.771	0.210	83.00
	SIMCA-500	**30.64**	0.855	**0.121**	40.45	27.67	**0.774**	0.211	86.42
	SIMCA-50	30.29	**0.863**	0.131	43.60	26.77	0.753	0.236	112.84

Table 2: Results on 100 test images of the dataset CelebA.

	Denoising			Deblurring			Super-resolution			Random inpainting			Box inpainting		
Method	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
PnP-GS	**32.64**	0.910	0.035	34.03	0.924	0.041	31.31	0.892	0.064	29.22	0.875	0.070	–	–	–
DiffPIR	31.20	0.885	0.060	32.77	0.912	0.060	31.52	0.895	0.033	31.74	0.917	0.025	–	–	–
OT-ODE	30.54	0.859	0.032	33.01	0.921	0.029	31.46	0.907	0.025	28.68	0.871	0.051	29.40	0.920	0.038
D-Flow	26.04	0.607	0.092	31.25	0.854	0.038	30.47	0.843	0.026	33.67	0.943	0.015	30.70	0.899	0.026
Flow-Priors	29.34	0.768	0.134	31.54	0.858	0.056	28.35	0.713	0.102	32.88	0.871	0.019	30.07	0.858	0.048
PnP-Flow	31.80	0.905	0.044	34.48	0.936	0.040	31.09	0.902	0.045	33.05	0.944	0.018	30.47	0.933	0.037
Flower-OT	32.28	0.914	0.034	34.98	**0.947**	0.026	32.36	0.923	0.034	33.08	0.944	0.018	31.19	0.945	0.022
SIMCA (ren.)	**32.64**	**0.919**	**0.020**	**35.17**	**0.947**	0.023	**33.02**	**0.929**	0.021	**34.80**	**0.962**	**0.012**	**32.40**	**0.955**	**0.018**
SIMCA + OT (ren.)	32.42	0.914	0.021	34.87	0.943	**0.022**	32.89	**0.929**	**0.019**	34.40	0.960	0.013	31.77	0.951	0.020
SIMCA (eul.)	31.83	0.897	0.026	33.30	0.927	0.031	31.45	0.910	0.025	31.10	0.928	0.022	29.45	0.935	0.026
SIMCA + OT (eul.)	31.65	0.895	0.028	33.09	0.924	0.033	31.26	0.908	0.026	30.77	0.924	0.025	28.96	0.933	0.028

5.3 Ablation studies

Calibration set size. Table 5 varies the number of calibration images on FFHQ random and box inpainting. A single image already suffices. This is expected, since only two scalars are fitted per time step while each image contributes a full-dimensional inner product. Calibration is correspondingly cheap: solving for one weight pair takes 0.11 s from a single image and 6.76 s from one hundred, so a full schedule is obtained in seconds to minutes.

Simulation-free versus simulation-based calibration. Table 4 compares the two strategies of Section 4 on FFHQ random inpainting. Both improve substantially on the uncalibrated sampler. SIMCA is nonetheless uniformly better, with the largest margin on the perceptual metrics: for DPS,

Table 3: Results on 100 test images of the dataset AFHQ-Cat.

	Denoising			Deblurring			Super-resolution			Random inpainting			Box inpainting		
Method	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
PnP-GS	**32.58**	**0.894**	0.072	27.91	0.753	0.349	24.15	0.632	0.362	29.42	0.836	0.126	–	–	–
DiffPIR	30.58	0.835	0.189	27.56	0.728	0.342	23.65	0.624	0.402	31.70	0.881	0.062	–	–	–
OT-ODE	30.03	0.815	0.076	27.06	0.713	**0.123**	25.91	0.716	**0.108**	29.40	0.839	0.090	24.62	0.875	0.085
D-Flow	26.13	0.574	0.175	27.82	0.721	0.164	24.64	0.601	0.190	32.20	0.894	0.040	26.26	0.842	0.077
Flow-Priors	29.41	0.763	0.153	26.47	0.700	0.181	23.51	0.570	0.272	32.37	0.906	0.047	26.20	0.818	0.118
PnP-Flow	31.18	0.863	0.135	27.87	0.760	0.304	26.94	0.763	0.171	33.00	0.918	0.037	26.00	0.897	0.103
Flower-OT	31.69	0.879	0.102	**28.64**	**0.775**	0.255	26.23	0.741	0.272	32.97	0.918	0.040	26.19	0.915	0.063
SIMCA (ren.)	31.58	0.880	**0.054**	27.69	0.757	0.245	**27.18**	**0.776**	0.169	**33.53**	**0.926**	**0.031**	**27.46**	**0.921**	**0.049**
SIMCA + OT (ren.)	31.53	0.876	0.061	28.12	0.767	0.269	24.29	0.754	0.205	33.34	0.924	0.036	26.87	0.919	0.058
SIMCA (eul.)	31.40	0.864	0.060	26.64	0.680	0.169	26.14	0.710	0.125	31.57	0.894	0.039	24.85	0.899	0.055
SIMCA + OT (eul.)	31.44	0.864	0.060	26.47	0.668	0.174	25.98	0.705	0.129	31.55	0.893	0.042	24.47	0.739	0.058

Table 4: Calibration applied to three measurement-consistency terms, on FFHQ random inpainting.

	PSNR	SSIM	LPIPS	FID
DPS	25.46	0.823	0.203	69.20
+ sim.-free	31.69	0.897	0.103	33.71
+ SIMCA	32.91	0.916	0.069	19.16
ΠGDM	25.10	0.719	0.248	67.93
+ sim.-free	32.12	0.903	0.096	30.68
+ SIMCA	32.97	0.916	0.069	18.35
TMPD	27.89	0.835	0.156	56.43
+ sim.-free	32.02	0.879	0.115	30.72
+ SIMCA	32.84	0.915	0.070	20.21

Table 5: Effect of the calibration set size on FFHQ, with the compute time per timestep.

	Random inp.		Box inp.		
N	LPIPS	FID	LPIPS	FID	Time (s)
1	0.069	19.03	0.107	44.13	0.11
2	0.069	19.14	0.106	43.84	0.18
5	0.069	19.18	0.106	43.61	0.38
10	0.069	19.18	0.106	43.79	0.68
20	0.069	19.20	0.106	43.62	1.35
50	0.069	19.17	0.106	43.71	3.37
100	0.069	19.16	0.106	43.68	6.76

LPIPS falls from 0.103 to 0.069 and FID from 33.71 to 19.16, a 43% reduction for the same number of samples, and the same ordering holds for ΠGDM and TMPD.

Choice of the measurement-consistency term. Table 4 also compares different measurement-consistency terms: the isotropic Gaussian likelihood of DPS which all other experiments are based on (see Eq. 8), the pseudoinverse-based covariance correction of ΠGDM, and the Tweedie moment projection of TMPD. Uncalibrated, they span 25.10 to 27.89 dB and 0.156 to 0.248 LPIPS, in the order the literature reports. After simulation-based calibration they become nearly indistinguishable: 32.91, 32.97, and 32.84 dB, with LPIPS of 0.069, 0.069, and 0.070. TMPD is the telling case, leading by 2.4 dB uncalibrated and ending within 0.07 dB of DPS.

The advantage of a more elaborate surrogate over the isotropic Gaussian one is therefore, in these settings, an advantage in effective scaling rather than in expressiveness, and it almost vanishes after calibration with SIMCA.

Sampler and discretization. Recalibrating SIMCA for four samplers across seven step counts (Appendix E) shows two patterns. Stochastic samplers dominate the deterministic ones at every budget. They also degrade differently: DDPM improves in PSNR down to 50 steps and only fails below 20, whereas Euler-Maruyama collapses below 100 steps, losing 14 dB by 20. Each cell is calibrated at its own discretization, which is what makes the low-step regime accessible.

6 Conclusion

In this work, we introduced SIMCA, a simple and principled offline strategy for tuning guidance weights in training-free posterior sampling. By exploiting the least-squares structure of conditional score- and flow-matching objectives, our method reduces guidance optimization to a two-dimensional linear least-squares problem at each sampling step. The resulting weights are optimized once for a given inverse problem setting, without retraining or fine-tuning the underlying generative model. Evaluated across four datasets and multiple inverse problems, using both diffusion- and flow-matching methods, our approach consistently matches or outperforms state-of-the-art methods.

USE OF AI TOOLS

In this work, we used a large language model to aid and polish writing. Its use was limited to rephrasing and condensing author-written text, drafting figure and table captions, and assisting with LaTeX formatting. We did not use generative AI tools for research ideation, for deriving or verifying the theoretical results, or for writing the code used in our experiments.

REFERENCES

Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*, 2021.

Heli Ben-Hamu, Omri Puny, Itai Gat, Brian Karrer, Uriel Singer, and Yaron Lipman. D-flow: Differentiating through flows for controlled generation. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 3462–3483. PMLR / OpenReview.net, 2024.

Benjamin Boys, Mark Girolami, Jakiw Pidstrigach, Sebastian Reich, Alan Mosca, and O Deniz Akyildiz. Tweedie moment projected diffusions for inverse problems. *arXiv preprint arXiv:2310.06721*, 2023.

Gabriel Cardoso, Sylvain Le Corff, Eric Moulines, et al. Monte carlo guided denoising diffusion models for bayesian linear inverse problems. In *International Conference on Learning Representations*, volume 2024, pp. 44001–44037, 2024.

Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022a.

Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints. *Advances in neural information processing systems*, 35:25683–25696, 2022b.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

Zehao Dou and Yang Song. Diffusion posterior sampling for linear inverse problem solving: A filtering perspective. In *International Conference on Learning Representations*, volume 2024, pp. 12069–12095, 2024.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

Ruiqi Gao, Emiel Hoogeboom, Jonathan Heek, Valentin De Bortoli, Kevin P. Murphy, and Tim Salimans. Diffusion meets flow matching: Two sides of the same coin. 2024. URL `https://diffusionflow.github.io/`.

Alexandros Graikos, Nikolay Malkin, Nebojsa Jojic, and Dimitris Samaras. Diffusion models as plug-and-play priors. *Advances in Neural Information Processing Systems*, 35:14715–14728, 2022.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Samuel Hurault, Arthur Leclaire, and Nicolas Papadakis. Gradient step denoiser for convergent plug-and-play. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 4396–4405. IEEE, 2019.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. *Advances in neural information processing systems*, 35:23593–23606, 2022a.

Bahjat Kawar, Jiaming Song, Stefano Ermon, and Michael Elad. Jpeg artifact correction using denoising diffusion restoration models. *arXiv preprint arXiv:2209.11888*, 2022b.

Filip Ekström Kelvinius, Zheng Zhao, and Fredrik Lindsten. Solving linear-gaussian bayesian inverse problems with decoupled diffusion sequential monte carlo. *arXiv preprint arXiv:2502.06379*, 2025.

Jeongsol Kim, Bryan Sangwoo Kim, and Jong Chul Ye. Flowdps: Flow-driven posterior sampling for inverse problems. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12328–12337. IEEE, 2025.

Henry Li and Marcus Pereira. Solving inverse problems via diffusion optimal control. *Advances in Neural Information Processing Systems*, 37:73549–73571, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Qiang Liu. Rectified flow: A marginal preserving approach to optimal transport. *arXiv preprint arXiv:2209.14577*, 2022.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Dawid Malarz, Artur Kasymov, Maciej Zieba, Jacek Tabor, and Przemysław Spurek. Classifier-free guidance with adaptive scaling. In *ECAI*, pp. 435–442. SAGE Publications 1 Oliver's Yard, 55 City Road, London, EC1Y 1SP, 2025.

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*, volume 2024, pp. 28027–28053, 2024.

Ségolène Martin, Anne Gagneux, Paul Hagemann, and Gabriele Steidl. Pnp-flow: Plug-and-play image restoration with flow matching. In *International Conference on Learning Representations*, volume 2025, pp. 45466–45492, 2025.

Mang Ning, Enver Sangineto, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Input perturbation reduces exposure bias in diffusion models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 26245–26265. PMLR, 2023.

Mang Ning, Mingxiao Li, Jianlin Su, Albert Ali Salah, and Itir Onal Ertugrul. Elucidating the exposure bias in diffusion models. In *International Conference on Learning Representations*, volume 2024, pp. 15167–15189, 2024.

Xinyu Peng, Ziyang Zheng, Wenrui Dai, Nuoqian Xiao, Chenglin Li, Junni Zou, and Hongkai Xiong. Improving diffusion models for inverse problems using optimal posterior covariance. In *ICML*, volume 235 of *Proceedings of Machine Learning Research*, pp. 40347–40370. PMLR / OpenReview.net, 2024.

Ashwini Pokle, Matthew J. Muckley, Ricky T. Q. Chen, and Brian Karrer. Training-free linear image inverses via flows. *Trans. Mach. Learn. Res.*, 2024, 2024.

Mehrsa Pourya, Bassam El Rawas, and Michael Unser. Flower: A flow-matching solver for inverse problems. In *International Conference on Learning Representations*, volume 2026, pp. 91318–91343, 2026.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International conference on learning representations*, 2023.

Kaiyu Song, Hanjiang Lai, Yan Pan, Kun Yue, and Jian Yin. Flow matching posterior sampling: A training-free conditional generation for flow matching. *IEEE Transactions on Image Processing*, 2026.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging with score-based generative models. *arXiv preprint arXiv:2111.08005*, 2021.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482*, 2023.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

Lingyu Wang and Xiangming Meng. Saip: A plug-and-play scale-adaptive module in diffusion-based inverse problems. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 9457–9461. IEEE, 2026.

Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. *arXiv preprint arXiv:2212.00490*, 2022.

Bingliang Zhang, Wenda Chu, Julius Berner, Chenlin Meng, Anima Anandkumar, and Yang Song. Improving diffusion inverse problem solving with decoupled noise annealing. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20895–20905. IEEE, 2025.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Yasi Zhang, Peiyu Yu, Yaxuan Zhu, Yingshan Chang, Feng Gao, Ying N Wu, and Oscar Leong. Flow priors for linear inverse problems via iterative corrupted trajectory matching. *Advances in Neural Information Processing Systems*, 37:57389–57417, 2024.

Yuanzhi Zhu, Kai Zhang, Jingyun Liang, Jiezhang Cao, Bihan Wen, Radu Timofte, and Luc Van Gool. Denoising diffusion models for plug-and-play image restoration. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1219–1229. IEEE, 2023.

APPENDIX

A EXTENDED RELATED WORK

Control formulation. Li & Pereira (2024) derive a controller that recovers the idealized posterior sampling equation as a special case. Such methods optimize a control signal of full dimension, which requires differentiating through the sampling dynamics at substantial cost. This places them outside the regime we target, where a practitioner needs the guidance to be set in seconds for each new measurement setup.

Exposure bias. SIMCA addresses the mismatch between the forward marginals and the states visited at inference, known as exposure bias in diffusion models (Ning et al., 2023; 2024). Epsilon scaling (Ning et al., 2024) rescales the network output with a hand-tuned schedule, an unconditional analogue of our prior weight a_t, which we instead obtain from an explicit criterion.

B CALIBRATED WEIGHTS

Figures 3 and 4 report the calibrated weights $(w_{u,t}, w_{f,t})$ for additional datasets and operators, on the diffusion and flow-matching backbones respectively.

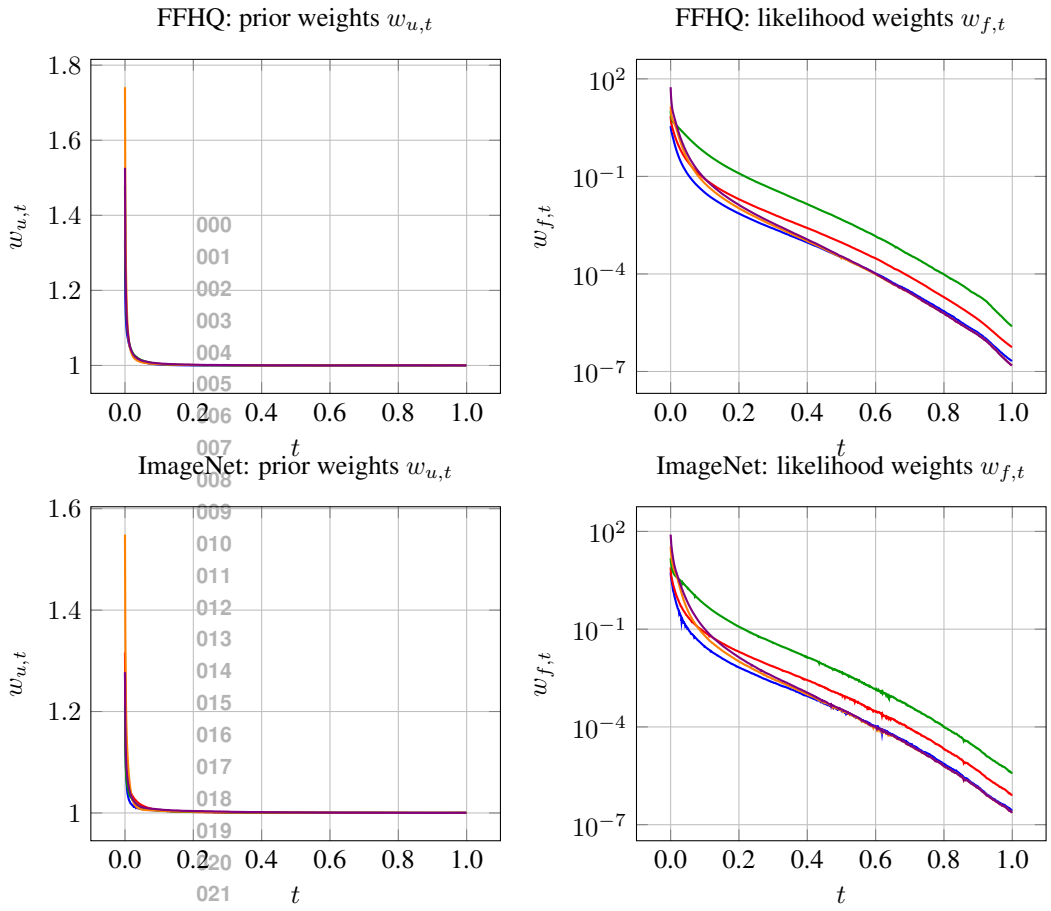


Figure 3: SIMCA calibrated prior weights $w_{u,t}$ (left column) and likelihood weights $w_{f,t}$ (right column, log scale) over diffusion step t, for FFHQ (top row) and ImageNet (bottom row). Colors indicate task: Box inpainting, Random inpainting, Super resolution, Motion blur, and Gaussian blur.

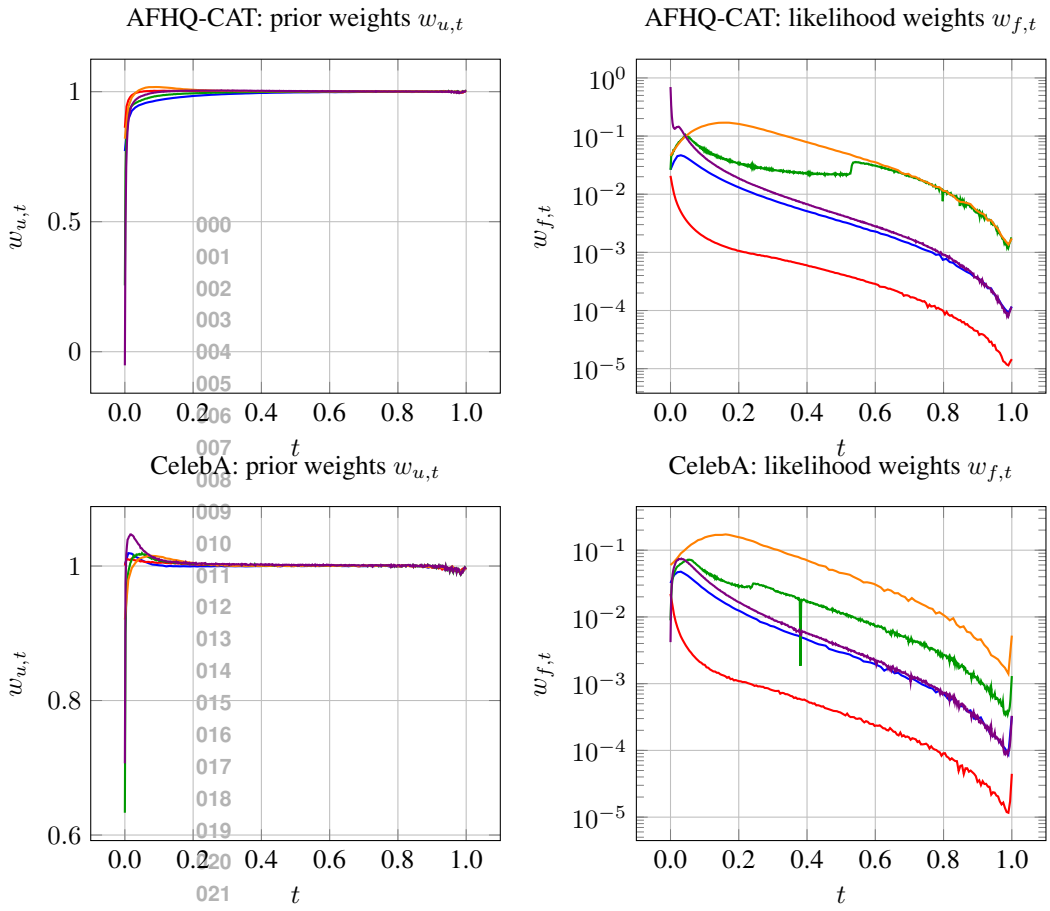


Figure 4: SIMCA calibrated prior weights $w_{u,t}$ (left column) and likelihood weights $w_{f,t}$ (right column, log scale) over diffusion step t, for AFHQ-CAT (top row) and CelebA (bottom row). Colors indicate task: Box inpainting, Random inpainting, Super resolution, Denoising, and Gaussian blur.

C Effect of the step count

Figures 5 to 9 show SIMCA reconstructions at $K = 1000$, 500 and 50 steps, recalibrated at each discretization, on the five diffusion tasks. On the four well-posed operators (Gaussian deblurring, motion deblurring, super-resolution, random inpainting) the three columns are visually almost indistinguishable, which is the qualitative counterpart of the metric behaviour reported in Section 5.2: reducing the budget twentyfold leaves reconstruction quality essentially unchanged. Box inpainting behaves differently, and instructively so. There the three budgets produce visibly different completions of the masked region, since that region is not determined by the measurement, while the unmasked part remains identical across columns. The step count therefore acts as a choice among plausible posterior samples rather than as a loss of fidelity, and the 50-step reconstructions are no less plausible than the 1000-step ones, consistent with their higher PSNR and slightly worse FID.

D Robustness to the backbone and the sampler.

In Tab. 2 and Tab. 3, two comparisons isolate what the gains depend on. First, in the flow-matching experiments, the OT-coupled and independently coupled backbones behave almost identically once calibrated, differing by at most 0.2 dB and 0.002 LPIPS on CelebA. Second, the renoising sampler outperforms the plain Euler integrator by 3.7 dB on CelebA random inpainting and 3.0 dB on box inpainting. This matches the deterministic-sampler penalty of Section 5.3 and is a property of the sampler rather than of calibration: the Euler rows still improve on most flow matching baselines.

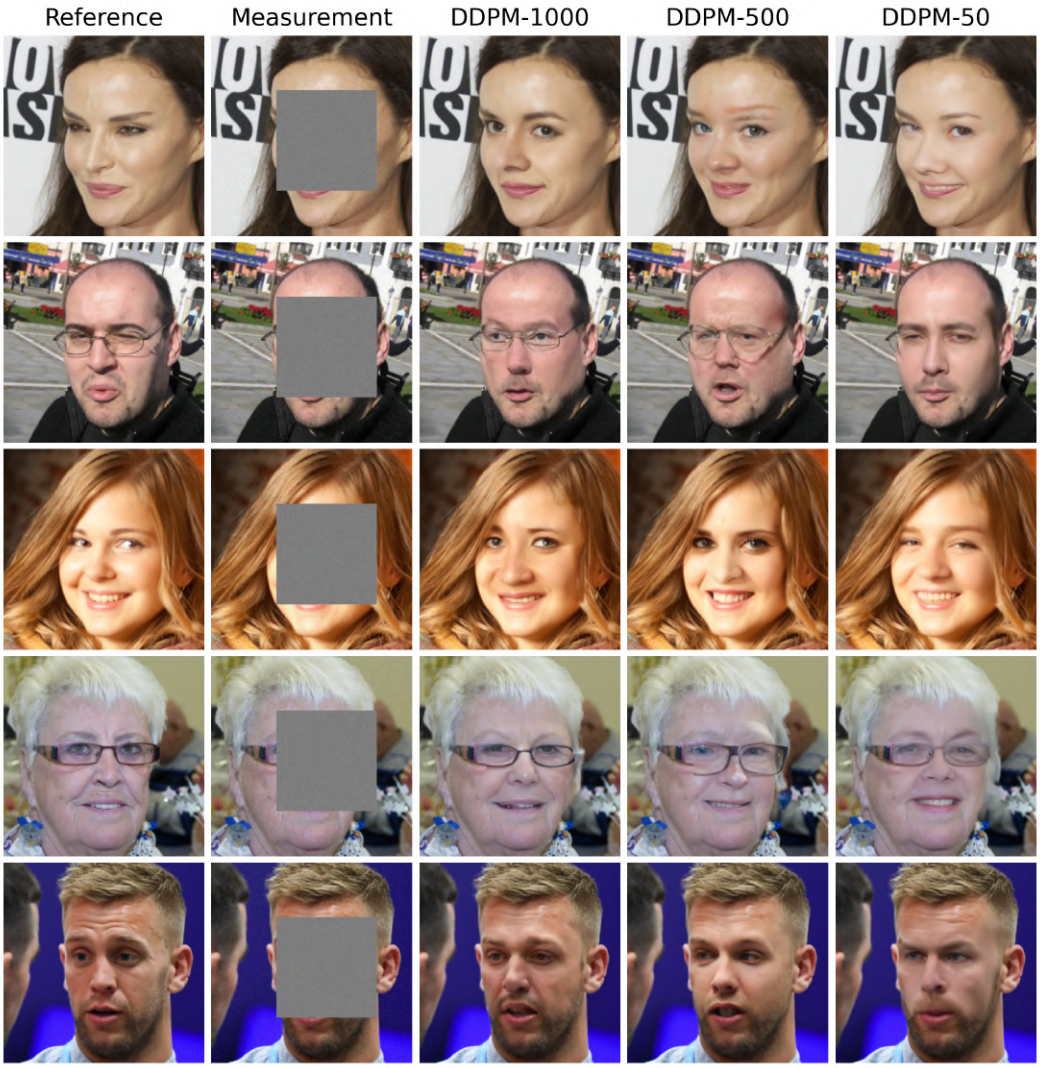


Figure 5: SIMCA on FFHQ box inpainting at $K = 1000$, 500 and 50 steps, recalibrated at each discretization.

E Sampler and discretization

Table 6 reports SIMCA recalibrated for four samplers across seven step counts on FFHQ. The two families degrade very differently as the budget shrinks: DDPM improves in PSNR down to 50 steps and only fails below 20, whereas Euler-Maruyama collapses below 100 steps, since its explicit discretization of the reverse SDE is unstable at large step sizes. DDIM degrades gracefully, remaining usable at 10 steps, at a consistently lower level throughout. The practical recommendation is DDPM with recalibrated weights at 50 steps, which gives the best distortion metrics of the table on random inpainting and super-resolution at a twentieth of the nominal cost. The table also quantifies the dependence noted in Section 4: since the weights depend on the sampler Φ and on the number of timesteps, transferring them across these rows rather than recalibrating would forfeit the gains.

F Experiments - technical details

Backbones and protocols. For diffusion, we use the pixel-space DDPM checkpoints for FFHQ and ImageNet at 256×256 released by (Chung et al., 2022a), following the evaluation protocol of

Table 6: Effect of the sampler and the discretization, on FFHQ. SIMCA is recalibrated independently for each sampler and step count K. Stochastic samplers (DDPM, Euler-Maruyama) outperform their deterministic counterparts (DDIM, Euler-ODE) at every budget, and DDPM alone remains stable down to $K = 50$ where it attains the best distortion metrics of the table on random inpainting and super-resolution.

		Random Inpainting				Box Inpainting				SRx4			
Samplers	K	PSNR	SSIM	LPIPS	FID	PSNR	SSIM	LPIPS	FID	PSNR	SSIM	LPIPS	FID
DDPM	1000	32.91	0.916	0.069	19.16	22.90	0.846	0.106	43.68	29.12	0.821	0.145	46.39
	500	33.07	0.920	0.068	18.31	23.28	0.850	0.105	39.77	29.26	0.826	0.143	43.14
	200	33.34	0.926	0.068	20.76	23.28	0.853	0.106	43.26	29.48	0.836	0.141	44.51
	100	33.62	0.930	0.068	23.04	23.36	0.856	0.108	44.52	29.71	0.845	0.140	45.50
	50	33.76	0.930	0.072	24.88	23.43	0.855	0.110	47.12	30.00	0.853	0.144	49.02
	20	33.11	0.918	0.090	33.60	22.94	0.847	0.122	51.02	30.12	0.855	0.156	53.89
	10	31.06	0.886	0.132	59.91	23.13	0.841	0.140	54.67	29.50	0.833	0.195	74.42
Euler-SDE	1000	32.88	0.915	0.069	18.99	22.90	0.845	0.106	44.07	29.11	0.820	0.146	46.76
	500	33.04	0.919	0.068	18.41	23.25	0.848	0.105	40.67	29.26	0.825	0.144	43.60
	200	32.97	0.914	0.075	23.14	23.12	0.837	0.114	46.17	29.42	0.828	0.145	43.01
	100	32.20	0.883	0.109	31.91	22.88	0.808	0.143	57.17	29.34	0.809	0.160	45.54
	50	30.03	0.780	0.198	51.64	22.11	0.731	0.201	84.24	28.44	0.723	0.220	53.39
	20	24.57	0.462	0.404	96.08	19.69	0.535	0.352	186.90	24.77	0.462	0.382	87.39
	10	18.43	0.192	0.618	184.61	17.46	0.315	0.505	270.25	20.79	0.262	0.512	122.34
DDIM	1000	31.56	0.901	0.094	32.02	21.58	0.825	0.136	63.93	28.50	0.797	0.176	58.31
	500	32.04	0.908	0.086	28.35	21.97	0.832	0.125	55.03	28.65	0.796	0.174	56.06
	200	32.24	0.913	0.084	28.26	21.95	0.833	0.125	55.07	28.70	0.794	0.177	55.97
	100	32.49	0.917	0.083	28.54	21.95	0.833	0.126	55.11	28.81	0.795	0.179	55.87
	50	32.69	0.918	0.084	29.88	21.97	0.832	0.128	55.07	28.86	0.788	0.185	55.22
	20	32.08	0.905	0.106	41.88	21.99	0.828	0.139	57.09	28.44	0.741	0.224	61.46
	10	29.99	0.866	0.157	69.46	21.90	0.820	0.156	62.81	27.48	0.674	0.279	80.06
Euler-ODE	1000	31.77	0.902	0.090	29.32	21.82	0.827	0.127	56.81	28.58	0.787	0.187	57.18
	500	31.66	0.900	0.092	30.22	21.83	0.825	0.126	58.04	28.48	0.779	0.188	56.15
	200	31.27	0.889	0.101	34.71	21.81	0.814	0.138	64.12	28.27	0.760	0.193	54.63
	100	30.38	0.856	0.136	44.44	21.68	0.781	0.170	81.52	27.98	0.733	0.204	54.73
	50	28.16	0.734	0.236	64.77	21.13	0.694	0.252	128.57	27.22	0.664	0.246	60.99
	20	22.16	0.359	0.479	108.41	21.89	0.438	0.463	240.49	23.34	0.376	0.434	95.71
	10	15.92	0.119	0.684	213.88	15.73	0.189	0.632	280.34	17.90	0.150	0.616	150.32

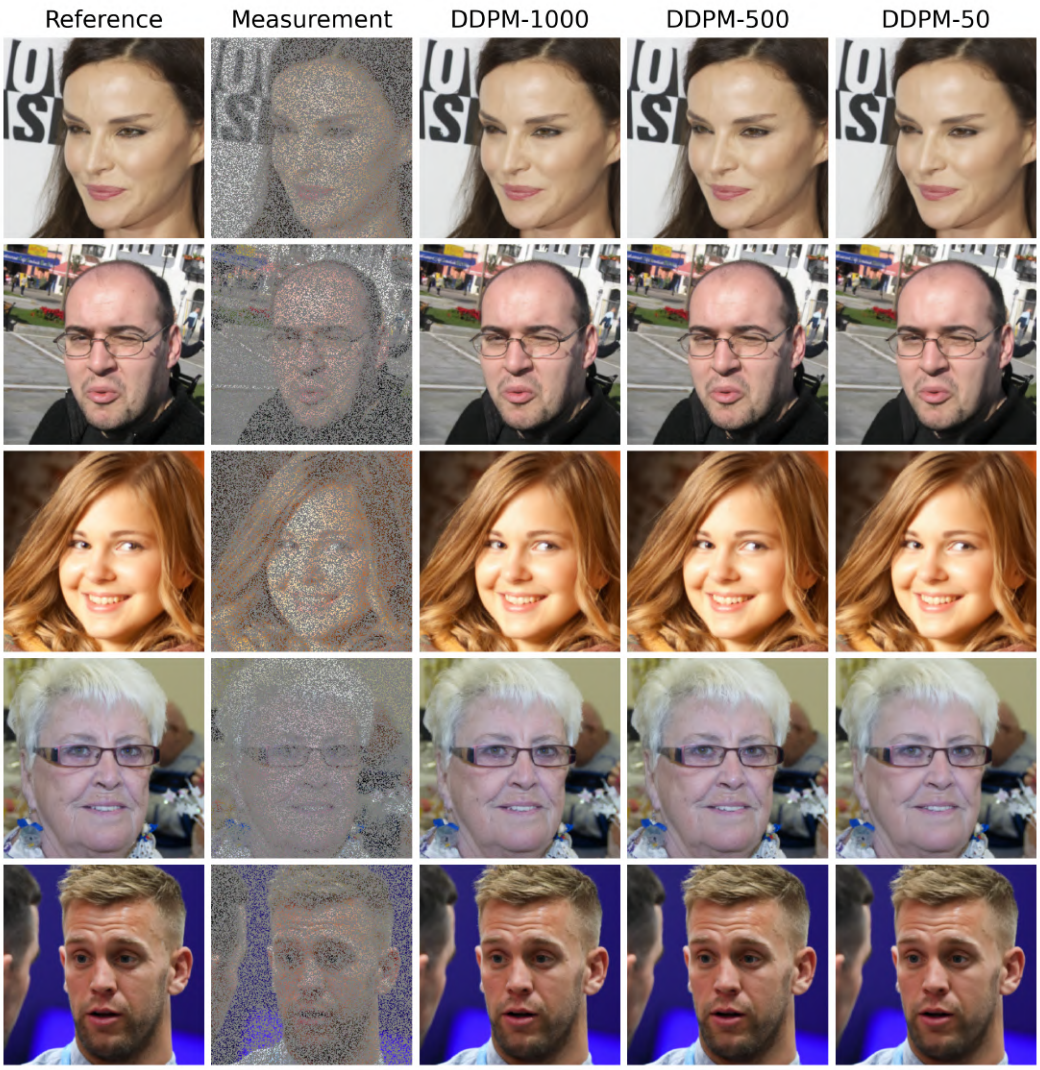


Figure 6: SIMCA on FFHQ random inpainting at $K = 1000$, 500 and 50 steps, recalibrated at each discretization.

DAPS (Zhang et al., 2025). For flow matching, we adopt the setup of Flower (Pourya et al., 2026), using the same U-Net velocity networks trained on CelebA and AFHQ-Cat. We evaluate both the OT-coupled network and its independently coupled variant, allowing us to assess calibration across two distinct pretrained velocity fields.

Datasets and metrics. We evaluate on 100 held-out images per dataset: FFHQ and ImageNet at 256×256 for diffusion, CelebA at 128×128 and AFHQ-Cat at 256×256 for flow matching. Images are scaled to $[-1, 1]$ for sampling and noise levels are expressed on this scale. We report PSNR, SSIM, and LPIPS (Zhang et al., 2018) on both benchmarks, and FID (Heusel et al., 2017) on the diffusion benchmark, averaged over individual images.

Inverse problems. On the diffusion side we consider super-resolution ($\times 4$ bicubic downsampling), Gaussian deblurring (61×61 kernel, intensity 3.0), motion deblurring (61×61 kernel, intensity 0.5), box inpainting (128×128 central mask), and random inpainting (70% masking rate), with additive white Gaussian noise of standard deviation $\sigma_m = 0.05$. On the flow-matching side we consider the five tasks of the Flower benchmark, with operators and noise levels unchanged: denoising, Gaussian deblurring, super-resolution, random inpainting, and box inpainting.

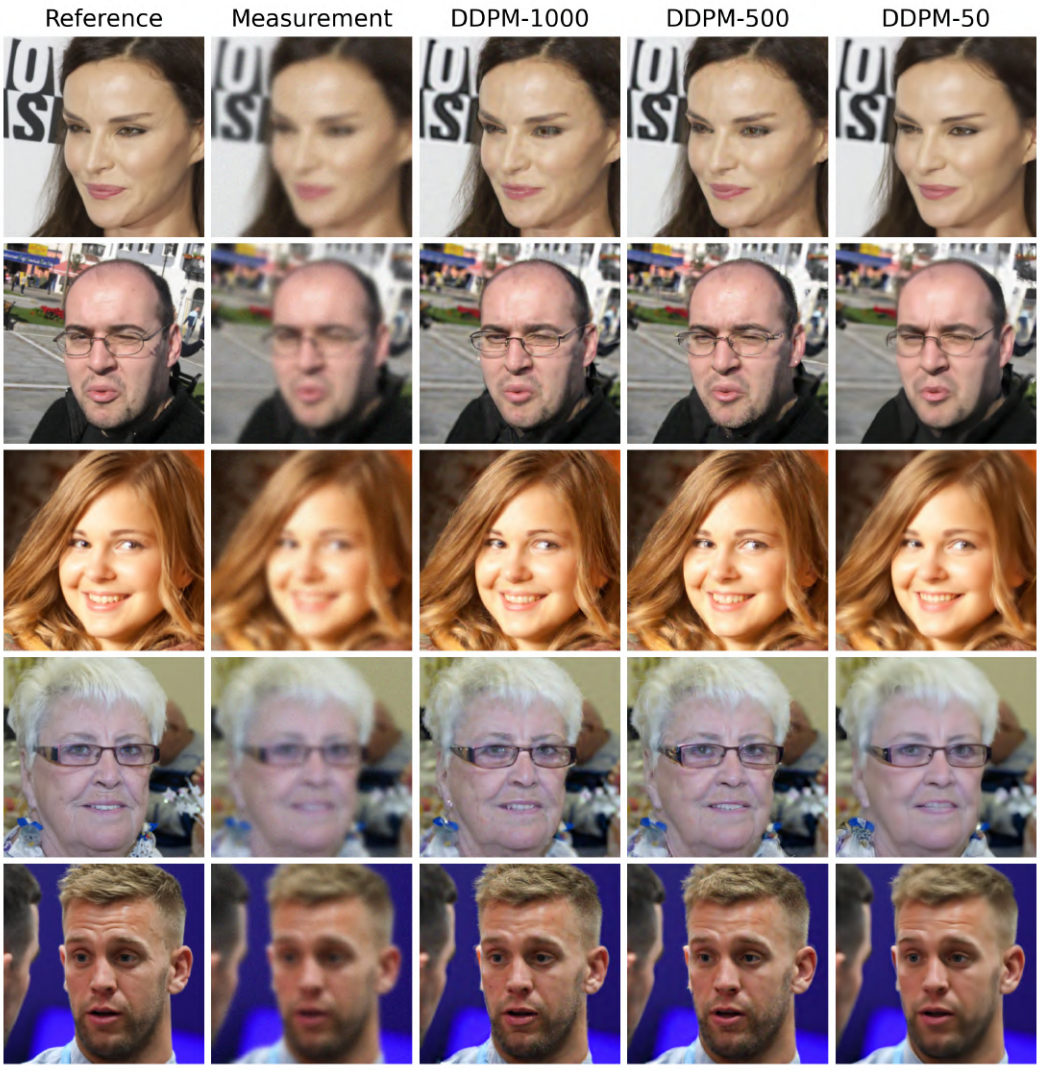


Figure 7: SIMCA on FFHQ Gaussian deblurring at $K = 1000$, 500 and 50 steps, recalibrated at each discretization.

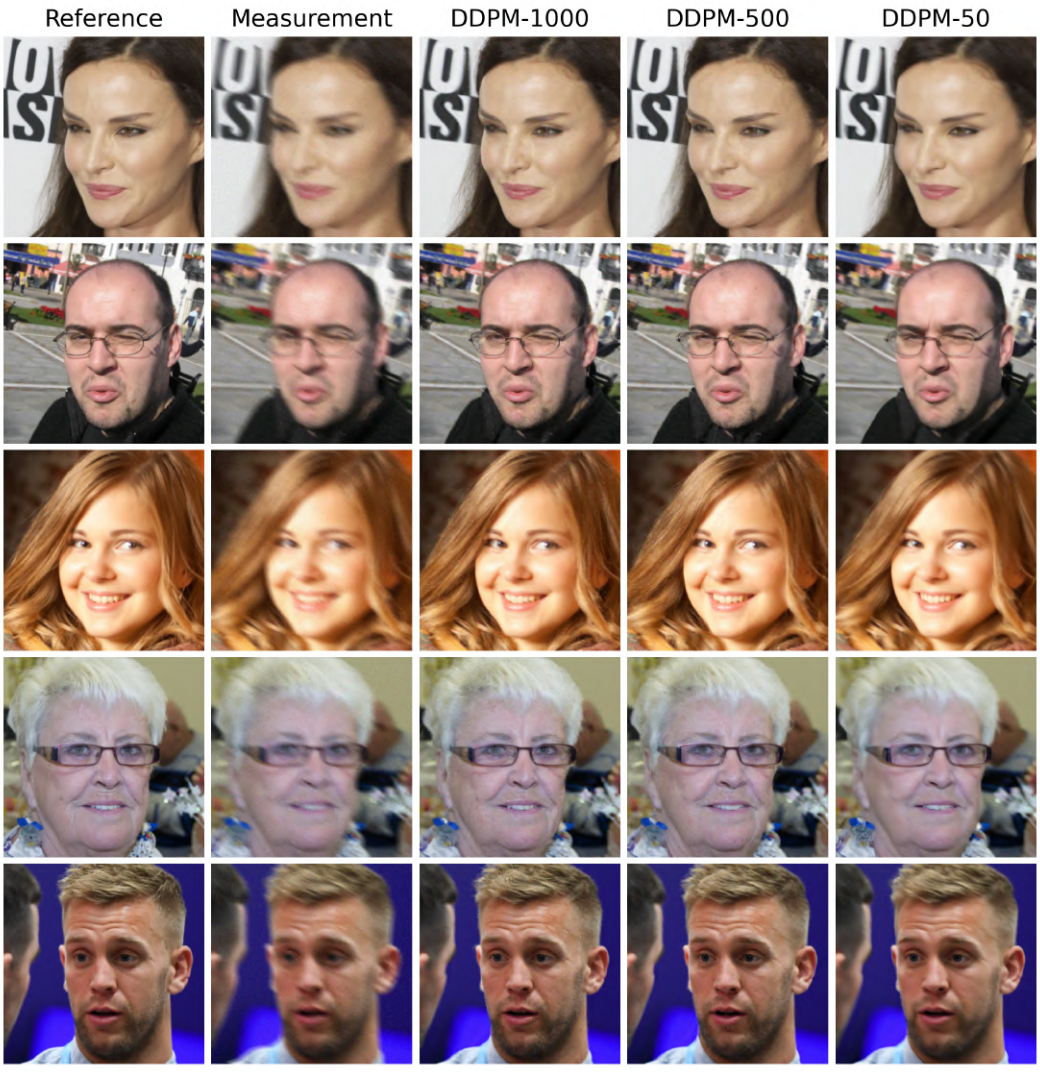


Figure 8: SIMCA on FFHQ motion deblurring at $K = 1000$, 500 and 50 steps, recalibrated at each discretization.

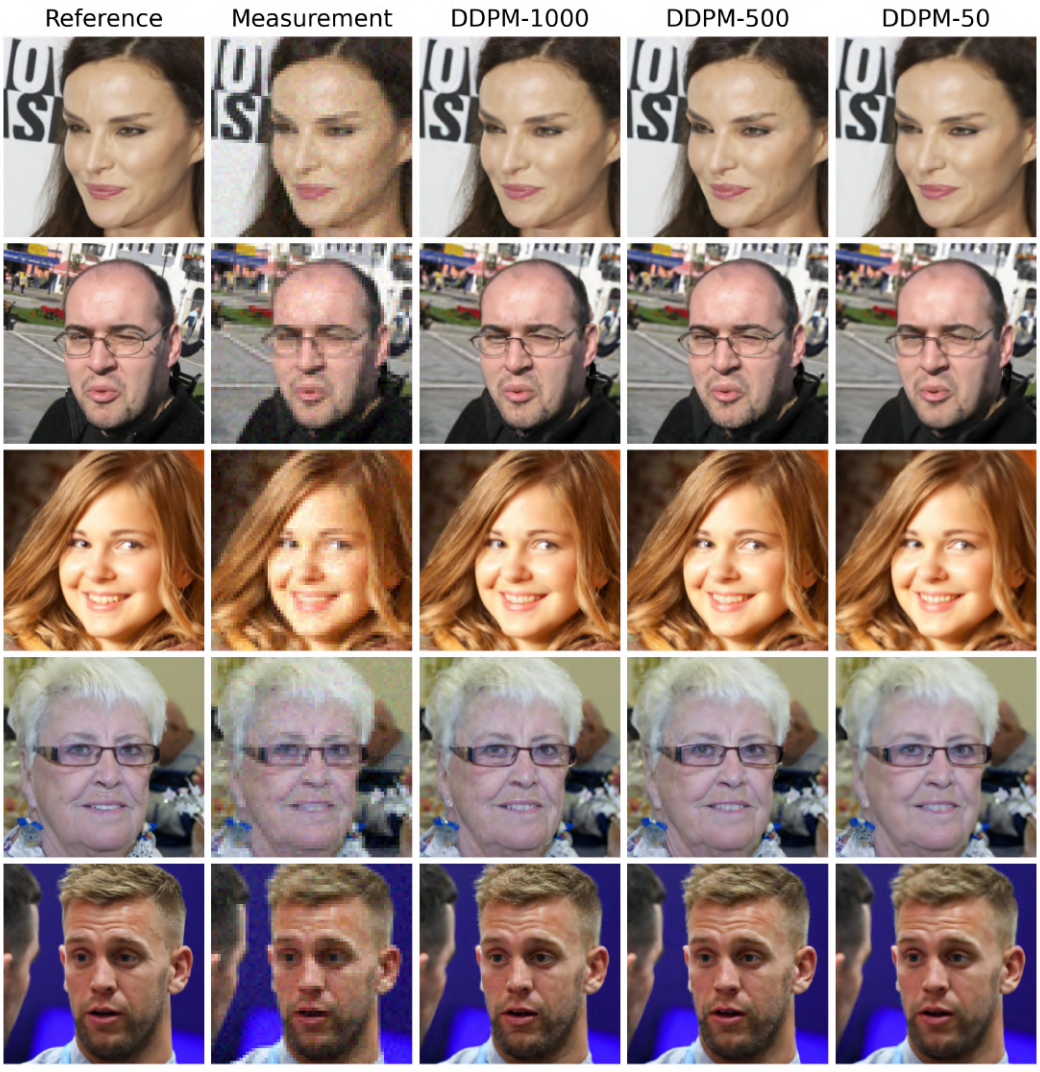


Figure 9: SIMCA on FFHQ super-resolution at $K = 1000$, 500 and 50 steps, recalibrated at each discretization.