Diffusion Based Unpaired Data Learning for Inverse Problems

Chenglong Bao* Yiming Dang† Chenguang Duan‡ Yuling Jiao§ Defeng Sun¶

Abstract

Data is important in many deep learning-based inverse problem solvers. However, obtaining sufficient paired data in many scenarios remains highly challenging, while unpaired data is cheap. To maximize data utilization, this paper proposes LUD-DIF, a diffusion-based approach for solving inverse problems with unpaired data. Starting from the evidence lower bound (ELBO) of the joint distribution, we decouple it into two independent diffusion processes under the weak-coupling assumption. The method provides theoretical support from a variational inference perspective, derives the loss function, quantitatively analyzes the error bound introduced by the assumption, and offers a theorem-motivated heuristic for hyperparameter selection. Experimental results demonstrate that LUD-DIF achieves outstanding performance on multiple image inverse problems, validating its effectiveness and generalization capability in unpaired inverse problem settings.

Keywords: unpaired data, diffusion, noise modeling, image restoration

2020 Mathematics Subject Classification: 65J22, 68T07, 94A08

1 Introduction

Inverse problems are widely encountered across numerous disciplines including computer vision [35], geophysical exploration [45], and biomedical imaging [6]. These problems share a classical mathematical framework:

$$y = \mathcal{A}(x) + \Xi,$$

where $\mathcal{A}$ denotes the forward observation operator and Ξ represents the observation noise. The goal of solving inverse problems is to robustly recover the unknown state x from the observed data y. However, in real-world scenarios, the noise distribution is often unknown, and the observation operators typically exhibit highly nonlinear or low-rank characteristics, leading to the typical ill-posedness of inverse problems [19]. Traditional methods typically rely on certain prior assumptions (such as sparsity, smoothness, or low-rank structures) [40, 38] or employ maximum a posteriori (MAP) estimation and Bayesian inference under the assumption of Gaussian noise [39].

In recent years, deep learning techniques have brought a paradigm shift to solving inverse problems by training deep neural networks to learn complex mappings from observations to the underlying states directly from data [20]. These learning-based approaches have surpassed traditional model-based regularization methods in many fields. They are capable of implicitly capturing high-dimensional and nonlinear data distributions and have achieved remarkable success in classical inverse problems such as image processing [32], seismic wave inversion [2], and X-ray imaging [11].

Nevertheless, the exceptional performance of such end-to-end learning methods heavily relies on a prerequisite that is difficult to satisfy in practical applications: the availability of a large number of paired samples $\{(x_i, y_i)\} \sim p(x, y)$ for supervised training. In some real-world scenarios, such as medical or astronomical imaging, acquiring paired data is prohibitively expensive or entirely infeasible. Researchers

*Yau Mathematical Sciences Center, Tsinghua University, Beijing, China (clbao@tsinghua.edu.cn).

†Department of Mathematical Sciences, Tsinghua University, Beijing, China (dym22@mails.tsinghua.edu.cn).

‡Institut für Geometrie und Praktische Mathematik, RWTH Aachen University, Aachen, Germany (duan@igpm.rwth-aachen.de).

§School of Artificial Intelligence, Wuhan University, Wuhan, Hubei, China (yulingjiaomath@whu.edu.cn).

¶Department of Applied Mathematics, The Hong Kong Polytechnic University, Hong Kong, China (defeng.sun@polyu.edu.hk).

are often restricted to accessing only independent ground-truth data $\{x_i\}_{i=1}^N \sim p(x)$ and observation data $\{y_j\}_{j=1}^M \sim p(y)$. This limitation has emerged as the primary bottleneck restricting the widespread application of deep learning in complex inverse problems. To alleviate this issue, some studies have attempted to perform self-supervised or unsupervised learning relying exclusively on the observation data $\{y_j\}$ [24, 22]; however, these approaches typically depend on strong heuristic priors. Another line of research utilizes only the ground-truth data $\{x_i\}$ to pre-train deep generative priors $p(x)$, which are subsequently combined with posterior sampling for problem resolution [8]. Yet, such methods generally require exact knowledge of both the forward observation operator and the noise model.

Unpaired learning attempts to simultaneously exploit information from the marginal distributions $p(x)$ and $p(y)$ but without paired samples. From a probabilistic perspective, the essence of unpaired learning lies in inferring and modeling the joint distribution $p(x, y)$ under the constraint of these marginal distributions. Researchers often formulate unpaired learning as a generative problem, i.e., the generation of the corresponding y from a given x. Following this line, Generative Adversarial Networks (GANs) have been extensively applied due to their powerful cross-domain generation capabilities [25, 43, 3]. Nevertheless, these methods suffer from mode collapse and training instability, and they lack a rigorous probabilistic explanation. Variational Autoencoder (VAE)-based methods offer enhanced interpretability by introducing latent variables and optimizing the Evidence Lower Bound (ELBO) [48]; however, constrained by strong assumptions regarding the latent space distribution, their generation quality is often significantly compromised. Recent diffusion-based methods have begun to address conditional generation from unpaired data. OTCS constructs an optimal-transport coupling and trains a conditional score model using unpaired or partially paired data [12]. Diffusion Distribution Matching learns an unknown forward operator from unpaired clean and degraded images through conditional flow matching [30], whereas RealDGen synthesizes paired super-resolution data from unpaired HR and LR images using a content–degradation-decoupled diffusion model [34]. Schrödinger-bridge methods provide another coupling-based approach to unpaired translation, with recent bridge-flow formulations reducing repeated diffusion-model training [28, 14, 21, 9]. Relatedly, DRDD interprets Gaussian perturbation as a domain-harmonization mechanism for data-efficient image translation [27].

Despite this progress, a principled framework for deriving a likelihood-based conditional diffusion objective from fully unpaired marginals and quantifying the bias induced by the surrogate coupling remains under-explored. To address this gap, we propose LUD-DIF (**L**earning from **U**npaired **D**ata via **Dif**fusion Model) for inverse problems with unpaired data. LUD-DIF constructs proxy pairs through diffusion alignment and derives a trainable dual-path conditional diffusion objective from a marginal–conditional decomposition of the joint ELBO. This formulation relies solely on unpaired marginal samples and explicitly quantifies the bias induced by the weak-coupling approximation. The main contributions are summarized as follows:

- From the perspective of variational inference, we decouple the ELBO of the joint distribution under a reasonable data prior assumption. This provides a feasible training algorithm for solving inverse problems with unpaired data using diffusion models.
- We quantitatively analyze the theoretical error bound introduced by our assumptions. Based on this analysis, we propose a theorem-motivated heuristic for selecting the hyperparameters (S, V), establishing a theoretically grounded, practical strategy for model tuning.
- We validate the proposed method on natural image denoising and super-resolution datasets. Experimental results demonstrate that LUD-DIF achieves outstanding performance across these tasks.

2 Preliminaries

Diffusion models [16, 37] learn to approximate an unknown data distribution p_{U_0} by gradually adding noise to the data and then learning to reverse this process. We briefly introduce the main components of diffusion models.

Forward process Let $T \in \mathbb{N}_+$. The forward process adds noise via a fixed Markovian chain:

$$p_{U_{1:T}|U_0}(u_{1:T}|u_0) \coloneqq \prod_{t=1}^{T} p_{U_t|U_{t-1}}(u_t|u_{t-1}), p_{U_t|U_{t-1}}(\cdot|u_{t-1}) \coloneqq \mathcal{N}(\sqrt{\alpha_t}u_{t-1}, (1-\alpha_t)I_d), \tag{2.1}$$

where $\{\alpha_t\}_{t=1}^T \subset (0,1)$ is a prescribed variance schedule [16]. Let $\bar{\alpha}_0 \coloneqq 1$ and $\bar{\alpha}_t \coloneqq \prod_{i=0}^t \alpha_i$. Iterating this recursion yields the closed-form conditional marginal distribution for U_t given $U_0 = u_0$:

$$p_{U_t|U_0}(\cdot|u_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t}u_0, (1-\bar{\alpha}_t)I_d). \tag{2.2}$$

As $\bar{\alpha}_T \to 0$, the final distribution $p_{U_T|U_0}$ converges to a standard Gaussian $\mathcal{N}(0, I_d)$.

Reverse process To construct a generative model, the reverse process is parameterized as a Markovian chain with learned transition kernels starting from $p_{U_T}(u_T) \coloneqq \mathcal{N}(0, I_d)$:

$$p^\theta_{U_{t-1}|U_t}(\cdot|u_t) \coloneqq \mathcal{N}(\mu^\theta_t(u_t), \sigma_t^2 I_d), \tag{2.3}$$

where $\mu^\theta_t : \mathbb{R}^d \to \mathbb{R}^d$ is a neural network and σ_t^2 is a prescribed variance schedule.

ELBO formulation Diffusion models are trained by optimizing the Evidence Lower Bound (ELBO) of the log-likelihood. Through Markovian properties, the ELBO can be rewritten as a tractable decomposition [16, Appendix A]:

$$\begin{aligned}\mathrm{ELBO}(\theta; u_0) = &-D_{\mathrm{KL}}(p_{U_T|U_0}(\cdot|u_0)\|p_{U_T}) + \mathbb{E}\big[\log p^\theta_{U_0|U_1}(u_0|U_1)\big|U_0 = u_0\big] \\ &- \sum_{t=2}^T \mathbb{E}\big[D_{\mathrm{KL}}\big(p_{U_{t-1}|U_t,U_0}(\cdot|U_t,u_0)\|p^\theta_{U_{t-1}|U_t}(\cdot|U_t)\big)\big|U_0 = u_0\big].\end{aligned} \tag{2.4}$$

The first term is constant due to the construction of the forward process, and the second term is the reconstruction term. Moreover, we have $p_{U_{t-1}|U_t,U_0}(\cdot|u_t,u_0) = \mathcal{N}(\tilde{\mu}_t(u_t,u_0), \tilde{\beta}_t I_d)$, where

$$\tilde{\mu}_t(u_t,u_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}u_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}u_t, \quad \tilde{\beta}_t = \frac{(1-\bar{\alpha}_{t-1})\beta_t}{1-\bar{\alpha}_t}, \tag{2.5}$$

where $\beta_t \coloneqq 1-\alpha_t$. Substituting (2.5) into (2.4) yields the mean-squared error loss function:

$$\mathcal{L}(\theta) = \mathbb{E}\left[\sum_{t=1}^T \frac{1}{2\sigma_t^2}\|\mu^\theta_t(U_t) - \tilde{\mu}_t(U_t,U_0)\|_2^2\right],$$

where $\tilde{\mu}_1(u_1,u_0) \coloneqq u_0$.

3 The LUD-DIF method

3.1 Problem Formulation

Consider a general inverse problem modeled as

$$Y_0 = \mathcal{A}(X_0) + \Xi, \tag{3.1}$$

where $\mathcal{A}$ is the forward operator and Ξ denotes the noise. In the unpaired data setting, we are given samples from the marginal distributions

$$X_0^1, \ldots, X_0^N \overset{\text{i.i.d.}}{\sim} p_{X_0}, \qquad Y_0^1, \ldots, Y_0^N \overset{\text{i.i.d.}}{\sim} p_{Y_0},$$

and the goal is to recover the joint distribution p_{X_0,Y_0}. If the noise distribution were known, one could directly characterize the conditional distribution $p_{Y_0|X_0}$ via the forward model (3.1), and thereby recover the joint distribution. However, in practice, the noise distribution is typically unknown and may be highly complex, which poses the main challenge in estimating p_{X_0,Y_0}.

In this work, we model the noise conditionally as $\Xi \sim p_{\Xi|\mathcal{A}(X_0)}$. By treating $\mathcal{A}(X_0)$ as an effective signal variable and relabeling it as X_0, the forward model can be reduced to

$$Y_0 = X_0 + \Xi, \qquad \Xi \sim p_{\Xi|X_0}. \tag{3.2}$$

This reformulation reduces the problem to estimating the conditional noise distribution, which we refer to as the *noise modeling problem*.

Remark 3.1. If the forward operator is unknown, it can be replaced by an approximate unbiased operator $\widehat{\mathcal{A}}$, i.e., $\mathbb{E}[\mathcal{A}(X_0)] = \mathbb{E}[\widehat{\mathcal{A}}(X_0)]$, that incorporates prior information. In this case, the model can be written as

$$Y_0 = \widehat{\mathcal{A}}(X_0) + \widetilde{\Xi}, \qquad \widetilde{\Xi} = \Xi + \mathcal{A}(X_0) - \widehat{\mathcal{A}}(X_0).$$

The residual term is absorbed into the effective noise $\widetilde{\Xi}$, and the model is again reduced to the form in (3.2) with $\mathbb{E}[\widetilde{\Xi}] = \mathbb{E}[\Xi]$.

Accordingly, in the following, we focus on this *noise modeling problem*.

Data preprocessing. Let $\sigma_X^2 = \frac{1}{d}\mathbb{E}[\|X_0 - \mu_X\|_2^2]$ and $\sigma_Y^2 = \frac{1}{d}\mathbb{E}[\|Y_0 - \mu_Y\|_2^2]$, where $\mu_X = \mathbb{E}[X_0]$ and $\mu_Y = \mathbb{E}[Y_0]$. We normalize the data by

$$\tilde{X}_0 = \frac{X_0}{\sigma_X}, \quad \tilde{Y}_0 = \frac{Y_0}{\sigma_Y}. \tag{3.3}$$

This normalization aligns the scales of the two marginal distributions, making their amplitudes comparable when constructing a joint diffusion process for X_0 and Y_0. For notational simplicity, we assume that this preprocessing step has been applied and, in the remainder of the paper, use X_0 and Y_0 to denote the normalized variables $\tilde{X}_0$ and $\tilde{Y}_0$.

3.2 Learning joint distribution via diffusion models

Let $Z_0 \coloneqq (X_0, Y_0) \sim p_{X_0,Y_0}$ denote the true but unobserved joint data distribution. We construct a diffusion process on the joint variable Z_0 by specifying an appropriate reverse process.

Forward process Treating the joint variable $Z_t \coloneqq (X_t, Y_t)$ as the counterpart of U_t in (2.1), we define the joint forward diffusion process as

$$X_t \coloneqq \sqrt{\alpha_t} X_{t-1} + \sqrt{1-\alpha_t}\,\varepsilon_{t-1}^X, \quad Y_t \coloneqq \sqrt{\alpha_t} Y_{t-1} + \sqrt{1-\alpha_t}\,\varepsilon_{t-1}^Y, \tag{3.4}$$

where $\varepsilon_{t-1}^X, \varepsilon_{t-1}^Y \sim \mathcal{N}(0, I_d)$ are independent standard Gaussian noise variables that are independent of X_{t-1} and Y_{t-1}. We establish the following factorization result that decouples across X_0 and Y_0.

Proposition 3.2. *Let $Z_t = (X_t, Y_t)$ follow the forward process* (3.4)*. Then we have*

$$p_{Z_t|Z_0}(z_t|z_0) = p_{X_t|X_0}(x_t|x_0) p_{Y_t|Y_0}(y_t|y_0), \tag{3.5a}$$

$$p_{Z_{t-1}|Z_t,Z_0}(z_{t-1}|z_t, z_0) = p_{X_{t-1}|X_t,X_0}(x_{t-1}|x_t, x_0) p_{Y_{t-1}|Y_t,Y_0}(y_{t-1}|y_t, y_0), \tag{3.5b}$$

where $z_t \coloneqq (x_t, y_t)$.

Proof. See Appendix A.1. □

Reverse process Since paired samples $(X_0, Y_0) \sim p_{X_0,Y_0}$ are unavailable in the unpaired setting, it is difficult to directly adopt the reverse process parameterization (2.3) used in classical diffusion models. To construct a tractable variational reverse process, we first note that, for the X-first autoregressive ordering, the exact reverse transition kernel admits the chain-rule factorization

$$p_{Z_{t-1}|Z_t}(z_{t-1} \mid z_t) = p_{X_{t-1}|X_t,Y_t}(x_{t-1} \mid x_t, y_t)\, p_{Y_{t-1}|X_{t-1},X_t,Y_t}(y_{t-1} \mid x_{t-1}, x_t, y_t).$$

This identity only motivates the autoregressive ordering. In the variational model, we do not attempt to represent the two exact conditionals above. Instead, we restrict the reverse model to a tractable factored Gaussian family:

$$\begin{aligned} p^\theta_{Z_{t-1}|Z_t}(z_{t-1} \mid z_t) &\coloneqq p^\theta_{X_{t-1}|X_t}(x_{t-1} \mid x_t)\, p^\theta_{Y_{t-1}|X_{t-1},Y_t}(y_{t-1} \mid x_{t-1}, y_t) \\ &= \mathcal{N}\big(x_{t-1}; \mu^\theta_{X,t}(x_t), \sigma_t^2 I_d\big)\, \mathcal{N}\big(y_{t-1}; \mu^\theta_{Y|X,t}(y_t, x_{t-1}), \sigma_t^2 I_d\big). \end{aligned} \tag{3.6}$$

The omission of y_t in the first factor and x_t in the second factor should be understood as a restriction of the variational family, rather than as a conditional-independence assumption on the true reverse process.

Intuitively, the X-reverse kernel is used to model the clean structural marginal, for which x_t is the most direct noisy observation, while the Y-reverse kernel is conditioned on the less diffused structural state x_{t-1} together with the noisy observation y_t. This design keeps the conditional diffusion model trainable from the weak-coupling samples introduced below. Since both factors in (3.6) are normalized conditional Gaussian densities, their product defines a valid conditional density over (x_{t-1}, y_{t-1}) given (x_t, y_t).

Based on the above parameterization, we show that the proposed reverse process admits a corresponding decomposition of the ELBO.

ELBO formulation Analogously to (2.4), we identify Z_t with U_t and define the objective function as the negative-ELBO loss function:

$$\mathcal{L}_{\text{joint}}(\theta) \coloneqq -\mathbb{E}_{Z_0}\left[\text{ELBO}_Z(\theta; Z_0)\right], \tag{3.7}$$

where ELBO_Z is given by

$$\begin{aligned}\text{ELBO}_Z(\theta; z_0) \coloneqq &- D_{\text{KL}}(p_{Z_T|Z_0}(\cdot|z_0)\|p_{Z_T}) + \mathbb{E}\left[\log p^\theta_{Z_0|Z_1}(z_0|Z_1)|Z_0 = z_0\right] \\ &- \sum_{t=2}^{T} \mathbb{E}\left[D_{\text{KL}}\big(p_{Z_{t-1}|Z_t,Z_0}(\cdot|Z_t, z_0)\|p^\theta_{Z_{t-1}|Z_t}(\cdot|Z_t)\big)|Z_0 = z_0\right].\end{aligned} \tag{3.8}$$

Since paired samples $z_0 = (x_0, y_0) \sim p_{X_0,Y_0}$ are unavailable, we decompose the joint ELBO under the variational family in (3.6) into an X-marginal ELBO and a conditional $Y|X$ ELBO.

Proposition 3.3 (Decoupling of joint ELBO)**.** *Let the joint diffusion process $\{Z_t\}_{t=0}^T$ be defined by* (3.4) *and* (3.6)*, and let* ELBO_Z *be given by* (3.8)*. Then the following decomposition holds:*

$$\text{ELBO}_Z(\theta; z_0) = \text{ELBO}_X(\theta; x_0) + \text{ELBO}_{Y|X}(\theta; z_0) + C(z_0),$$

where $C(z_0)$ is a constant independent of θ. The marginal ELBO ELBO_X *is*

$$\begin{aligned}\text{ELBO}_X(\theta; x_0) = &\mathbb{E}\left[\log p^\theta_{X_0|X_1}(x_0|X_1)|X_0 = x_0\right] \\ &- \sum_{t=2}^{T} \mathbb{E}\left[D_{\text{KL}}(p_{X_{t-1}|X_t,X_0}(\cdot|X_t, x_0)\|p^\theta_{X_{t-1}|X_t}(\cdot|X_t))\Big|X_0 = x_0\right].\end{aligned} \tag{3.9}$$

The conditional ELBO $\text{ELBO}_{Y|X}$ *is*

$$\begin{aligned}\text{ELBO}_{Y|X}(\theta; z_0) = &\mathbb{E}\big[\log p^\theta_{Y_0|X_0,Y_1}(y_0 \mid x_0, Y_1) \mid Z_0 = z_0\big] \\ &- \sum_{t=2}^{T} \mathbb{E}\Big[D_{\text{KL}}\Big(p_{Y_{t-1}|Y_t,Y_0}(\cdot|Y_t, y_0) \;\Big\|\; p^\theta_{Y_{t-1}|X_{t-1},Y_t}(\cdot \mid X_{t-1}, Y_t)\Big) \;\Big|\; Z_0 = z_0\Big].\end{aligned} \tag{3.10}$$

Proof. See Appendix A.2. □

Using the parameterizations in (3.6), we obtain, up to a constant,

$$-\mathbb{E}[\text{ELBO}_X(\theta; x_0)] = \mathbb{E}\sum_{t=1}^{T} \frac{1}{2\sigma_t^2}\|\mu^\theta_{X,t}(X_t) - \tilde{\mu}_t(X_t, X_0)\|_2^2 =: L_{\text{marg}}(\theta), \tag{3.11}$$

$$-\mathbb{E}[\text{ELBO}_{Y|X}(\theta; z_0)] = \mathbb{E}\sum_{t=1}^{T} \frac{1}{2\sigma_t^2}\|\mu^\theta_{Y|X,t}(Y_t, X_{t-1}) - \tilde{\mu}_t(Y_t, Y_0)\|_2^2 =: L_{\text{cond}}(\theta). \tag{3.12}$$

Collecting $-\mathbb{E}[C(Z_0)]$ and all other terms independent of θ into a constant C, the overall loss function becomes

$$\mathcal{L}_{\text{joint}}(\theta) \coloneqq -\mathbb{E}\left[\text{ELBO}_Z(\theta; Z_0)\right] = L_{\text{marg}}(\theta) + L_{\text{cond}}(\theta) + C. \tag{3.13}$$

This formulation serves as the ideal loss for learning the joint distribution. Notably, $L_{\text{marg}}(\theta)$ can be estimated using samples from p_{X_0}, whereas $L_{\text{cond}}(\theta)$ requires samples from the joint distribution of (X_{t-1}, Y_t, Y_0), which depends on

$$p_{X_{t-1},Y_t,Y_0}(x_{t-1}, y_t, y_0) = \int p_{X_{t-1}|X_0}(x_{t-1}|x_0)\, p_{X_0|Y_0}(x_0|y_0)\, p_{Y_t,Y_0}(y_t, y_0)\, \mathrm{d}x_0.$$

However, the available marginal distributions p_{X_0} and p_{Y_0} alone do not uniquely determine the joint distribution. As a result, the conditional distribution $p_{X_0|Y_0}$ remains inaccessible in the unpaired setting. In many inverse problems, high-dimensional noisy signals are more easily obscured by additional Gaussian perturbations than are the underlying dominant structures. Motivated by this observation, we introduce the following weak-coupling assumption.

Assumption 3.4 (Weak coupling)**.** There exist time steps $0 \le V, S \le T$ such that

$$p_{X_0|Y_0}(\cdot|y_0) = p^{\text{weak}}_{X_0|Y_0}(\cdot|y_0), \quad y_0 \in \mathbb{R}^d,$$

where $p^{\text{weak}}_{X_0|Y_0}$ and $p^{\text{weak}}_{X_0|Y_S}$ are defined as

$$p^{\text{weak}}_{X_0|Y_0}(\cdot|y_0) \coloneqq \mathbb{E}_{\xi\sim\mathcal{N}(0,I_d)}\left[p^{\text{weak}}_{X_0|Y_S}(\cdot|\sqrt{\bar\alpha_S}y_0 + \sqrt{1-\bar\alpha_S}\xi)\right], \tag{3.14}$$

$$p^{\text{weak}}_{X_0|Y_S}(\cdot|w) \coloneqq p_{X_0|X_V}(\cdot|w), \quad \forall w \in \mathbb{R}^d. \tag{3.15}$$

Remark 3.5. For Gaussian white noise with variance σ^2, this coupling assumption is exact when we set $S = 0$ and choose V such that $\bar\alpha_V = \frac{\sigma_X^2}{\sigma_X^2+\sigma^2}$, thereby matching the signal-to-noise ratio of the measurement.

Remark 3.6. For the general case, the assumption relies on two approximations: (i) Alignment approximation: $p_{X_0|Y_S}(\cdot \mid w) \approx p_{X_0|X_V}(\cdot \mid w)$, which assumes that the noisy representations of the two modalities become structurally indistinguishable at suitable noise levels; (ii) Perturbation approximation: $p_{X_0|Y_0}(\cdot \mid y_0) \approx \mathbb{E}_{\xi\sim\mathcal{N}(0,I_d)}\left[p_{X_0|Y_S}\big(\cdot \mid \sqrt{\bar\alpha_S}\, y_0 + \sqrt{1-\bar\alpha_S}\, \xi\big)\right]$, which assumes that the perturbation Y_S preserves sufficient information about X_0. These two approximations induce a trade-off in the choice of the time steps S and V. When S is small, the perturbation approximation is nearly exact, whereas the alignment approximation is difficult to satisfy. As S and V increase, the alignment approximation improves, but the perturbation approximation deteriorates due to information loss. Therefore, the practical effectiveness of the weak-coupling assumption depends on selecting S and V to balance these competing effects.

The weak-coupling distribution can be sampled using only marginal data:

1. Draw $(Y_0, \xi) \sim p_{Y_0} \otimes \mathcal{N}(0, I_d)$;
2. Define $\bar Y_S = \sqrt{\bar\alpha_S}\, Y_0 + \sqrt{1-\bar\alpha_S}\, \xi$;
3. Draw $X_0^{\text{weak}} \sim p_{X_0|X_V}(\cdot \mid \bar Y_S)$ via reverse process.

By replacing the inaccessible conditional distribution $p_{X_0|Y_0}$ with the proxy conditional distribution $p^{\text{weak}}_{X_0|Y_0}$ in (3.12), we define the weak conditional loss as

$$L^{\text{weak}}_{\text{cond}}(\theta) \coloneqq \mathbb{E}\left[\sum_{t=1}^{T} \frac{1}{2\sigma_t^2}\|\mu^\theta_{Y|X,t}(Y_t, X^{\text{weak}}_{t-1}) - \tilde\mu_t(Y_t, Y_0)\|_2^2\right], \tag{3.16}$$

where

$$X^{\text{weak}}_{t-1} \coloneqq \sqrt{\bar\alpha_{t-1}}X_0^{\text{weak}} + \sqrt{1-\bar\alpha_{t-1}}\varepsilon^X, \quad Y_t \coloneqq \sqrt{\bar\alpha_t}Y_0 + \sqrt{1-\bar\alpha_t}\varepsilon^Y,$$

with $\varepsilon^X, \varepsilon^Y \sim \mathcal{N}(0, I_d)$ independent of each other and independent of X_0^{weak} and Y_0. Therefore, X_0^{weak} is conditionally independent of Y_t given Y_0. This modification avoids the need for paired data and makes the conditional loss computable. The next theorem shows the equivalence between $L^{\text{weak}}_{\text{cond}}(\theta)$ in (3.16) and $L_{\text{cond}}(\theta)$ in (3.12) under Assumption 3.4. We define

$$\mathcal{L}_{\text{weak}}(\theta) \coloneqq L_{\text{marg}}(\theta) + L^{\text{weak}}_{\text{cond}}(\theta) + C, \tag{3.17}$$

where $L_{\text{marg}}(\theta)$ is defined in (3.11) and $L^{\text{weak}}_{\text{cond}}(\theta)$ is defined in (3.16).

Theorem 3.7. *Suppose that Assumption 3.4 holds. Let* $\mathcal{L}_{\text{joint}}(\theta)$ *and* $\mathcal{L}_{\text{weak}}(\theta)$ *be defined as* (3.13) *and* (3.17)*, respectively. Then we have*

$$L_{\text{cond}}(\theta) = L_{\text{cond}}^{\text{weak}}(\theta).$$

Thus, $\mathcal{L}_{\text{joint}}(\theta) = \mathcal{L}_{\text{weak}}(\theta)$.

Proof. See Appendix A.3. □

Theorem 3.7 provides a consistency result: when the weak-coupling construction matches the true posterior coupling, the computable weak objective coincides with the ideal paired-data objective. When the assumption is relaxed, Section 4 quantifies the resulting bias. Algorithm 1 summarizes the training procedure. Since the objective is decoupled into marginal and conditional terms, the two diffusion models can be trained separately, improving the training efficiency of the conditional model.

Algorithm 1 LUD-DIF Algorithm: Training Process

Initialize parameters S, V, α_t of the model.
for each training iteration **do**
  Sample $X_0 \sim p_{X_0}$, $Y_0 \sim p_{Y_0}$ and $t \sim \text{Uniform}\{1, \dots, T\}$.
  Generate $X_t \sim p_{X_t|X_0}(\cdot|X_0)$ and $Y_t \sim p_{Y_t|Y_0}(\cdot|Y_0)$ via the forward process (3.4).
  Generate proxy sample $X_0^{\text{weak}} \sim p_{X_0|Y_0}^{\text{weak}}(\cdot|Y_0)$ as defined in (3.14) and (3.15).
  Construct $X_{t-1} = \sqrt{\bar{\alpha}_{t-1}} sg(X_0^{\text{weak}}) + \sqrt{1-\bar{\alpha}_{t-1}}\xi$ where $\xi \sim \mathcal{N}(0, I_d)$.
  Compute $\mathcal{L}_X = \frac{1}{2\sigma_t^2}\|\mu_{X,t}^\theta(X_t) - \tilde{\mu}_t(X_t, X_0)\|^2$.
  Compute $\mathcal{L}_{Y|X} = \frac{1}{2\sigma_t^2}\|\mu_{Y|X,t}^\theta(Y_t, X_{t-1}) - \tilde{\mu}_t(Y_t, Y_0)\|^2$.
  Update θ by minimizing $\mathcal{L}_{\text{weak}} = \mathcal{L}_X + \mathcal{L}_{Y|X}$.
end for

4 The Bias Analysis without Assumption 3.4

Theorem 3.7 establishes the equivalence between $\mathcal{L}_{\text{joint}}$ and $\mathcal{L}_{\text{weak}}$ under Assumption 3.4. Here, we quantify their discrepancy when this assumption is relaxed:

$$\Delta \coloneqq |\mathcal{L}_{\text{joint}} - \mathcal{L}_{\text{weak}}|. \tag{4.1}$$

Throughout this section, $p_{X_0|X_V}$ denotes the exact conditional distribution, so learning and sampling errors are excluded. Following the normalization in (3.3), we consider

$$Y_0 = \frac{\sigma_X}{\sigma_Y} X_0 + \frac{1}{\sigma_Y}\Xi, \tag{4.2}$$

where Ξ is assumed to be independent of X_0. A concise extension to signal-dependent conditional noise is given in Appendix A.7. The resulting bounds also guide the selection of the weak-coupling time steps S and V. We first state the required technical assumptions.

Assumption 4.1 (Polynomial growth of networks)**.** The expectation neural network $\mu_{Y|X,t}^\theta : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ defined in (3.6) is bounded by a polynomial uniformly over the parameter set Θ, i.e., there exist $C > 0$ and $m \geq 1$ such that for any $\theta \in \Theta$ and $y_t, x \in \mathbb{R}^d$,

$$\|\mu_{Y|X,t}^\theta(y_t, x)\|_2^2 \leq C(1 + \|y_t\|_2^m + \|x\|_2^m).$$

The polynomial growth of the neural network in Assumption 4.1 can be ensured in practice by truncation or weight clipping. Note that the expectation $\tilde{\mu}_t(y_t, y_0)$ defined in (2.5) is a linear combination of y_t and y_0; thus, under Assumption 4.1, there exists a constant $K > 0$ such that

$$\|\mu_{Y|X,t}^\theta(y_t, x) - \tilde{\mu}_t(y_t, y_0)\|_2^2 \leq K(1 + \|y_t\|_2^{\max\{m,2\}} + \|x\|_2^m + \|y_0\|_2^2).$$

Here K is a constant depending only on C and the variance schedule of the diffusion model.

Before proceeding with the data distributions, we note the necessary variable assumptions supporting the scaling properties.

Assumption 4.2 (Data and noise distribution)**.** The normalized clean data X_0 admits a probability density p_{X_0} with respect to the Lebesgue measure on $\mathbb{R}^d$ and is supported on a compact set $\mathcal{X} \subset \mathbb{R}^d$. The noise Ξ is independent of X_0 and is centered, i.e.,

$$\mathbb{E}[\Xi] = \mathbf{0}.$$

Moreover, Ξ admits a probability density p_Ξ satisfying

$$\|p_\Xi\|_{L^\infty(\mathbb{R}^d)} \le M_\Xi,$$

and has a bounded $2\max\{m, 2\}$-th moment:

$$\mathbb{E}\left[\|\Xi\|_2^{2\max\{m,2\}}\right] \le M_{\Xi,m},$$

for some finite constants $M_\Xi, M_{\Xi,m} > 0$.

Since Y_0 is a linear combination of the bounded X_0 and the noise Ξ, it naturally follows that the noisy observation Y_0 has finite $2\max(m, 2)$-th order moments.

The Total Variation (TV) distance between two probability density functions p_1 and p_2 is defined as $\|p_1 - p_2\|_{\mathrm{TV}} \coloneqq \frac{1}{2}\int |p_1(x) - p_2(x)|\,\mathrm{d}x$. The following proposition shows that the error of the weakly coupled loss function can be controlled by the expected TV distance between the true posterior $p_{X_0|Y_0}(\cdot|Y_0)$ and the weakly coupled posterior $p^{\mathrm{weak}}_{X_0|Y_0}(\cdot|Y_0)$.

Proposition 4.3. *Suppose Assumptions 4.1 and 4.2 are fulfilled. Then there exists a sequence of positive finite constants $\{M_t\}_{t=1}^T$ such that the error function is bounded by*

$$\Delta \le \sum_{t=1}^{T} \frac{M_t}{\sigma_t^2} \mathbb{E}_{Y_0}^{\frac{1}{2}}\left[\|p_{X_0|Y_0}(\cdot|Y_0) - p^{\mathrm{weak}}_{X_0|Y_0}(\cdot|Y_0)\|_{\mathrm{TV}}\right],$$

where M_t is a constant depending on C, $R_{\mathcal{X}} \coloneqq \sup_{x\in\mathcal{X}} \|x\|_2$, dimension d, moment bound $M_{\Xi,m}$, scales σ_X, σ_Y, and the variance schedule.

Proof. See Appendix A.4. □

From Proposition 4.3, in order to estimate the error Δ of the weakly coupled loss function, it is sufficient to bound the expected TV distance between the weakly coupled posterior and the true posterior:

$$\Delta_p \coloneqq \mathbb{E}_{Y_0}\left[\|p^{\mathrm{weak}}_{X_0|Y_0}(\cdot|Y_0) - p_{X_0|Y_0}(\cdot|Y_0)\|_{\mathrm{TV}}\right]. \tag{4.3}$$

Recall that $p^{\mathrm{weak}}_{X_0|Y_0}(\cdot|Y_0) = \mathbb{E}_{Y_S|Y_0}[p_{X_0|X_V}(\cdot|Y_S)]$. By Jensen's inequality, the expected TV distance Δ_p between the weakly coupled posterior and the true posterior can be decomposed as

$$\begin{aligned}\Delta_p &\le \mathbb{E}_{Y_0}\mathbb{E}_{Y_S|Y_0}\left[\|p_{X_0|X_V}(\cdot|Y_S) - p_{X_0|Y_0}(\cdot|Y_0)\|_{\mathrm{TV}}\right] \\ &\le \underbrace{\mathbb{E}_{Y_S}\left[\|p_{X_0|X_V}(\cdot|Y_S) - p_{X_0|Y_S}(\cdot|Y_S)\|_{\mathrm{TV}}\right]}_{\text{alignment error}} + \underbrace{\mathbb{E}_{Y_0,Y_S}\left[\|p_{X_0|Y_S}(\cdot|Y_S) - p_{X_0|Y_0}(\cdot|Y_0)\|_{\mathrm{TV}}\right]}_{\text{perturbation error}}.\end{aligned} \tag{4.4}$$

The first summand in (4.4) represents the **alignment error**, arising from the structural discrepancy between the noisy signal X_V and the noisy measurement Y_S. This corresponds to the error introduced by the alignment approximation in (3.15). The second summand represents the **perturbation error**, which stems from the information loss in Y_S due to the perturbation to Y_0.

To establish rigorous bounds for these errors, we introduce the stability assumption on the posterior distribution with respect to the change of observation.

Assumption 4.4 (Posterior stability)**.** There exists a constant $L_{\mathrm{post}} > 0$ such that

$$\|p_{X_0|Y_0}(\cdot|y_0^1) - p_{X_0|Y_0}(\cdot|y_0^2)\|_{\mathrm{TV}} \le L_{\mathrm{post}}\|y_0^1 - y_0^2\|_2, \quad \forall\, y_0^1, y_0^2 \in \mathbb{R}^d.$$

The stability of the posterior distribution with respect to perturbations of the observations has been extensively studied in the context of Bayesian inverse problems [39]. We adopt it here as a mild regularity assumption. Appendix A.7 records a sufficient condition: under Assumption 4.2, a positive noise density whose log-density has a bounded Hessian induces a TV-Lipschitz posterior map. This condition covers Gaussian noise and a broad class of non-Gaussian cases.

The following theorem provides explicit upper bounds for the perturbation error and the alignment error in terms of the diffusion time steps S, V and the noise Ξ under the stated assumptions.

Theorem 4.5 (Error bound for the weakly coupled loss function)**.** *Suppose Assumptions 4.1, 4.2, and 4.4 are fulfilled. Let $\sigma_\Xi^2 \coloneqq \frac{1}{d}\mathbb{E}[\|\Xi\|_2^2]$ be the scalar variance of the noise Ξ. Then*

$$\Delta_p \le \underbrace{L_{\text{post}}\sqrt{2d\frac{1-\bar\alpha_S}{\bar\alpha_S}}}_{\textit{perturbation error}} + \underbrace{\zeta_S\frac{|\sigma_Y\sqrt{\bar\alpha_V}-\sigma_X\sqrt{\bar\alpha_S}|}{\sqrt{1-\bar\alpha_V}} + \eta_S\sqrt{D_{\text{KL}}(p_\Xi \,\|\, \mathcal{N}(0,\sigma_\Xi^2 I_d))}}_{\textit{alignment error}},$$

where the constants ζ_S and η_S are defined as

$$\zeta_S \coloneqq \frac{R_{\mathcal{X}}}{\sigma_Y} + \sqrt{\frac{8d}{\sigma_Y^2-\bar\alpha_S\sigma_X^2}}, \quad \eta_S \coloneqq \sqrt{\frac{2\bar\alpha_S\sigma_\Xi^2}{\sigma_Y^2-\bar\alpha_S\sigma_X^2}}.$$

Moreover, the error function Δ satisfies

$$\Delta \le \sum_{t=1}^{T}\frac{M_t}{\sigma_t^2}\left(L_{\text{post}}\sqrt{2d\frac{1-\bar\alpha_S}{\bar\alpha_S}} + \zeta_S\frac{|\sigma_Y\sqrt{\bar\alpha_V}-\sigma_X\sqrt{\bar\alpha_S}|}{\sqrt{1-\bar\alpha_V}} + \eta_S\sqrt{D_{\text{KL}}(p_\Xi \,\|\, \mathcal{N}(0,\sigma_\Xi^2 I_d))}\right)^{\frac{1}{2}}.$$

Proof. See Appendix A.5. □

We define a conditional distribution in (3.14), which induces a conditional distribution of Y_0 given X_0^{weak}:

$$p^{\text{weak}}_{Y_0|X_0}(\cdot|x_0) \propto p^{\text{weak}}_{X_0|Y_0}(x_0|\cdot)p_{Y_0}, \quad x_0 \in \mathcal{X}. \tag{4.5}$$

The following corollary provides an expected TV-bound.

Corollary 4.6 (Conditional generative error)**.** *Under the same assumptions as in Theorem 4.5, let $p^{\text{weak}}_{Y_0|X_0}$ be the weakly coupled conditional distribution defined in* (4.5)*. Then we have*

$$\begin{aligned}&\mathbb{E}_{X_0}\left[\|p_{Y_0|X_0}(\cdot|X_0) - p^{\text{weak}}_{Y_0|X_0}(\cdot|X_0)\|_{\text{TV}}\right]\\ &\le 2L_{\text{post}}\sqrt{2d\frac{1-\bar\alpha_S}{\bar\alpha_S}} + 2\zeta_S\frac{|\sigma_Y\sqrt{\bar\alpha_V}-\sigma_X\sqrt{\bar\alpha_S}|}{\sqrt{1-\bar\alpha_V}} + 2\eta_S\sqrt{D_{\text{KL}}(p_\Xi \,\|\, \mathcal{N}(0,\sigma_\Xi^2 I_d))}.\end{aligned}$$

Proof. See Appendix A.6. □

Remark 4.7 (Selection of time steps S and V). The bounds motivate the variance-alignment condition $\bar\alpha_V\sigma_Y^2 = \bar\alpha_S\sigma_X^2$, which eliminates the mean- and variance-mismatch terms. For additive Gaussian white noise, the non-Gaussianity term vanishes, while setting $S = 0$ eliminates the perturbation term, recovering the exact formulation in Remark 3.5.

For practical noise, bounding the exact discrepancy is challenging. We therefore propose a simplified heuristic objective for selecting a suitable S. Specifically, we use $\sigma_X \approx \sigma_Y$ to simplify $\eta_S \approx \frac{\sqrt{2\bar\alpha_S}\sigma_\Xi}{\sigma_Y\sqrt{1-\bar\alpha_S}}$ and approximate the KL divergence as $D_{\text{KL}}(p_\Xi \,\|\, \mathcal{N}(0,\sigma_\Xi^2 I_d)) \approx \big|\kappa(Y)-\kappa(X)\big|^2$, which gives

$$\bar\alpha_S^* = \operatorname*{arg\,min}_{\bar\alpha_S\in(0,1]}\left(\lambda\sqrt{\frac{1-\bar\alpha_S}{\bar\alpha_S}} + \sqrt{\frac{\bar\alpha_S}{1-\bar\alpha_S}}\big|\kappa(Y)-\kappa(X)\big|\right) = \frac{\lambda}{\lambda+\big|\kappa(Y)-\kappa(X)\big|}, \tag{4.6}$$

where the hyperparameter $\lambda \coloneqq \frac{L_{\text{post}}\sigma_Y\sqrt{d}}{\sigma_\Xi}$ theoretically aggregates the constants from the error bound to balance the two components, and $\kappa(\cdot)$ denotes the standardized dimension-averaged excess kurtosis, defined by

$$\kappa(X) \coloneqq \frac{1}{d\sigma_X^4}\mathbb{E}\big[\|X-\mathbb{E}[X]\|_2^4\big] - (d+2).$$

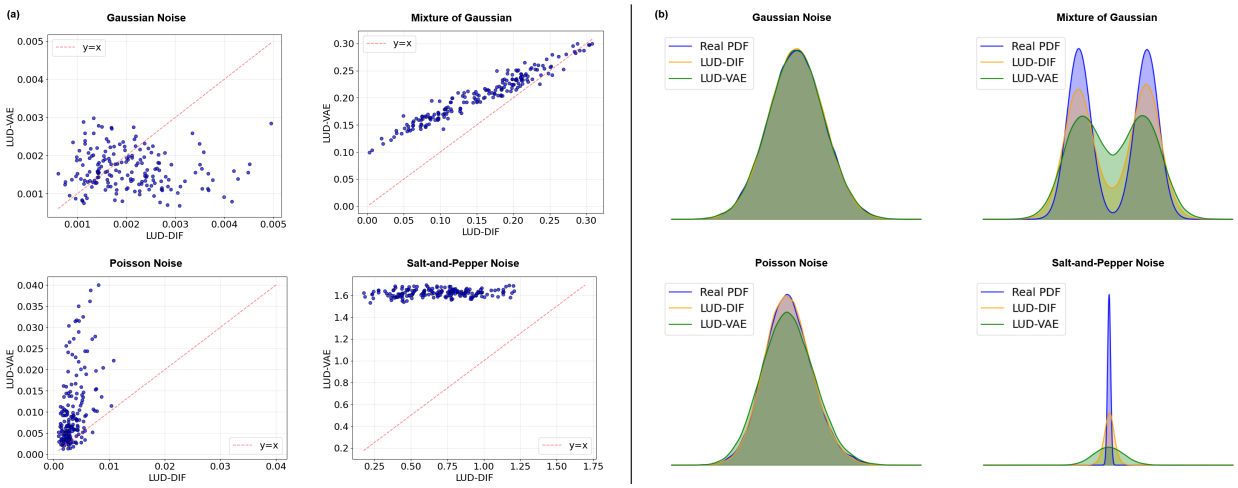

Figure 1: Comparison of the modeling capabilities of LUD-VAE and LUD-DIF for different simulated noises on the BSDS300 dataset. (a) Scatter plot of the KL divergence between the generated noise and the ground-truth noise for each image in the validation set. Points below the $y = x$ line indicate that LUD-DIF yields worse results on the corresponding sample. (b) Comparison of the generated noise distribution curves between the two methods. The selected sample points correspond to cases where both methods perform within the 25%–50% best range across all noise types.

Subsequently, the coupling step V is determined by the variance alignment constraint $\bar{\alpha}_V = (\sigma_X^2/\sigma_Y^2)\bar{\alpha}_S^*$. In practice, λ can be empirically calibrated using synthetic noise data.

5 Experimental Results

In this section, we evaluate the performance of LUD-DIF on different tasks. First, we assess its ability to model various simulated noise distributions on natural images. Next, we evaluate the model's capability in capturing complex real-world noise using a smartphone image denoising dataset (SIDD). Finally, we demonstrate the applicability of our method to image super-resolution.

Given a clean sample x_0, we first diffuse it to the aligned state x_V, and then use the learned conditional reverse process to generate the corresponding noisy or degraded observation y_0. The generated pair (x_0, y_0) is then used either for evaluating the learned degradation model or for training a downstream restoration network.

For implementation, we report the diffusion levels by their equivalent additive noise scales τ_S and τ_V, rather than directly listing $\bar{\alpha}_S$ and $\bar{\alpha}_V$. On the original data scale, these quantities are related by

$$\tau_S = \sigma_Y \sqrt{\frac{1-\bar{\alpha}_S}{\bar{\alpha}_S}}, \qquad \tau_V = \sigma_X \sqrt{\frac{1-\bar{\alpha}_V}{\bar{\alpha}_V}}.$$

Under the variance-alignment condition $\bar{\alpha}_V \sigma_Y^2 = \bar{\alpha}_S \sigma_X^2$, this is equivalent to $\tau_V^2 = \tau_S^2 + \sigma^2$, where σ denotes the standard deviation of the degradation noise.

5.1 Simulated Image Noise

Dataset: We evaluate the capability of LUD-DIF in simulating image noise distributions on the BSDS300 dataset [29]. The BSDS300 dataset contains 300 natural images, of which 200 images are used as the training set and the remaining 100 images as the test set. For images in the training set, we conduct training on 64×64 patches, while for images in the test set, we perform evaluation on 256×256 patches. The training set is evenly split into two halves. One half is used to create four noisy datasets by adding Gaussian noise, Poisson noise, salt-and-pepper noise, and a mixture of Gaussian noises to achieve a PSNR of 20 dB. The other half serves as the unpaired clean dataset.

Implementation Details: Our model consists of a diffusion model for clean images and a conditional diffusion model for noisy images, both employing a UNet architecture. It is trained for 100k iterations using the Adam optimizer with an initial learning rate of 1×10^{-4}, which is gradually decayed to 1×10^{-6} based on the loss reduction. The batch size is set to 8 during training. For sampling, we employ the DDIM sampler [36] with 50 steps and set the stochastic term η to 0.0. For Gaussian noise, we set $\tau_S = 0$. For Poisson noise and mixed Gaussian noise, we set $\tau_S = 0.5\sigma$, where σ is the standard deviation of the added noise, and choose $\tau_V = \sqrt{\tau_S^2 + \sigma^2}$ following the corresponding variance-matching rule in our implementation. For salt-and-pepper noise, we use the practical smoothing choice $\tau_V = \tau_S = \sigma$. Furthermore, when constructing the weak reconstruction $p_{X_0|X_V}(\cdot|Y_S)$, we first smooth the input y by replacing all pixels with values of 0 or 255 with the mean value of their neighboring pixels.

Results: We visualize the generation results of LUD-DIF under different noise distributions, as shown in Fig. 1 and Table 1. It can be observed that for Gaussian noise, both methods demonstrate strong modeling capabilities. While LUD-VAE yields more stable generation, LUD-DIF exhibits larger errors on certain instances. For unimodal distributions such as Poisson noise, our method effectively captures the noise characteristics, generating noise images that closely approximate the real noise distribution. In contrast, LUD-VAE fails to capture the noise properties in a significant number of samples.

For multimodal structures like Gaussian mixture distributions, our method successfully captures the multimodal characteristics but fails to precisely reconstruct the true noise distribution. This is attributed to the stronger non-Gaussianity and higher-order moment discrepancy of multimodal distributions, which is qualitatively consistent with the non-Gaussianity term in Theorem 4.5. For discrete distributions such as salt-and-pepper noise, the smoothing technique described above improves the generation quality of the weak reconstruction $p_{X_0|X_V}(\cdot|Y_S)$, thereby achieving better visual performance. However, since salt-and-pepper noise is zero-valued at most pixel locations and diffusion models struggle to guarantee identity mapping at these points, the KL divergence remains relatively high. Conversely, LUD-VAE is unable to effectively model such extreme noise patterns.

Noise Type	Gaussian	Mixture Gaussian	Poisson	Salt-and-Pepper
LUD-VAE	**0.0017**	0.2103	0.0147	1.6275
LUD-DIF	0.0023	**0.1644**	**0.0043**	**0.7129**

Table 1: Average KL divergence of LUD-VAE and LUD-DIF for different simulated noise types.

5.2 Real-world Noise Modeling

Dataset: We employ the Smartphone Image Denoising Dataset (SIDD) [1] to evaluate our method's ability to model complex real-world noise. The SIDD dataset contains noisy images and their corresponding clean images captured by different smartphones under various lighting conditions. We use the SIDD-Medium subset, which consists of 320 pairs of noisy and clean images. For each pair, we randomly crop 300 patches of size 256×256, resulting in a total of 96,000 paired patches. When training the model, we use all 96,000 images but do not utilize their pairing information, meaning that each iteration uses either clean or noisy images. We refer to this training setup, in which paired data exist in the training set but the pairing relationships are unknown, as Weak-Unpaired (**WUP**). To simulate a strict unpaired setting, we split these 96,000 images into two halves: 48,000 images serve as the clean image set, and the remaining 48,000 images serve as the noisy image set. We refer to this setting, in which no paired data are available during training, as Strong-Unpaired (**SUP**). We use the SIDD validation set, which contains 1,280 images of size 256×256, for evaluation.

Evaluation Metrics: To assess the quality of the noise images generated by our method, we use the clean images from the SIDD validation set to generate their corresponding noisy images. We then evaluate the discrepancy between these generated noisy images and the real noisy images using the Fréchet Inception Distance (FID) [15] and the Average KL Divergence (AKLD) between the synthesized and real noisy images. Motivated by the kurtosis-based heuristic in Remark 4.7, we also report the Average Fourth-order Moment Difference (AFMD) to measure the difference in fourth-order statistics between the generated noise

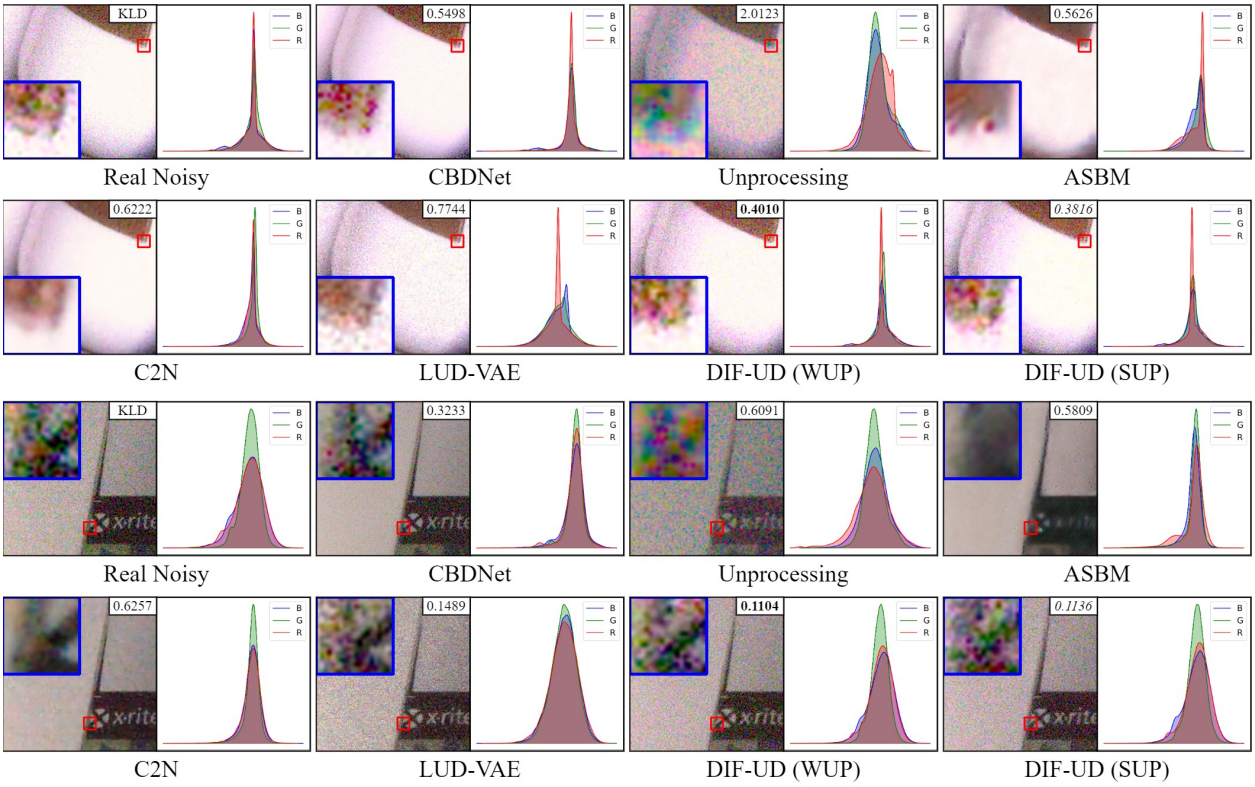

Figure 2: Comparison of different methods for unpaired noise modeling on the SIDD dataset.

distribution and the real noise distribution:

$$\text{AFMD} = \frac{1}{N}\sum_{i=1}^{N}\left|\kappa(n_{\text{real}}^{(i)}) - \kappa(n_{\text{fake}}^{(i)})\right|$$

where $n_{\text{real}}^{(i)}$ and $n_{\text{fake}}^{(i)}$ represent the real noise and the generated noise of the i-th sample, respectively, and κ denotes the standardized excess kurtosis of the noise. Additionally, we generate noisy images on a portion of the training set to obtain several pairs of noisy and clean images. These paired data are then used to train a DnCNN denoising network[1]. The denoising performance is evaluated on the test set using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) [42].

Implementation Details: The network architecture is the same as that described in Section 5.1. The model is trained for 300k iterations using the Adam optimizer, with the learning rate set to 1×10^{-4} and gradually decreased to 1×10^{-6} based on the decline of the loss. During training, the batch size is set to 64, and the model is trained on randomly cropped image patches of size 64×64. For sampling, the DDIM sampler is employed. To achieve better generation quality when generating noise for the validation set, we use 200 sampling steps and set the stochastic term η to 0.5. For generating the training set for DnCNN, which prioritizes faster generation speed while ensuring noise randomness, we use 50 sampling steps and set η to 1.0. Due to the significant variation in noise distributions within the SIDD dataset, we select V and S separately for each image. Specifically, we first train a noise variance estimator on clean data using Gaussian-simulated noise. For a noisy image, this estimator is used to predict its noise variance $\hat{\sigma}^2$. The values of V and S are then selected by an empirically calibrated rule motivated by the heuristic objective in Remark 4.7, with V chosen to match the corresponding variance level after S is fixed. Empirically, setting $\tau_S = 0.5\sigma$ yields favorable results. To enable the generation of noisy images with specified noise intensity, an additional channel is introduced in the input of the conditional diffusion model to feed in the noise level

[1]More advanced denoisers often possess stronger generalization capabilities, which is not conducive to reflecting quality differences in the training set.

Table 2: Quantitative comparison of noise generation quality on the SIDD validation set and denoising performance on the SIDD benchmark. The best results are highlighted in **bold**, and the second-best results are underlined. Note that "Paired" refers to the UNet backbone performance using paired data.

Method	Noise Generation Quality			Downstream Performance	
	FID ↓	AKLD ↓	AFMD ↓	PSNR ↑	SSIM ↑
CBDNet[13]	141.34	1.795	2.186	34.48	0.848
Unprocessing[7]	95.10	0.964	1.338	21.74	0.440
C2N[17]	33.97	0.169	1.141	34.12	0.818
DeFlow[46]	39.45	0.205	-	33.82	0.846
LUD-VAE[48]	35.31	0.108	0.780	34.91	0.892
ASBM[14]	68.93	0.552	0.848	34.32	0.866
LUD-DIF (Ours)	**16.99**	**0.046**	**0.420**	**35.47**	**0.896**
LUD-DIF(WUP)	16.30	0.026	0.340	35.60	0.898
Paired	13.86	0.014	0.206	37.89	0.906

Table 3: Quantitative comparison on the SIDD Validation and SIDD Benchmark datasets.

Method	SIDD Validation		SIDD Benchmark	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
N2V [22]	29.35	0.651	27.68	0.668
N2S [5]	30.72	0.787	29.56	0.808
CVF-SID [31]	34.17	0.872	34.71	0.917
AP-BSN + R^3 [23]	35.91	0.882	35.97	0.925
SCPGabNet [26]	36.53	0.886	36.53	0.925
SDAP(S)(E) [33]	**37.55**	0.894	37.53	**0.936**
LUD-DIF (Ours)	37.37	**0.906**	**37.59**	0.898

information. When generating noise simulations on the validation set, we directly use the given corresponding variance to generate the noisy images. Conversely, when generating noisy images on the training set for DnCNN training, we randomly sample the noise standard deviation from the interval $[5, 40]$ to produce the noisy images.

Compared Methods: We compare the LUD-DIF method with several unpaired noise modeling approaches, including C2N [17], DeFlow [46], and LUD-VAE [48]. These models are all trained using the hyperparameters specified in [48] and employ the SUP training data. We also compare against two model-based noise modeling methods: CBDNet [13] and Unprocessing [7]. For EOT-based methods, we utilize ASBM [14], which achieves state-of-the-art generation speed and quality. The diffusion term parameter for ASBM is set to 0.3 to match the overall noise intensity of the dataset. Furthermore, we evaluate the noise generation capability of the UNet backbone used in our method under a supervised setting labeled "Paired". We also compare our method in the WUP case to several unsupervised denoising methods, including N2V [22], N2S [5], CVF-SID [31], AP-BSN + R3 [23], SCPGabNet [26], and SDAP(S)(E) [33]. Here, we use DRUNet [47] as the downstream denoising network to get better results.

Results: The results are presented in Table 2. Compared to other unpaired noise modeling methods, LUD-DIF achieves the best performance on most metrics. Notably, it shows substantial improvement across all three metrics for noisy image generation quality. The two model-based, non-learning noise modeling methods perform poorly in the noisy domain due to their inability to learn the underlying distributional characteristics of noise from data. The EOT-based ASBM method also yields subpar generation quality, which is attributed to error accumulation across different iteration steps. It is worth noting that the model trained on WUP data slightly outperforms the one trained on SUP data, indicating that implicit pairing within the dataset enhances the model's ability to capture data coupling. Furthermore, the performance gap between unpaired and paired training is qualitatively consistent with the weak-coupling analysis, since more accurate coupling information reduces the approximation error. Visual results, as shown in Figure 2, demonstrate that our method can more effectively characterize the spatial variation characteristics of noise.

Table 4: Quantitative comparison of pseudo-degraded image generation quality in the noisy domain on the AIM19 and NTIRE20 datasets.

Method	AIM19		NTIRE20	
	AKLD ↓	FID ↓	AKLD ↓	FID ↓
Bicubic	0.701	112.9	0.701	56.3
FSSR [10]	0.379	67.6	0.224	40.2
Impressionism [18]	0.549	87.3	0.275	26.5
DASR [44]	<u>0.328</u>	72.4	0.346	48.6
DeFlow [46]	0.356	119.8	0.156	34.7
LUD-VAE[48]	0.329	<u>57.2</u>	<u>0.120</u>	<u>24.9</u>
LUD-DIF (Ours)	**0.271**	**56.6**	**0.106**	**24.2**

Table 5: Quantitative comparison on the AIM19 and NTIRE20 datasets in terms of PSNR, SSIM, and LPIPS.

Method	AIM19			NTIRE20		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Bicubic	21.69	0.5517	0.517	20.45	0.3241	0.675
FSSR [10]	20.81	0.5242	0.387	21.07	0.4356	0.414
Impressionism [18]	21.99	<u>0.6060</u>	0.420	25.27	0.6731	0.229
DASR [44]	21.06	0.5658	0.375	23.70	0.5748	0.328
DeFlow [46]	21.06	0.5842	0.346	24.81	0.6777	0.225
LUD-VAE[48]	**22.32**	**0.6197**	<u>0.341</u>	**25.79**	<u>0.7178</u>	<u>0.219</u>
LUD-DIF (Ours)	<u>22.09</u>	0.6046	**0.332**	<u>25.77</u>	**0.7200**	**0.214**

5.3 Image Super-Resolution

Dataset: We employ two unpaired super-resolution datasets, AIM19 and NTIRE20, to evaluate the applicability of LUD-DIF for image super-resolution tasks. Track 2 of AIM19 and Track 1 of NTIRE20 both contain several sets of unpaired high-resolution images and degraded images. We crop these into 128×128 patches for training. Both datasets include a test set consisting of 100 paired low-resolution and high-resolution images, which we use to evaluate our method.

Implementation Details: The network architecture is the same as described in Section 5.1. Since the exact forward downsampling operator for the super-resolution problem is unknown, we approximate it using bicubic interpolation. Specifically,

$$y = D_{\text{bicubic}}(x) + n + D_{\text{real}}(x) - D_{\text{bicubic}}(x),$$

where D_{bicubic} denotes the bicubic interpolation downsampling operator, D_{real} represents the true downsampling operator, and n is the real noise. We combine the real noise with the approximation error of the downsampling operator as an effective composite noise to be learned by the noise model; this is a practical extension of the additive formulation used in the analysis. Due to the low noise intensity in the super-resolution task, we directly model the diffusion process for x, $p_{X_0|X_V}$, as a single-step denoising model $\mathbb{E}[X_0|X_V]$. In practice, for the AIM19 data, we set $\tau_V = 15, \tau_S = 10$; for the NTIRE20 data, we set $\tau_V = 8, \tau_S = 3$, consistent with the settings in [48].

Results: Table 4 compares the generated degradations with real degraded images using AKLD and FID. LUD-DIF achieves the best results on both metrics for the AIM19 and NTIRE20 datasets. For downstream evaluation, we train ESRGAN [41] using the generated degraded images paired with their corresponding high-resolution images, and report PSNR, SSIM, and LPIPS in Table 5. LUD-DIF achieves competitive restoration performance, including the best LPIPS on both datasets and the best SSIM on NTIRE20.

5.4 Ablation Study

Appendix B.1 tests the independence of generated noise samples. Appendix B.2 studies training and sampling hyperparameters. Appendix B.3 examines the effect of the degradation level τ_S, while Appendix B.4 analyzes the number of denoising steps used in the weak reconstruction. Finally, Appendix B.5 studies the role of the two-path diffusion architecture in noise modeling. Overall, these results support the practical choices used in the main experiments and are consistent with the weak-coupling trade-off described in Section 4.

6 Conclusion and Future Work

This paper proposes LUD-DIF, a diffusion-based method for unpaired data learning in inverse problems. Under the weak-coupling assumption, the joint ELBO is decoupled into two independently trainable diffusion processes; without this assumption, we quantitatively analyze the error bound and provide a theorem-motivated heuristic for hyperparameter selection. Experiments on simulated and real-world image noise and image super-resolution demonstrate the effectiveness of the method.

Despite these results, LUD-DIF has two main limitations. First, it requires a known or approximate forward operator, limiting its use in blind inverse problems and semantic modality translation. Second, although we bound the error introduced by the weak-coupling assumption, how to reduce this error remains open. Future work may investigate transformed or latent representations, such as wavelet spaces and neural-network embeddings, to tighten the bound, together with efficient sampling and model distillation to improve scalability.

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S. Brown. A high-quality denoising dataset for smartphone cameras. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1692–1700, 2018.

[2] Amir Adler, Mauricio Araya-Polo, and Tomaso Poggio. Deep learning for seismic inverse problems: Toward the acceleration of geophysical analysis workflows. *IEEE signal processing magazine*, 38(2):89–119, 2021.

[3] Andrei Arhire and Radu Timofte. Learned lightweight smartphone ISP with unpaired data. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR) Workshops*, pages 1878–1887, 2025.

[4] Dominique Bakry and Michael Émery. Diffusions hypercontractives. In Jacques Azéma and Marc Yor, editors, *Séminaire de Probabilités XIX 1983/84*, pages 177–206. Springer Berlin Heidelberg, 1985.

[5] Joshua Batson and Loic Royer. Noise2self: Blind denoising by self-supervision. In *International conference on machine learning*, pages 524–533. PMLR, 2019.

[6] Mario Bertero and Michele Piana. Inverse problems in biomedical imaging: Modeling and methods of solution. In Alfio Quarteroni, Luca Formaggia, and Alessandro Veneziani, editors, *Complex Systems in Biomedicine*, pages 1–33. Springer Milan, 2006.

[7] Tim Brooks, Ben Mildenhall, Tianfan Xue, Jiawen Chen, Dillon Sharlet, and Jonathan T Barron. Unprocessing images for learned raw denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11036–11045, 2019.

[8] Giannis Daras, Hyungjin Chung, Chieh-Hsin Lai, Yuki Mitsufuji, Jong Chul Ye, Peyman Milanfar, Alexandros G Dimakis, and Mauricio Delbracio. A survey on diffusion models for inverse problems. *arXiv:2410.00083*, 2024.

[9] Valentin De Bortoli, Iryna Korshunova, Andriy Mnih, and Arnaud Doucet. Schrödinger bridge flow for unpaired data translation. In *Advances in Neural Information Processing Systems*, volume 37, pages 103384–103441, 2024.

[10] Manuel Fritsche, Shuhang Gu, and Radu Timofte. Frequency separation for real-world super-resolution. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3599–3608. IEEE, 2019.

[11] Martin Genzel, Ingo Gühring, Jan Macdonald, and Maximilian März. Near-exact recovery for tomographic inverse problems via deep learning. In *International Conference on Machine Learning*, pages 7368–7381. PMLR, 2022.

[12] Xiang Gu, Liwei Yang, Jian Sun, and Zongben Xu. Optimal transport-guided conditional score-based diffusion model. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[13] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1712–1722, 2019.

[14] Nikita Gushchin, Daniil Selikhanovych, Sergei Kholkin, Evgeny Burnaev, and Alexander Korotin. Adversarial Schrödinger bridge matching. *Advances in Neural Information Processing Systems*, 37:89612–89651, 2024.

[15] 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.

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020.

[17] Geonwoon Jang, Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. C2N: Practical generative noise modeling for real-world denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2350–2359, 2021.

[18] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 466–467, 2020.

[19] Sergey I Kabanikhin. *Inverse and ill-posed problems: theory and applications*. de Gruyter, 2011.

[20] Shima Kamyab, Zohreh Azimifar, Rasool Sabzi, and Paul Fieguth. Deep learning methods for inverse problems. *PeerJ Computer Science*, 8:e951, 2022.

[21] Beomsu Kim, Gihyun Kwon, Kwanyoung Kim, and Jong Chul Ye. Unpaired image-to-image translation via neural Schrödinger bridge. In *The Twelfth International Conference on Learning Representations*, 2024.

[22] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2129–2137, 2019.

[23] Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. AP-BSN: Self-supervised denoising for real-world images via asymmetric pd and blind-spot network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17725–17734, 2022.

[24] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2Noise: Learning image restoration without clean data. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2965–2974. PMLR, 2018.

[25] Yu Li, Sheng Tang, Rui Zhang, Yongdong Zhang, Jintao Li, and Shuicheng Yan. Asymmetric GAN for unpaired image-to-image translation. *IEEE Transactions on Image Processing*, 28(12):5881–5896, 2019.

[26] Xin Lin, Chao Ren, Xiao Liu, Jie Huang, and Yinjie Lei. Unsupervised image denoising in real-world scenarios via self-collaboration parallel generative adversarial branches. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12642–12652, 2023.

[27] Ziyue Lin, Jiahe Hou, Hongyu Xia, Xinrui Xie, Feifei Wang, Yuyin Zhou, Wei Wang, Jiawei Liu, and Liangqiong Qu. Decoupled residual denoising diffusion models for unified and data efficient image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 35967–35977, 2026.

[28] Guan-Horng Liu, Yaron Lipman, Maximilian Nickel, Brian Karrer, Evangelos Theodorou, and Ricky T. Q. Chen. Generalized Schrödinger bridge matching. In *The Twelfth International Conference on Learning Representations*, 2024.

[29] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.

[30] Giacomo Meanti, Thomas Ryckeboer, Michael Arbel, and Julien Mairal. Unsupervised imaging inverse problems with diffusion distribution matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 28364–28374, 2025.

[31] Reyhaneh Neshatavar, Mohsen Yavartanoo, Sanghyun Son, and Kyoung Mu Lee. CVF-SID: Cyclic multi-variate function for self-supervised image denoising by disentangling noise from image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17583–17591, 2022.

[32] Gregory Ongie, Ajil Jalal, Christopher A Metzler, Richard G Baraniuk, Alexandros G Dimakis, and Rebecca Willett. Deep learning techniques for inverse problems in imaging. *IEEE Journal on Selected Areas in Information Theory*, 1(1):39–56, 2020.

[33] Yizhong Pan, Xiao Liu, Xiangyu Liao, Yuanzhouhan Cao, and Chao Ren. Random sub-samples generation for self-supervised real image denoising. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12150–12159, 2023.

[34] Long Peng, Wenbo Li, Renjing Pei, Jingjing Ren, Jiaqi Xu, Yang Wang, Yang Cao, and Zheng-Jun Zha. Towards realistic data generation for real-world super-resolution. In *The Thirteenth International Conference on Learning Representations*, 2025.

[35] Alejandro Ribes and Francis Schmitt. Linear inverse problems in imaging. *IEEE Signal Processing Magazine*, 25(4):84–99, 2008.

[36] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

[37] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[38] Alessio Spantini, Antti Solonen, Tiangang Cui, James Martin, Luis Tenorio, and Youssef Marzouk. Optimal low-rank approximations of Bayesian linear inverse problems. *SIAM Journal on Scientific Computing*, 37(6):A2451–A2487, 2015.

[39] Andrew M Stuart. Inverse problems: a Bayesian perspective. *Acta numerica*, 19:451–559, 2010.

[40] Joel A Tropp and Stephen J Wright. Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, 98(6):948–958, 2010.

[41] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. ESRGAN: Enhanced super-resolution generative adversarial networks. In *Computer Vision – ECCV 2018 Workshops*, pages 63–79. Springer, 2019.

[42] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[43] Yanyan Wei, Zhao Zhang, Yang Wang, Haijun Zhang, Mingbo Zhao, Mingliang Xu, and Meng Wang. Semi-deraingan: A new semi-supervised single image deraining. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2021.

[44] Yunxuan Wei, Shuhang Gu, Yawei Li, Radu Timofte, Longcun Jin, and Hengjie Song. Unsupervised real-world image super resolution via domain-distance aware training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13385–13394, 2021.

[45] Ralph A Wiggins. The general linear inverse problem: Implication of surface waves and free oscillations for earth structure. *Reviews of Geophysics*, 10(1):251–285, 1972.

[46] Valentin Wolf, Andreas Lugmayr, Martin Danelljan, Luc Van Gool, and Radu Timofte. DeFlow: Learning complex image degradations from unpaired data with conditional flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 94–103, 2021.

[47] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6360–6376, 2021.

[48] Dihan Zheng, Xiaowen Zhang, Kaisheng Ma, and Chenglong Bao. Learn from unpaired data for image restoration: A variational Bayes approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5889–5903, 2023.

A Theoretical details in LUD-DIF

A.1 Proof of Proposition 3.2

Since the noises in the forward processes are independent, X_t and Y_t are conditionally independent given (X_0, Y_0) for any t. Hence, (3.5a) directly holds. Similarly, the transition probability factorizes as

$$p_{Z_t|Z_{t-1}}(z_t|z_{t-1}) = p_{X_t|X_{t-1}}(x_t|x_{t-1})p_{Y_t|Y_{t-1}}(y_t|y_{t-1}). \tag{A.1}$$

Then it follows that

$$\begin{aligned}
p_{Z_{t-1}|Z_t,Z_0}(z_{t-1}|z_t,z_0) &= \frac{p_{Z_t|Z_{t-1},Z_0}(z_t|z_{t-1},z_0)p_{Z_{t-1}|Z_0}(z_{t-1}|z_0)}{p_{Z_t|Z_0}(z_t|z_0)} \\
&= \frac{p_{Z_t|Z_{t-1}}(z_t|z_{t-1})p_{Z_{t-1}|Z_0}(z_{t-1}|z_0)}{p_{Z_t|Z_0}(z_t|z_0)} \\
&= \frac{p_{X_t|X_{t-1}}(x_t|x_{t-1})p_{X_{t-1}|X_0}(x_{t-1}|x_0)}{p_{X_t|X_0}(x_t|x_0)} \frac{p_{Y_t|Y_{t-1}}(y_t|y_{t-1})p_{Y_{t-1}|Y_0}(y_{t-1}|y_0)}{p_{Y_t|Y_0}(y_t|y_0)} \\
&= p_{X_{t-1}|X_t,X_0}(x_{t-1}|x_t,x_0)p_{Y_{t-1}|Y_t,Y_0}(y_{t-1}|y_t,y_0),
\end{aligned}$$

where the first equality is due to Bayes' rule, the second equality invokes the Markov property of Z_t, the third equality follows from (3.5a) and (A.1), and the last equality follows from Bayes' rule.

A.2 Proof of Proposition 3.3

Using arguments similar to those in the marginal derivation in (2.4), we expand the joint ELBO:

$$\begin{aligned}
\mathrm{ELBO}_Z(\theta; z_0) = &- \overbrace{D_{\mathrm{KL}}(p_{Z_T|Z_0}(\cdot|z_0)\|p_{Z_T})}^{A_1} + \mathbb{E}\Big[\overbrace{\log p^{\theta}_{Z_0|Z_1}(z_0|Z_1)}^{A_2}\,|Z_0 = z_0\Big] \\
&- \sum_{t=2}^{T} \mathbb{E}\Big[\underbrace{D_{\mathrm{KL}}\big(p_{Z_{t-1}|Z_t,Z_0}(\cdot|Z_t,z_0)\|p^{\theta}_{Z_{t-1}|Z_t}(\cdot|Z_t)\big)}_{A_3}\,|Z_0 = z_0\Big].
\end{aligned} \tag{A.2}$$

The term A_1 in (A.2) is a constant independent of θ. For the term A_2, it follows directly from the reverse factorization (3.6) that

$$\log p^{\theta}_{Z_0|Z_1}(z_0|Z_1) = \log p^{\theta}_{X_0|X_1}(x_0|X_1) + \log p^{\theta}_{Y_0|X_0,Y_1}(y_0 \mid x_0, Y_1). \tag{A.3}$$

For the KL divergence term A_3 in (A.2), we apply both the forward posterior factorization (3.5b) and the reverse factorization (3.6):

$$\begin{aligned}
&D_{\mathrm{KL}}(p_{Z_{t-1}|Z_t,Z_0}(\cdot|Z_t,z_0)\|p^{\theta}_{Z_{t-1}|Z_t}(\cdot|Z_t)) \\
&= \mathbb{E}\left[\log \frac{p_{Z_{t-1}|Z_t,Z_0}(Z_{t-1}|Z_t,z_0)}{p^{\theta}_{Z_{t-1}|Z_t}(Z_{t-1}|Z_t)}\middle| Z_t, Z_0 = z_0\right] \\
&= \mathbb{E}\left[\log \frac{p_{X_{t-1}|X_t,X_0}(X_{t-1}|X_t,x_0)p_{Y_{t-1}|Y_t,Y_0}(Y_{t-1}|Y_t,y_0)}{p^{\theta}_{X_{t-1}|X_t}(X_{t-1}\mid X_t)p^{\theta}_{Y_{t-1}|X_{t-1},Y_t}(Y_{t-1}\mid X_{t-1},Y_t)}\middle| Z_t, Z_0 = z_0\right] \\
&= \mathbb{E}\left[\log \frac{p_{X_{t-1}|X_t,X_0}(X_{t-1}|X_t,x_0)}{p^{\theta}_{X_{t-1}|X_t}(X_{t-1}|X_t)}\middle| X_t, X_0 = x_0\right] \\
&\quad + \mathbb{E}\left[\log \frac{p_{Y_{t-1}|Y_t,Y_0}(Y_{t-1}|Y_t,y_0)}{p^{\theta}_{Y_{t-1}|X_{t-1},Y_t}(Y_{t-1}\mid X_{t-1},Y_t)}\middle| Z_t, Z_0 = z_0\right] \\
&= D_{\mathrm{KL}}(p_{X_{t-1}|X_t,X_0}(\cdot|X_t,x_0)\|p^{\theta}_{X_{t-1}|X_t}(\cdot|X_t)) \\
&\quad + \mathbb{E}\left[D_{\mathrm{KL}}\big(p_{Y_{t-1}|Y_t,Y_0}(\cdot|Y_t,y_0)\big\|p^{\theta}_{Y_{t-1}|X_{t-1},Y_t}(\cdot\mid X_{t-1},Y_t)\big)|Z_t,Z_0=z_0\right],
\end{aligned} \tag{A.4}$$

where the second equality follows from (3.5b) and (3.6). Substituting (A.3) and (A.4) into (A.2) yields

$$\mathrm{ELBO}_Z(\theta; z_0) = \mathrm{ELBO}_X(\theta; x_0) + \mathrm{ELBO}_{Y|X}(\theta; z_0) + C(z_0),$$

where $C(z_0) \coloneqq -D_{\mathrm{KL}}(p_{Z_T|Z_0}(\cdot|z_0)\|p_{Z_T})$ is independent of θ and can be evaluated using the standard Gaussian KL formula. This completes the proof.

A.3 Proof of Theorem 3.7

Proof of Theorem 3.7. Comparing the objectives in (3.13) and (3.17), it suffices to show

$$(X^{\mathrm{weak}}_{t-1}|Y_t,Y_0) \overset{\mathrm{d}}{=} (X_{t-1}|Y_t,Y_0),$$

where the weakly coupled variable is defined as

$$X^{\mathrm{weak}}_{t-1} \coloneqq \sqrt{\bar{\alpha}_{t-1}}X^{\mathrm{weak}}_0 + \sqrt{1-\bar{\alpha}_{t-1}}\xi. \tag{A.5}$$

According to the Chapman–Kolmogorov equation, we have

$$\begin{aligned}
&p_{X^{\mathrm{weak}}_{t-1}|Y_t,Y_0}(x_{t-1}|y_t,y_0) \\
&= \int \underbrace{p_{X^{\mathrm{weak}}_{t-1}|X^{\mathrm{weak}}_0,Y_t,Y_0}(x_{t-1}|x_0,y_t,y_0)}_{A_1}\underbrace{p_{X^{\mathrm{weak}}_0|Y_t,Y_0}(x_0|y_t,y_0)}_{A_2}\,\mathrm{d}x_0.
\end{aligned} \tag{A.6}$$

We first focus on the term A_1 in (A.6). By the definition of the weakly coupled variable in (A.5), X^{weak}_{t-1} is determined by X^{weak}_0 and an independent variable ξ. Therefore, X^{weak}_{t-1} is conditionally independent of (Y_t, Y_0) given X^{weak}_0. Moreover, the definition in (A.5) shows that the process $X^{\mathrm{weak}}_{0:T}$ shares the same forward process as $X_{0:T}$ in (2.2), i.e., $p_{X^{\mathrm{weak}}_{t-1}|X^{\mathrm{weak}}_0} = p_{X_{t-1}|X_0}$. Consequently, we have

$$p_{X^{\mathrm{weak}}_{t-1}|X^{\mathrm{weak}}_0,Y_t,Y_0}(x_{t-1}|x_0,y_t,y_0) = p_{X^{\mathrm{weak}}_{t-1}|X^{\mathrm{weak}}_0}(x_{t-1}|x_0) = p_{X_{t-1}|X_0}(x_{t-1}|x_0). \tag{A.7}$$

We next consider the term A_2 in (A.6). From (3.4), conditioning on a fixed $Y_0 = y_0$, the variable Y_t is determined by $\varepsilon^Y_{1:t-1}$, which is independent of X_0^{weak}. This implies

$$p_{X_0^{\text{weak}}|Y_t,Y_0}(x_0|y_t,y_0) = p_{X_0^{\text{weak}}|Y_0}(x_0|y_0) = p^{\text{weak}}_{X_0|Y_0}(x_0|y_0) = p_{X_0|Y_0}(x_0|y_0), \tag{A.8}$$

where the second equality is owing to the definition of X_0^{weak} in (3.14), and the last equality invokes Assumption 3.4. Substituting (A.7) and (A.8) into (A.6) yields

$$\begin{aligned} p_{X^{\text{weak}}_{t-1}|Y_t,Y_0}(x_{t-1}|y_t,y_0) &= \int p_{X_{t-1}|X_0}(x_{t-1}|x_0) p_{X_0|Y_0}(x_0|y_0)\,\mathrm{d}x_0 \\ &= \int p_{X_{t-1}|X_0,Y_t,Y_0}(x_{t-1}|x_0,y_t,y_0) p_{X_0|Y_t,Y_0}(x_0|y_t,y_0)\,\mathrm{d}x_0 \\ &= p_{X_{t-1}|Y_t,Y_0}(x_{t-1}|y_t,y_0), \end{aligned}$$

where the second equality uses the facts that X_{t-1} is conditionally independent of (Y_t, Y_0) given X_0 and that X_0 is conditionally independent of Y_t given Y_0, while the last equality follows from the Chapman–Kolmogorov equation. This completes the proof. □

A.4 Proof of Proposition 4.3

Proof of Proposition 4.3. Define an auxiliary function

$$D_t(x_0, Y_t, Y_0, \varepsilon^X) \coloneqq \|\mu^\theta_{Y|X,t}(Y_t, \sqrt{\bar\alpha_{t-1}}x_0 + \sqrt{1-\bar\alpha_{t-1}}\varepsilon^X) - \tilde\mu_t(Y_t, Y_0)\|_2^2.$$

Using (3.13) and (3.17), the triangle inequality yields

$$\Delta \le \sum_{t=1}^T \frac{1}{2\sigma_t^2} \left| \mathbb{E}\Big[D_t(X_0, Y_t, Y_0, \varepsilon^X)\Big] - \mathbb{E}\Big[D_t(X_0^{\text{weak}}, Y_t, Y_0, \varepsilon^X)\Big] \right|, \tag{A.9}$$

where $(X_0^{\text{weak}}|Y_0 = y_0) \sim p^{\text{weak}}_{X_0|Y_0}(\cdot|y_0)$ and $\varepsilon^X \sim \mathcal{N}(0, I_d)$. Using Jensen's inequality, we have

$$\begin{aligned} &\left| \mathbb{E}\left[D_t(X_0, Y_t, Y_0, \varepsilon^X) \,\middle|\, Y_0, Y_t, \varepsilon^X\right] - \mathbb{E}\left[D_t(X_0^{\text{weak}}, Y_t, Y_0, \varepsilon^X) \,\middle|\, Y_0, Y_t, \varepsilon^X\right] \right| \\ &\le \int_{\mathcal{X}} D_t(x_0, Y_t, Y_0, \varepsilon^X) |p_{X_0|Y_0}(x_0|Y_0) - p^{\text{weak}}_{X_0|Y_0}(x_0|Y_0)| \mathrm{d}x_0. \end{aligned}$$

It follows from Assumptions 4.2 and 4.1 that

$$D_t(x_0, Y_t, Y_0, \varepsilon^X) \le K'(1 + \|Y_t\|_2^{\max(m,2)} + \|\varepsilon^X\|_2^{\max(m,2)} + \|Y_0\|_2^2), \quad x_0 \in \mathcal{X},$$

where K' is a constant depending on C, $R_{\mathcal{X}}$, σ_X, σ_Y, and the variance schedule of the diffusion model. As a consequence,

$$\begin{aligned} &\int_{\mathcal{X}} D_t(x_0, Y_t, Y_0, \varepsilon^X) |p_{X_0|Y_0}(x_0|Y_0) - p^{\text{weak}}_{X_0|Y_0}(x_0|Y_0)| \mathrm{d}x_0 \\ &\le 2K'(1 + \|Y_t\|_2^{\max(m,2)} + \|\varepsilon^X\|_2^{\max(m,2)} + \|Y_0\|_2^2) \|p_{X_0|Y_0}(\cdot|Y_0) - p^{\text{weak}}_{X_0|Y_0}(\cdot|Y_0)\|_{\text{TV}}. \end{aligned}$$

Taking expectations on both sides of the inequality implies

$$\begin{aligned} &\left| \mathbb{E}\Big[D_t(X_0, Y_t, Y_0, \varepsilon^X)\Big] - \mathbb{E}\Big[D_t(X_0^{\text{weak}}, Y_t, Y_0, \varepsilon^X)\Big] \right| \\ &\le \mathbb{E}\left[\left| \mathbb{E}\left[D_t(X_0, Y_t, Y_0, \varepsilon^X) \,\middle|\, Y_0, Y_t, \varepsilon^X\right] - \mathbb{E}\left[D_t(X_0^{\text{weak}}, Y_t, Y_0, \varepsilon^X) \,\middle|\, Y_0, Y_t, \varepsilon^X\right] \right| \right] \\ &\le \mathbb{E}\Big[2K'(1 + \|Y_t\|_2^{\max(m,2)} + \|\varepsilon^X\|_2^{\max(m,2)} + \|Y_0\|_2^2) \|p_{X_0|Y_0}(\cdot|Y_0) - p^{\text{weak}}_{X_0|Y_0}(\cdot|Y_0)\|_{\text{TV}}\Big] \\ &\le 2K' \mathbb{E}^{\frac{1}{2}}\Big[(1 + \|Y_t\|_2^{\max(m,2)} + \|\varepsilon^X\|_2^{\max(m,2)} + \|Y_0\|_2^2)^2\Big] \mathbb{E}^{\frac{1}{2}}\Big[\|p_{X_0|Y_0}(\cdot|Y_0) - p^{\text{weak}}_{X_0|Y_0}(\cdot|Y_0)\|_{\text{TV}}^2\Big] \\ &\le M_t \mathbb{E}^{\frac{1}{2}}\Big[\|p_{X_0|Y_0}(\cdot|Y_0) - p^{\text{weak}}_{X_0|Y_0}(\cdot|Y_0)\|_{\text{TV}}\Big], \end{aligned} \tag{A.10}$$

where the third inequality is due to the Cauchy–Schwarz inequality, and the last inequality uses Assumption 4.2 and the property $\|p\|_{\text{TV}} \le 1$. Here M_t is a constant depending on C, $R_{\mathcal{X}}$, $M_{\Xi,m}$, d, σ_X, σ_Y, and the variance schedule. Substituting (A.10) into (A.9) completes the proof. □

A.5 Proof of Theorem 4.5

Before presenting the main proof of the theorem, we first establish the following lemmas.

Lemma A.1 (KL divergence between Gaussians)**.** *Let $\mu_1, \mu_2 \in \mathbb{R}^d$, and let $\sigma_1, \sigma_2 > 0$. Then*

$$D_{\mathrm{KL}}(\mathcal{N}(\mu_1, \sigma_1^2 I_d) \,\|\, \mathcal{N}(\mu_2, \sigma_2^2 I_d)) = \frac{d}{2}\left(\frac{\sigma_1^2}{\sigma_2^2} - 1 - \log\frac{\sigma_1^2}{\sigma_2^2}\right) + \frac{\|\mu_1 - \mu_2\|_2^2}{2\sigma_2^2}.$$

Lemma A.2 (Gaussian-smoothed KL-divergence bound)**.** *Let $U \in \mathbb{R}^d$ be a random variable with a bounded density function $p_U \le M_U$ on $\mathbb{R}^d$ and a finite second moment $\mathbb{E}[\|U\|_2^2] = d\sigma_U^2$. Let $G \sim \mathcal{N}(0, \sigma_U^2 I_d)$ be an independent Gaussian random variable. Then we have the following properties:*

(i) The KL divergence of p_U with respect to p_G is finite, i.e., $D_{\mathrm{KL}}(p_U \,\|\, p_G) < \infty$.

(ii) For any $\sigma_V > 0$, the KL divergence of the Gaussian-smoothed distributions satisfies the bound

$$D_{\mathrm{KL}}\big(p_U * \gamma_{\sigma_V^2} \,\|\, p_G * \gamma_{\sigma_V^2}\big) \le \left(\frac{\sigma_U^2}{\sigma_U^2 + \sigma_V^2}\right) D_{\mathrm{KL}}(p_U \,\|\, p_G),$$

where $*$ denotes the convolution, and $\gamma_{\sigma_V^2}$ represents the density of $\mathcal{N}(0, \sigma_V^2 I_d)$.

Proof. **Part (i):** We first prove that $D_{\mathrm{KL}}(p_U \,\|\, p_G) < \infty$. By the definition of KL divergence, we decompose it into negative differential entropy and cross-entropy:

$$D_{\mathrm{KL}}(U \,\|\, G) = \int_{\mathbb{R}^d} p_U(x) \log p_U(x) \mathrm{d}x - \int_{\mathbb{R}^d} p_U(x) \log p_G(x) \mathrm{d}x.$$

Since $p_U \le M_U$, the negative differential entropy is strictly bounded above:

$$\int_{\mathbb{R}^d} p_U(x) \log p_U(x) \mathrm{d}x \le \int_{\mathbb{R}^d} p_U(x) \log(M_U) \mathrm{d}x = \log M_U.$$

For the cross-entropy term, substituting the Gaussian density yields

$$-\int_{\mathbb{R}^d} p_U(x) \log p_G(x) \mathrm{d}x = \frac{d}{2}\log(2\pi\sigma_U^2) + \frac{1}{2\sigma_U^2}\int_{\mathbb{R}^d} \|x\|_2^2 p_U(x) \mathrm{d}x = \frac{d}{2}\log(2\pi e \sigma_U^2) < \infty.$$

Summing these two bounds guarantees $D_{\mathrm{KL}}(p_U \,\|\, p_G) \le \log M_U + \frac{d}{2}\log(2\pi e\sigma_U^2) < \infty$.

Part (ii): Consider the Ornstein–Uhlenbeck (OU) processes

$$\begin{aligned} \mathrm{d}U_t &= -U_t \,\mathrm{d}t + \sqrt{2}\sigma_U \,\mathrm{d}W_t^U, \quad U_0 \sim p_U, \\ \mathrm{d}G_t &= -G_t \,\mathrm{d}t + \sqrt{2}\sigma_U \,\mathrm{d}W_t^G, \quad G_0 \sim p_G, \end{aligned} \tag{A.11}$$

where W_t^U and W_t^G are d-dimensional independent Wiener processes. The stationary distribution of these OU processes is $\mathcal{N}(0, \sigma_U^2 I_d)$, which satisfies a log-Sobolev inequality. From [4], p_{U_t} converges to the stationary distribution $\mathcal{N}(0, \sigma_U^2 I_d)$ exponentially:

$$D_{\mathrm{KL}}\big(p_{U_t} \,\|\, \gamma_{\sigma_U^2}\big) \le e^{-2t} D_{\mathrm{KL}}\big(p_U \,\|\, \gamma_{\sigma_U^2}\big) = e^{-2t} D_{\mathrm{KL}}(p_U \,\|\, p_G). \tag{A.12}$$

On the other hand, since $G \sim \mathcal{N}(0, \sigma_U^2 I_d)$, we have $G_t \sim \mathcal{N}(0, \sigma_U^2 I_d)$ for any $t > 0$. Combining this with (A.12) yields

$$D_{\mathrm{KL}}\big(p_{U_t} \,\|\, p_{G_t}\big) \le e^{-2t} D_{\mathrm{KL}}(p_U \,\|\, p_G). \tag{A.13}$$

Note that the two linear SDEs in (A.11) admit explicit solutions

$$U_t = e^{-t}U + \sqrt{1 - e^{-2t}}\sigma_U \varepsilon_U, \quad G_t = e^{-t}G + \sqrt{1 - e^{-2t}}\sigma_U \varepsilon_G,$$

where $(\varepsilon_U, \varepsilon_G) \sim \mathcal{N}(0, I_d) \otimes \mathcal{N}(0, I_d)$ is independent of U and G. As a result, $e^t U_t \sim p_U * \gamma_{\sigma_t^2}$ and $e^t G_t \sim p_G * \gamma_{\sigma_t^2}$, where $\sigma_t \coloneqq \sqrt{e^{2t}-1}\sigma_U$. Then for any $t > 0$, since the KL divergence is invariant under the affine map, we have

$$D_{\mathrm{KL}}\big(p_U * \gamma_{\sigma_t^2} \,\|\, p_G * \gamma_{\sigma_t^2}\big) = D_{\mathrm{KL}}\big(p_{e^t U_t} \,\|\, p_{e^t G_t}\big) = D_{\mathrm{KL}}\big(p_{U_t} \,\|\, p_{G_t}\big). \tag{A.14}$$

By combining (A.13) and (A.14), we have $D_{\mathrm{KL}}\big(p_U * \gamma_{\sigma_t^2} \,\|\, p_G * \gamma_{\sigma_t^2}\big) \le e^{-2t} D_{\mathrm{KL}}(p_U \,\|\, p_G)$. Setting $t = \frac{1}{2}\log(\frac{\sigma_U^2+\sigma_V^2}{\sigma_U^2})$ completes the proof. □

Proof of Theorem 4.5. Recall the forward processes (3.4) and the normalization (4.2):

$$\begin{aligned} X_V &= \sqrt{\bar\alpha_V} X_0 + \sqrt{1-\bar\alpha_V}\varepsilon_X, \\ Y_S &= \sqrt{\bar\alpha_S} Y_0 + \sqrt{1-\bar\alpha_S}\varepsilon_Y = \sqrt{\bar\alpha_S}\frac{\sigma_X}{\sigma_Y} X_0 + \sqrt{\bar\alpha_S}\frac{1}{\sigma_Y}\Xi + \sqrt{1-\bar\alpha_S}\varepsilon_Y, \end{aligned} \tag{A.15}$$

where $\varepsilon_X, \varepsilon_Y \sim \mathcal{N}(0, I_d)$. The proof is divided into two steps.

Step 1. Perturbation error estimate. Since X_0 and Y_S are conditionally independent given Y_0, for any $(x_0, y_S) \in \mathcal{X} \times \mathbb{R}^d$, we have

$$p_{X_0|Y_S}(x_0|y_S) = \int p_{X_0|Y_0}(x_0|y_0')\, p_{Y_0|Y_S}(y_0'|y_S)\,\mathrm{d}y_0' = \mathbb{E}_{Y_0'}\left[p_{X_0|Y_0}(x_0|Y_0') \mid Y_S = y_S\right].$$

As a result, using the law of total expectation and Jensen's inequality, we obtain

$$\begin{aligned} \mathcal{E}_{\mathrm{pert}} &\coloneqq \mathbb{E}_{Y_0,Y_S}\left[\|p_{X_0|Y_S}(\cdot|Y_S) - p_{X_0|Y_0}(\cdot|Y_0)\|_{\mathrm{TV}}\right] \\ &\le \mathbb{E}_{Y_S}\left[\mathbb{E}_{Y_0,Y_0'}\left[\|p_{X_0|Y_0}(\cdot|Y_0') - p_{X_0|Y_0}(\cdot|Y_0)\|_{\mathrm{TV}} \mid Y_S\right]\right], \end{aligned}$$

where Y_0 and Y_0' are conditionally independent and identically distributed given Y_S. Further, under Assumption 4.4, we have

$$\mathcal{E}_{\mathrm{pert}} \le L_{\mathrm{post}} \mathbb{E}_{Y_S}\left[\mathbb{E}_{Y_0,Y_0'}\left[\|Y_0' - Y_0\|_2 \mid Y_S\right]\right], \tag{A.16}$$

where Y_0, Y_0' are conditionally independent and identically distributed given Y_S. Then

$$\begin{aligned} \mathbb{E}_{Y_0,Y_0'}\left[\|Y_0' - Y_0\|_2^2 \mid Y_S\right] &= \mathbb{E}_{Y_0,Y_0'}\left[\|Y_0' - \mathbb{E}[Y_0' \mid Y_S] + \mathbb{E}[Y_0 \mid Y_S] - Y_0\|_2^2 \mid Y_S\right] \\ &= \mathbb{E}_{Y_0'}\left[\|Y_0' - \mathbb{E}[Y_0' \mid Y_S]\|_2^2 \mid Y_S\right] + \mathbb{E}_{Y_0}\left[\|\mathbb{E}[Y_0 \mid Y_S] - Y_0\|_2^2 \mid Y_S\right] \\ &= 2\mathbb{E}_{Y_0}\left[\|\mathbb{E}[Y_0 \mid Y_S] - Y_0\|_2^2 \mid Y_S\right]. \end{aligned} \tag{A.17}$$

For any $z \in \mathbb{R}^d$, we have

$$\begin{aligned} \mathbb{E}_{Y_0}\left[\|z - Y_0\|_2^2 \mid Y_S\right] &= \mathbb{E}_{Y_0}\left[\|z - \mathbb{E}[Y_0 \mid Y_S] + \mathbb{E}[Y_0 \mid Y_S] - Y_0\|_2^2 \mid Y_S\right] \\ &= \mathbb{E}_{Y_0}\left[\|z - \mathbb{E}[Y_0 \mid Y_S]\|_2^2 \mid Y_S\right] + \mathbb{E}_{Y_0}\left[\|\mathbb{E}[Y_0 \mid Y_S] - Y_0\|_2^2 \mid Y_S\right] \\ &\quad + 2\mathbb{E}_{Y_0}\left[\langle z - \mathbb{E}[Y_0 \mid Y_S], \mathbb{E}[Y_0 \mid Y_S] - Y_0\rangle \mid Y_S\right] \\ &= \mathbb{E}_{Y_0}\left[\|z - \mathbb{E}[Y_0 \mid Y_S]\|_2^2 \mid Y_S\right] + \mathbb{E}_{Y_0}\left[\|\mathbb{E}[Y_0 \mid Y_S] - Y_0\|_2^2 \mid Y_S\right] \\ &\ge \mathbb{E}_{Y_0}\left[\|\mathbb{E}[Y_0 \mid Y_S] - Y_0\|_2^2 \mid Y_S\right]. \end{aligned}$$

Setting $z = \frac{1}{\sqrt{\bar\alpha_S}} Y_S$ in this inequality and using (A.15), we have

$$\mathbb{E}_{Y_0}\left[\|\mathbb{E}[Y_0 \mid Y_S] - Y_0\|_2^2 \mid Y_S\right] \le \mathbb{E}_{Y_0}\left[\|\sqrt{\frac{1-\bar\alpha_S}{\bar\alpha_S}}\varepsilon_Y\|_2^2 \,\middle|\, Y_S\right], \tag{A.18}$$

where $\varepsilon_Y \sim \mathcal{N}(0, I_d)$ is independent of Y_0. Combining (A.17) and (A.18) and taking the expectation with respect to Y_S yields

$$\mathbb{E}_{Y_S}\left[\mathbb{E}_{Y_0,Y_0'}\left[\|Y_0' - Y_0\|_2^2 \mid Y_S\right]\right] \le 2\mathbb{E}_{\varepsilon_Y \sim \mathcal{N}(0,I_d)}\left[\|\sqrt{\frac{1-\bar\alpha_S}{\bar\alpha_S}}\varepsilon_Y\|_2^2\right] = 2d\frac{1-\bar\alpha_S}{\bar\alpha_S}. \tag{A.19}$$

Substituting (A.19) into (A.16) and using Jensen's inequality, we obtain

$$\mathcal{E}_{\text{pert}} \le L_{\text{post}} \sqrt{\mathbb{E}_{Y_S}\big[\mathbb{E}_{Y_0, Y_0'}\big[\|Y_0' - Y_0\|_2^2 \mid Y_S\big]\big]} \le L_{\text{post}} \sqrt{2d \frac{1-\bar{\alpha}_S}{\bar{\alpha}_S}}. \tag{A.20}$$

Step 2. Alignment error estimate. Applying the triangle inequality, the alignment error can be decomposed as

$$\begin{aligned}
\mathcal{E}_{\text{align}} &\coloneqq \mathbb{E}_{Y_S}\big[\|p_{X_0|X_V}(\cdot|Y_S) - p_{X_0|Y_S}(\cdot|Y_S)\|_{\text{TV}}\big] \\
&= \frac{1}{2} \iint |p_{X_0|X_V}(x_0|y_S) - p_{X_0|Y_S}(x_0|y_S)|\, p_{Y_S}(y_S)\, \mathrm{d}x_0\, \mathrm{d}y_S \\
&\le \frac{1}{2} \iint p_{X_0|X_V}(x_0|y_S) |p_{Y_S}(y_S) - p_{X_V}(y_S)|\, \mathrm{d}x_0\, \mathrm{d}y_S \\
&\quad + \frac{1}{2} \iint |p_{X_0|X_V}(x_0|y_S) p_{X_V}(y_S) - p_{X_0|Y_S}(x_0|y_S) p_{Y_S}(y_S)|\, \mathrm{d}x_0\, \mathrm{d}y_S \\
&= \|p_{Y_S} - p_{X_V}\|_{\text{TV}} + \mathbb{E}_{X_0}\big[\|p_{X_V|X_0}(\cdot|X_0) - p_{Y_S|X_0}(\cdot|X_0)\|_{\text{TV}}\big],
\end{aligned}$$

where the last equality is due to Bayes' rule. For the first summand, it follows from Jensen's inequality that

$$\begin{aligned}
\|p_{Y_S} - p_{X_V}\|_{\text{TV}} &= \| \int p_{Y_S|X_0}(\cdot|x_0) p_{X_0}(x_0)\, \mathrm{d}x_0 - \int p_{X_V|X_0}(\cdot|x_0) p_{X_0}(x_0)\, \mathrm{d}x_0 \|_{\text{TV}} \\
&\le \mathbb{E}_{X_0}\big[\|p_{X_V|X_0}(\cdot|X_0) - p_{Y_S|X_0}(\cdot|X_0)\|_{\text{TV}}\big].
\end{aligned}$$

As a consequence, we have

$$\mathcal{E}_{\text{align}} \le 2\mathbb{E}_{X_0}\big[\|p_{X_V|X_0}(\cdot|X_0) - p_{Y_S|X_0}(\cdot|X_0)\|_{\text{TV}}\big]. \tag{A.21}$$

From (A.15), we have

$$\begin{aligned}
(X_V \mid X_0 = x_0) &\overset{\text{d}}{=} \sqrt{\bar{\alpha}_V} x_0 + \sqrt{1-\bar{\alpha}_V}\varepsilon_X, \\
(Y_S \mid X_0 = x_0) &\overset{\text{d}}{=} \sqrt{\bar{\alpha}_S}\frac{\sigma_X}{\sigma_Y} x_0 + \sqrt{\bar{\alpha}_S}\frac{1}{\sigma_Y}\Xi + \sqrt{1-\bar{\alpha}_S}\varepsilon_Y.
\end{aligned}$$

Then we construct two auxiliary random variables

$$\begin{aligned}
(E \mid X_0 = x_0) &\overset{\text{d}}{=} \sqrt{\bar{\alpha}_S}\frac{\sigma_X}{\sigma_Y} x_0 + \sqrt{1-\bar{\alpha}_V}\varepsilon_X, \\
(F \mid X_0 = x_0) &\overset{\text{d}}{=} \sqrt{\bar{\alpha}_S}\frac{\sigma_X}{\sigma_Y} x_0 + \sqrt{\bar{\alpha}_S}\frac{1}{\sigma_Y}\varepsilon_\Xi + \sqrt{1-\bar{\alpha}_S}\varepsilon_Y,
\end{aligned}$$

where $\varepsilon_\Xi \sim \mathcal{N}(0, \sigma_\Xi^2 I_d)$. Applying the triangle inequality gives, for any $x_0 \in \mathcal{X}$,

$$\begin{aligned}
&\|p_{X_V|X_0}(\cdot|x_0) - p_{Y_S|X_0}(\cdot|x_0)\|_{\text{TV}} \\
&\le \underbrace{\|p_{X_V|X_0}(\cdot|x_0) - p_{E|X_0}(\cdot|x_0)\|_{\text{TV}}}_{\text{mean shift}} + \underbrace{\|p_{E|X_0}(\cdot|x_0) - p_{F|X_0}(\cdot|x_0)\|_{\text{TV}}}_{\text{variance mismatch}} \\
&\quad + \underbrace{\|p_{F|X_0}(\cdot|x_0) - p_{Y_S|X_0}(\cdot|x_0)\|_{\text{TV}}}_{\text{error of Gaussian approximation}},
\end{aligned} \tag{A.22}$$

For the first summand in (A.22), using Pinsker's inequality and Lemma A.1, we have

$$\begin{aligned}
\|p_{X_V|X_0}(\cdot|x_0) - p_{E|X_0}(\cdot|x_0)\|_{\text{TV}} &\le \sqrt{\frac{1}{2} D_{\text{KL}}(p_{X_V|X_0}(\cdot|x_0) \| p_{E|X_0}(\cdot|x_0))} \\
&\le \frac{|\sigma_Y\sqrt{\bar{\alpha}_V} - \sigma_X\sqrt{\bar{\alpha}_S}|\|x_0\|_2}{2\sigma_Y\sqrt{1-\bar{\alpha}_V}}.
\end{aligned} \tag{A.23}$$

For the second summand in (A.22), Pinsker's inequality and Lemma A.1 imply

$$\begin{aligned}\|p_{E|X_0}(\cdot|x_0) - p_{F|X_0}(\cdot|x_0)\|_{\mathrm{TV}} &\le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p_{E|X_0}(\cdot|x_0) \,\|\, p_{F|X_0}(\cdot|x_0))} \\ &\le \sqrt{\frac{d}{2}} \frac{|\sigma_X^2 \bar{\alpha}_S - \sigma_Y^2 \bar{\alpha}_V|}{\sigma_Y \sqrt{1-\bar{\alpha}_V} \sqrt{\sigma_Y^2 - \bar{\alpha}_S \sigma_X^2}} \\ &\le \sqrt{\frac{d}{2}} \frac{(\sigma_X + \sigma_Y)|\sigma_X \sqrt{\bar{\alpha}_S} - \sigma_Y \sqrt{\bar{\alpha}_V}|}{\sigma_Y \sqrt{1-\bar{\alpha}_V} \sqrt{\sigma_Y^2 - \bar{\alpha}_S \sigma_X^2}} \\ &\le \sqrt{\frac{2d}{\sigma_Y^2 - \bar{\alpha}_S \sigma_X^2}} \frac{|\sigma_X \sqrt{\bar{\alpha}_S} - \sigma_Y \sqrt{\bar{\alpha}_V}|}{\sqrt{1-\bar{\alpha}_V}}, \end{aligned} \tag{A.24}$$

For the third summand in (A.22), it follows from Lemma A.2 that

$$\begin{aligned}\|p_{F|X_0}(\cdot|x_0) - p_{Y_S|X_0}(\cdot|x_0)\|_{\mathrm{TV}} &\le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p_{Y_S|X_0}(\cdot|x_0) \,\|\, p_{F|X_0}(\cdot|x_0))} \\ &\le \sqrt{\frac{\bar{\alpha}_S \sigma_\Xi^2}{2(\sigma_Y^2 - \bar{\alpha}_S \sigma_X^2)}} \sqrt{D_{\mathrm{KL}}(p_\Xi \,\|\, \mathcal{N}(0, \sigma_\Xi^2 I_d))}. \end{aligned} \tag{A.25}$$

Substituting (A.23), (A.24) and (A.25) into (A.22) yields, for any $x_0 \in \mathcal{X}$,

$$\begin{aligned}&\|p_{X_V|X_0}(\cdot|x_0) - p_{Y_S|X_0}(\cdot|x_0)\|_{\mathrm{TV}} \\ &\le \left(\frac{R_{\mathcal{X}}}{2\sigma_Y} + \sqrt{\frac{2d}{\sigma_Y^2 - \bar{\alpha}_S \sigma_X^2}} \right) \frac{|\sigma_Y \sqrt{\bar{\alpha}_V} - \sigma_X \sqrt{\bar{\alpha}_S}|}{\sqrt{1-\bar{\alpha}_V}} \\ &\quad + \sqrt{\frac{\bar{\alpha}_S \sigma_\Xi^2}{2(\sigma_Y^2 - \bar{\alpha}_S \sigma_X^2)}} \sqrt{D_{\mathrm{KL}}(p_\Xi \,\|\, \mathcal{N}(0, \sigma_\Xi^2 I_d))}. \end{aligned}$$

Substituting this into (A.21) implies

$$\begin{aligned}\mathcal{E}_{\mathrm{align}} &\le \left(\frac{R_{\mathcal{X}}}{\sigma_Y} + \sqrt{\frac{8d}{\sigma_Y^2 - \bar{\alpha}_S \sigma_X^2}} \right) \frac{|\sigma_Y \sqrt{\bar{\alpha}_V} - \sigma_X \sqrt{\bar{\alpha}_S}|}{\sqrt{1-\bar{\alpha}_V}} \\ &\quad + \sqrt{\frac{2\bar{\alpha}_S \sigma_\Xi^2}{\sigma_Y^2 - \bar{\alpha}_S \sigma_X^2}} \sqrt{D_{\mathrm{KL}}(p_\Xi \,\|\, \mathcal{N}(0, \sigma_\Xi^2 I_d))}. \end{aligned} \tag{A.26}$$

Finally, combining (4.4), (A.20) and (A.26) completes the proof. □

A.6 Proof of Corollary 4.6

Proof of Corollary 4.6. By the same argument used to obtain (A.21), we have

$$\mathbb{E}_{X_0}\big[\|p_{Y_0|X_0}(\cdot|X_0) - p^{\mathrm{weak}}_{Y_0|X_0}(\cdot|X_0)\|_{\mathrm{TV}}\big] \le 2\mathbb{E}_{Y_0}\big[\|p_{X_0|Y_0}(\cdot|Y_0) - p^{\mathrm{weak}}_{X_0|Y_0}(\cdot|Y_0)\|_{\mathrm{TV}}\big].$$

Combining this with Theorem 4.5 completes the proof. □

A.7 Extension to signal-dependent noise

The analysis in Section 4 assumes that the additive noise is independent of the clean signal. Here we summarize how the same argument extends to signal-dependent noise. Consider the normalized observation model

$$Y_0 = \frac{\sigma_X}{\sigma_Y} X_0 + \frac{1}{\sigma_Y} \Xi(X_0), \tag{A.27}$$

where the conditional law of $\Xi(X_0)$ may depend on X_0. We assume that X_0 is supported on the compact set $\mathcal{X}$, and, for every $x \in \mathcal{X}$,

$$\mathbb{E}[\Xi(X_0) \mid X_0 = x] = 0, \qquad \mathrm{Cov}(\Xi(X_0) \mid X_0 = x) = \Sigma_x.$$

The conditional densities are uniformly bounded, the traces $\mathrm{Tr}(\Sigma_x)$ and the moments required by Assumption 4.1 are uniformly bounded, and the conditional noise field is independent of the Gaussian diffusion noises. Define

$$\sigma_\Xi^2 \coloneqq \frac{1}{d}\mathbb{E}_{X_0}[\mathrm{Tr}(\Sigma_{X_0})].$$

Conditional centering then gives the averaged variance identity $\sigma_X^2 + \sigma_\Xi^2 = \sigma_Y^2$.

Let

$$\mathfrak{D}_{\Xi|X} \coloneqq \mathbb{E}_{X_0}\left[\sqrt{D_{\mathrm{KL}}\big(p_{\Xi|X_0}(\cdot \mid X_0) \,\big\|\, \mathcal{N}(0, \sigma_\Xi^2 I_d)\big)}\right].$$

This quantity captures anisotropy, spatially varying conditional variance, and higher-order non-Gaussian structure through a single trace-matched Gaussian reference. Assume additionally that the posterior map is L_{post}-Lipschitz in total variation, as in Assumption 4.4.

Proposition A.3 (Conditional-noise extension)**.** *Under the conditions above and Assumption 4.1, the loss gap is controlled by the posterior mismatch exactly as in Proposition 4.3. If the diffusion times satisfy the trace-averaged alignment condition*

$$\bar\alpha_V \sigma_Y^2 = \bar\alpha_S \sigma_X^2,$$

then

$$\begin{aligned}\Delta_p \le L_{\mathrm{post}} &\sqrt{\frac{2d(1-\bar\alpha_S)}{\bar\alpha_S}} \\ &+ \sqrt{2}\sqrt{\frac{\bar\alpha_S \sigma_\Xi^2}{\sigma_Y^2 - \bar\alpha_S \sigma_X^2}}\,\mathfrak{D}_{\Xi|X}.\end{aligned} \tag{A.28}$$

Consequently,

$$\Delta \le \sum_{t=1}^{T} \frac{M_t}{\sigma_t^2}\sqrt{\Delta_p},$$

for finite constants M_t depending on the diffusion schedule, the network growth bound, and the stated moment bounds.

Proof sketch. Conditioning on $X_0 = x$, compare the forward laws of X_V and Y_S with a Gaussian having the global trace-matched variance. Pinsker's inequality splits their total-variation discrepancy into mean, variance, and conditional non-Gaussianity terms. The alignment relation cancels the first two terms, leaving the second term in (A.28). The posterior Lipschitz property bounds the information loss incurred by diffusing Y_0 to Y_S, which gives the first term. The loss-gap estimate then follows by the same conditional-expectation and Cauchy–Schwarz argument used in Proposition 4.3. □

For completeness, the posterior stability assumption has a simple sufficient condition in the independent-noise special case. If $p_\Xi > 0$, $\log p_\Xi \in C^2(\mathbb{R}^d)$, and $\sup_\xi \|\nabla^2 \log p_\Xi(\xi)\|_{\mathrm{op}} \le H_\Xi$, then

$$\big\|p_{X_0|Y_0}(\cdot \mid y_1) - p_{X_0|Y_0}(\cdot \mid y_2)\big\|_{\mathrm{TV}} \le \frac{\sigma_X \sigma_Y H_\Xi}{4}\,\mathrm{diam}(\mathcal{X})\,\|y_1 - y_2\|_2. \tag{A.29}$$

For Gaussian noise $\mathcal{N}(0, \Sigma)$, one may take $H_\Xi = \|\Sigma^{-1}\|_{\mathrm{op}}$.

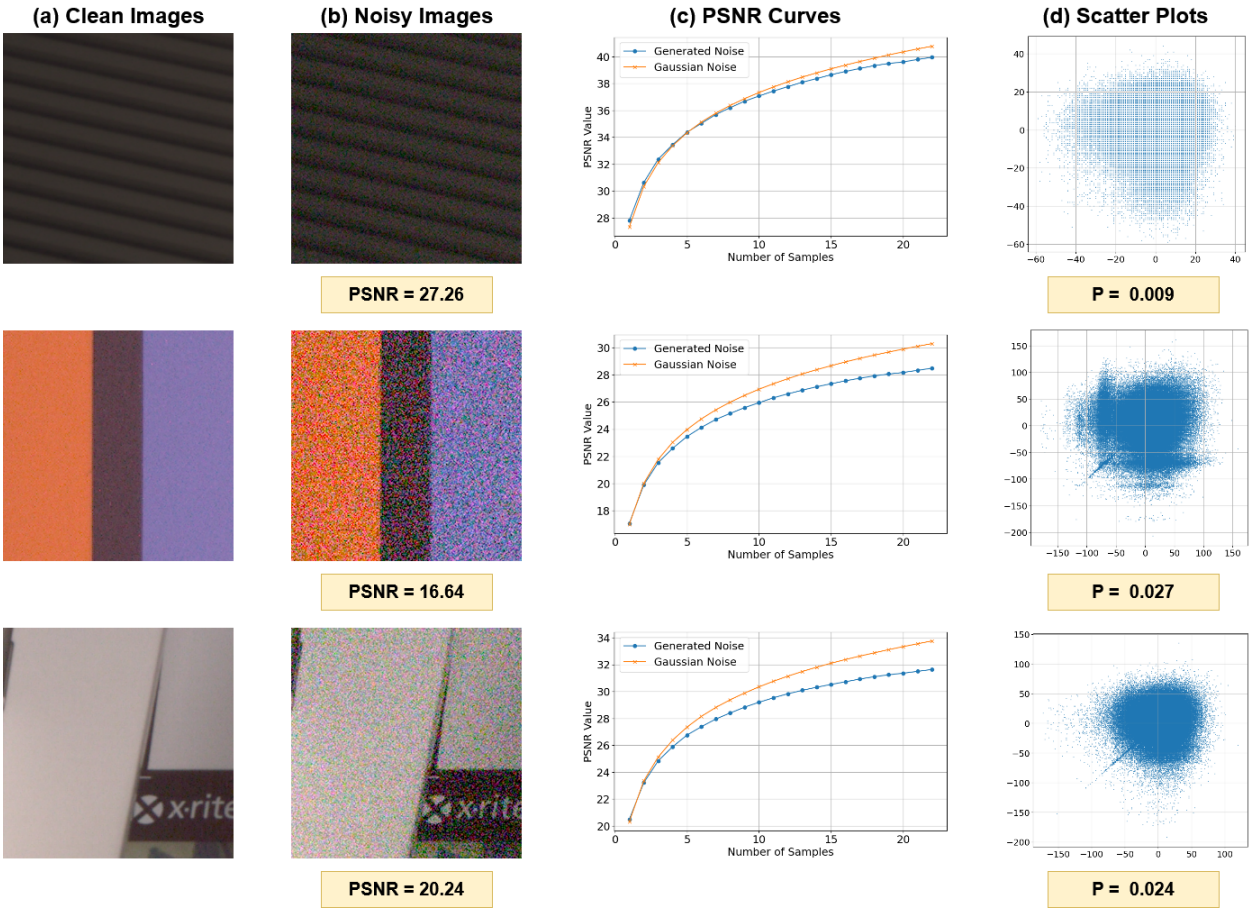

Figure 3: Independence test of the generated noise. (c) shows the change in PSNR of the averaged image as the number of noise samples increases. The curve closely follows the PSNR variation of simulated independent Gaussian noise, asymptotically approaching a theoretical logarithmic curve. (d) presents a scatter plot of the noise distribution between two generated noise images. The approximately circular shape of the distribution indicates that the two noise images exhibit weak correlation. The displayed P values in (d) denote Pearson correlation coefficients.

B Additional Experiments

B.1 Independence Test

To verify that our model can generate random noise rather than learning a fixed noise pattern for each image, we generate multiple noise samples for the same image on the SIDD validation set and analyze the independence among these noise samples. First, according to the law of large numbers, if the noise is independent, the average of multiple noise images should converge to the clean image. We compute the change in PSNR between the averaged image and the clean image as the number of noise samples used for averaging increases. For comparison, we also average independent Gaussian noise added to the clean image. The results are shown in Figure 3(c). It can be observed that the PSNR curve of the averaged generated noise closely aligns with that of the independent Gaussian noise, indicating strong independence in the generated noise.

Secondly, we randomly select two generated noise images and plot a scatter diagram of their pixel values, as shown in Figure 3(d). If the noise were highly correlated, the scatter plot would exhibit a distinct diagonal distribution; whereas if the noise is independent, the scatter plot should approximate a circular distribution. As shown in the figure, the scatter plot is nearly circular, with a few strongly correlated regions observed in

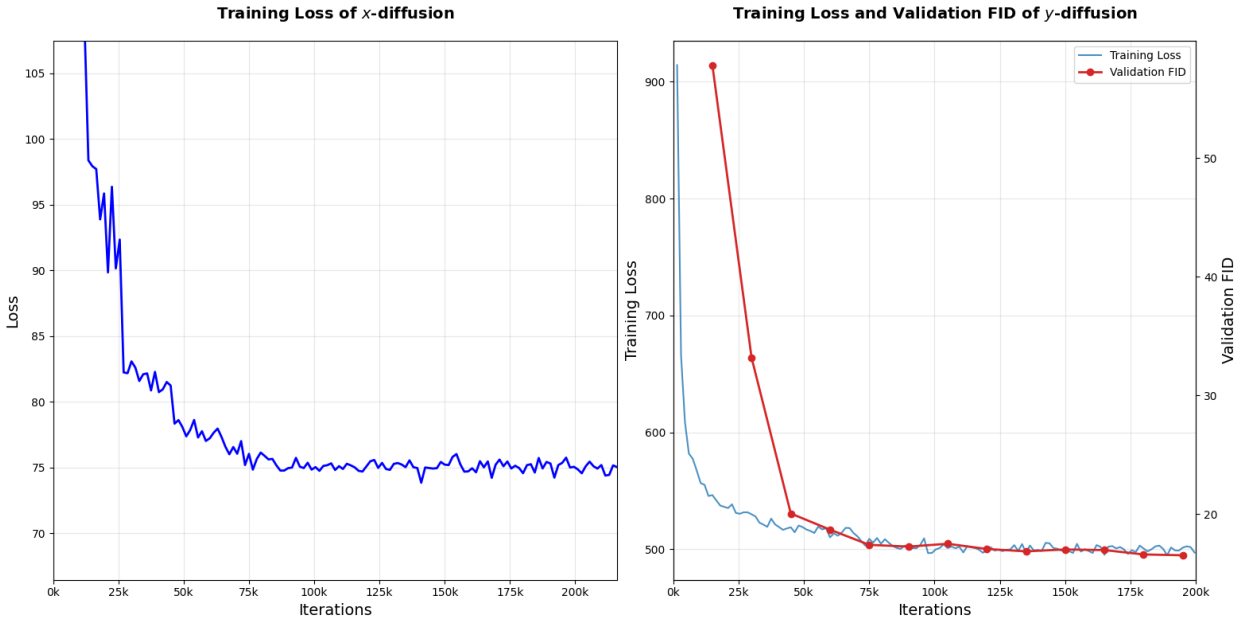

Figure 4: Curves of loss and FID versus the number of iteration steps. (a) shows the decreasing loss curve for the diffusion network regarding x; (b) shows the loss and FID variation curves for the conditional diffusion network regarding y.

high-noise images. This may be attributed to the increased difficulty in distinguishing between signal and noise under high-intensity noise conditions, leading the model to misinterpret certain noise as signal and thereby introducing some correlation. We also calculate the Pearson correlation coefficient between these two noise samples, which is consistently below 0.05, further confirming that the generated noise samples are independent of each other.

B.2 Training and Sampling Hyperparameters

To select the optimal number of iteration steps for training the two diffusion networks, we recorded the variation curves of their losses with increasing iteration steps on the WUP-SIDD data. For the conditional diffusion network, we also tracked the FID variation curve for validation set sampling at every 50 steps. The results are shown in Figure 4. It can be observed that the losses of both models stop decreasing after 100k iterations, and the FID on the validation set for the conditional diffusion model no longer shows significant improvement. Therefore, 100k iteration steps already provide a stable validation trend in this setting. In the main experiments, we use task-dependent training budgets: 100k iterations for simulated image noise and a more conservative 300k iterations for SIDD, as reported in the main text.

To determine the optimal sampling steps and the random term η, we evaluated the generation quality on the SIDD validation set across different sampling steps and different values of η. The results are shown in Figure 5. From the line graph, it can be seen that selecting 200 sampling steps with $\eta = 0.5$ yields the best generation quality. However, when sampling from the training set to generate simulated paired data, we prioritize faster sampling while preserving stochasticity; in the SIDD experiments, we use 50 sampling steps and set $\eta = 1.0$, as stated in the main text.

B.3 Analysis of Degradation Level

We investigate the influence of different effective diffusion noise levels τ_S on noise generation outcomes using the AIM19 and NTIRE20 datasets. For τ_V, we consistently set

$$\tau_V^2 = \tau_S^2 + \hat{\sigma}_n^2,$$

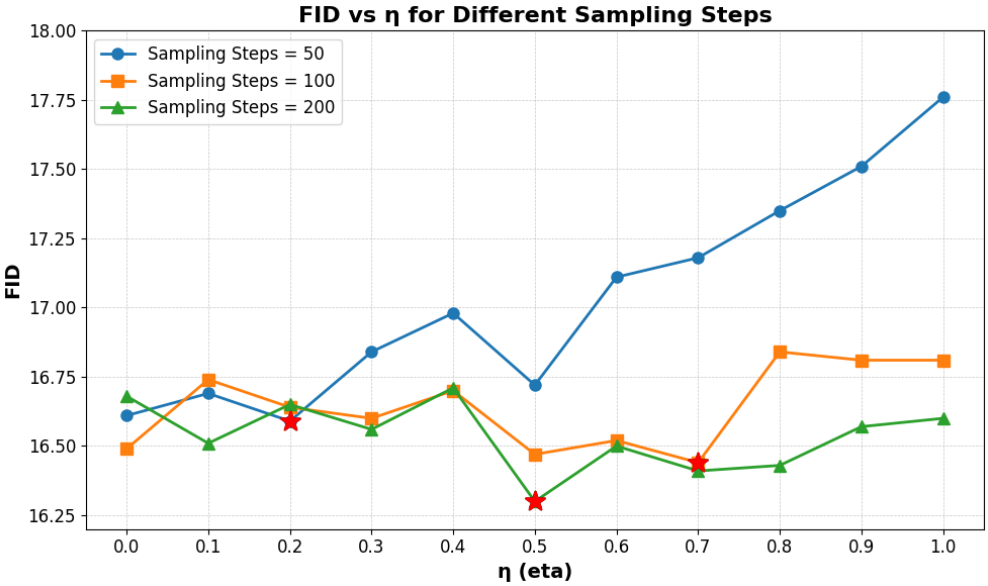

Figure 5: FID variation curves with respect to the number of sampling steps and the random term η.

where $\hat{\sigma}_n$ denotes the average noise standard deviation in the dataset. In this ablation, the labels "c2n" and "n2n" indicate the source of the condition used by the conditional diffusion model. The "c2n" curve uses a clean image as the condition to generate a noisy image, whereas "n2n" first obtains a weak reconstruction from a noisy image and then uses this reconstruction as the condition to generate a noisy image. The results are shown in Figure 6. It can be observed that on both datasets, the FID curve for "n2n" is nearly monotonically increasing. The "n2n" condition is already close to the weak-reconstruction condition used during training when τ_S is small. Increasing τ_S injects additional Gaussian perturbations into this condition and therefore disrupts structural details, leading to worse FID. The "c2n" curves initially decline rapidly and then rise slowly. The decline corresponds to improved alignment between the corrupted clean and noisy states, while the subsequent rise reflects the increasing information loss caused by excessive perturbation. This illustrates the weak-coupling trade-off described in Section 4. Furthermore, note that the τ_S required for the FID to reach its optimal value on AIM19 is significantly larger than that on NTIRE20, which is qualitatively consistent with the dataset-dependent noise statistics in the two test sets.

B.4 Optimal Denoising Steps

Because the diffusion model for x is applied only from the aligned state x_V, the reconstruction distribution $p_{X_0|X_V}$ is sharply concentrated around the clean image for datasets with relatively small noise variance, such as AIM19 and NTIRE20. Therefore, we choose one-step sampling to improve the efficiency of subsequent training for the conditional diffusion model.

For the SIDD dataset, where the noise variance is larger, we investigated the impact of different sampling steps on both the weak reconstruction $p_{X_0|X_V}(\cdot|Y_S)$ and the final generation results. The results are shown in Figure 7. It can be observed that the final generation FID improves rapidly within the first few steps and reaches a near-optimal range around 4–6 steps. Using more steps does not further improve generation quality and may even degrade it. We therefore set the number of sampling steps to 5 as an efficiency-quality trade-off.

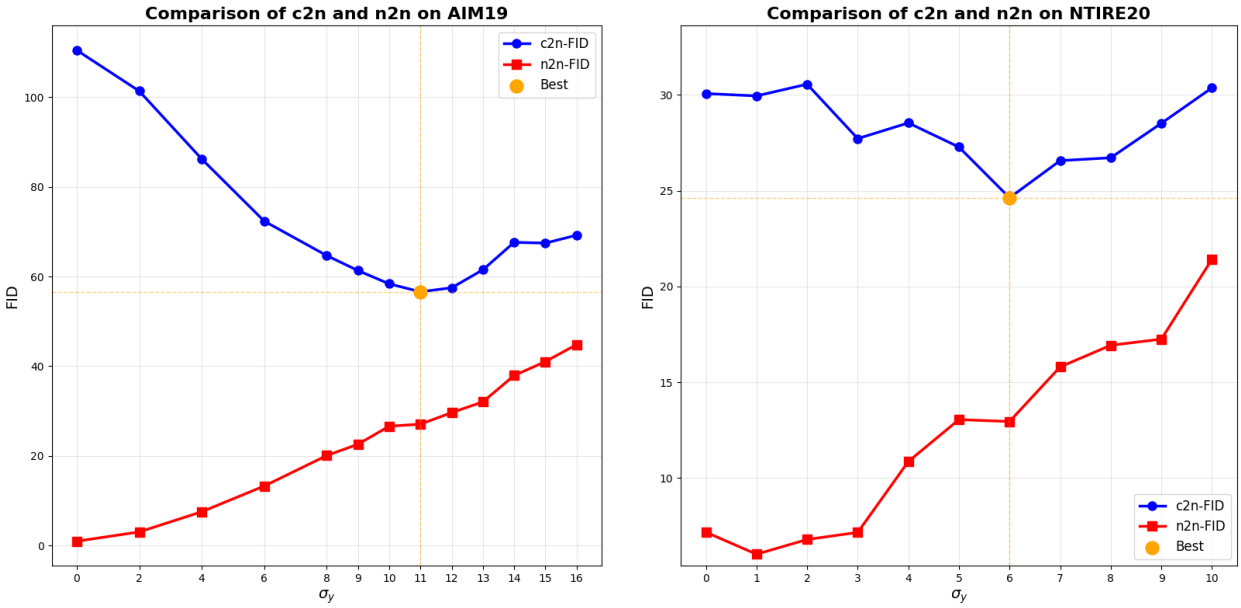

Figure 6: Impact of different τ_S values on the generated results for the AIM19 and NTIRE20 datasets.

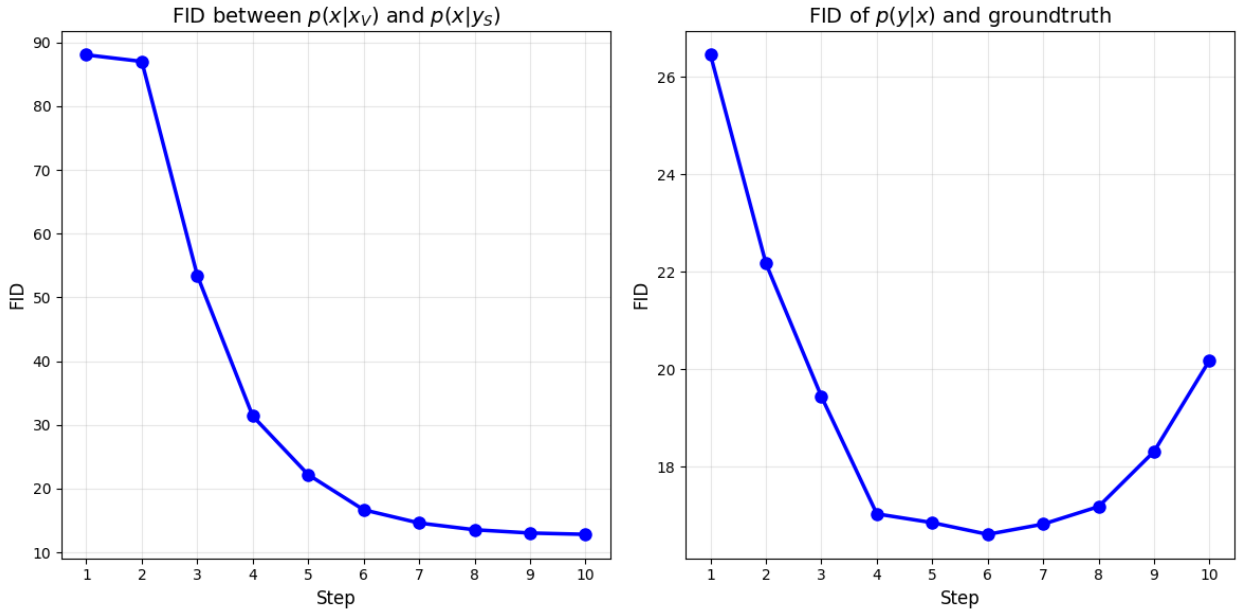

Figure 7: Impact of different sampling steps on the generation results for the SIDD dataset.

B.5 Role of the Two-Path Diffusion Architecture

Note that the clean-domain diffusion model in LUD-DIF can be used by itself to map a noisy observation to a clean reconstruction. Given a noisy observation y_0, one may first diffuse it to y_S, identify the aligned state with x_V through the weak-coupling approximation $Y_S \equiv X_V$, and then apply the reverse clean-domain process $p_{X_0|X_V}(\cdot|y_S)$ to obtain x_0. This direct weak-reconstruction baseline can be summarized as $y_0 \to y_S \equiv x_V \to x_0$, and we refer to it as the noise-to-clean (N2C) baseline in Table 6.

In contrast, the full two-path diffusion architecture uses both the clean-domain diffusion model and the learned conditional diffusion model for $Y|X$. In the clean-to-noisy (C2N) generation mode, we synthesize noisy observations from clean images, and the generated clean–noisy pairs are used to train the same downstream DnCNN denoiser. The comparison in Table 6 evaluates whether explicitly modeling the noisy-domain conditional path provides useful information beyond direct weak reconstruction. The results show that the two-path C2N scheme achieves better downstream denoising performance.

Table 6: Effect of the Two-Path Diffusion Architecture for Noise Modeling

Method	PSNR ↑	SSIM ↑
Direct weak reconstruction (N2C)	32.17	0.847
Two-path C2N (Ours)	**35.46**	**0.896**