Latent Diffusion with Critic–Gate: Score Distillation from Stochastic VAE Posterior

Training Recipe for Minimal-Variance Score Estimation

November 16, 2025

Abstract

We present a fully explicit training recipe for latent diffusion models that distills Tweedie and Conditional Score Expectation Matching (CSEM) signals via a Critic–and–Gate architecture. The VAE encoder is trained with deliberately undertuned KL weighting ($\beta = 0.05$) to emphasize crisp reconstructions while preserving stochastic posteriors. The key innovation is using the full stochastic latent $x_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$ to compute both score targets, enabling the gate to perform automatic geometric feature selection. This approach yields superior few-step sampling by exploiting variance complementarity between manifold-tangent (CSEM) and manifold-normal (Tweedie) directions.

Problem Setup and Notation

Data and VAE. Let $x_{\text{img}} \in \mathbb{R}^{H\times W\times 3}$ be a training image (pixels normalized to $[-1, 1]$). The VAE encoder $\mathcal{E}_\phi$ outputs posterior parameters for the latent variable x_0:

$$\mu_0 = \mu_\phi(x_{\text{img}}) \in \mathbb{R}^{h\times w\times c}, \quad \log \Sigma_0 = \log \Sigma_\phi(x_{\text{img}}) \in \mathbb{R}^{h\times w\times c}$$

with $\Sigma_0 = \text{diag}(\exp(\log \Sigma_0))$. The VAE is pretrained with $\beta = 0.05$ KL weighting (frozen during diffusion training). The decoder $\mathcal{D}_\phi$ is likewise frozen.

OU Diffusion in Latent Space. The forward Ornstein-Uhlenbeck (OU) process in the latent space $y \in \mathbb{R}^{h\times w\times c}$ is defined by:

$$y = e^{-t}x_0 + \sigma_t\epsilon, \quad \epsilon \sim \mathcal{N}(0, I)$$

with the variance schedule $\sigma_t^2 = 1 - e^{-2t}$. $x_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$ is the stochastic initial latent state from the VAE.

Critic–Gate Training Recipe

For each training step (operating on a batch):

1. **Sample VAE posterior**: For an image x_{img} from the batch, get μ_0, Σ_0 and sample the initial latent state:
$$x_0 = \mu_0 + \Sigma_0^{1/2}\epsilon_{\text{vae}}, \quad \epsilon_{\text{vae}} \sim \mathcal{N}(0, I)$$
2. **Sample diffusion**: Sample a time $t \sim \mathcal{U}(0, T)$ and noise $\epsilon \sim \mathcal{N}(0, I)$. Compute the noisy latent state y:
$$y = e^{-t}x_0 + \sigma_t\epsilon$$

3. **Compute Per-Particle Signals (a, b)**:

   - **Tweedie Signal (b)**: This is the standard DSM/Tweedie target.

   $$b = b(x_0, y, t) = -\sigma_t^{-2}(y - e^{-t}x_0)$$

   - **CSEM Signal (a)**: First compute the initial score proxy from the VAE posterior.

   $$s_0(x_0) = \Sigma_0^{-1}(x_0 - \mu_0)$$

   Then transport it using the CSEM identity.

   $$a = a(x_0, t) = e^t s_0(x_0)$$

4. **Compute Blended Target (z_g)**: Pass the noisy state y and time t through the gate network $g(y, t; \psi) \in [0, 1]$.

   $$g_\psi = g(y, t; \psi)$$

   Compute the per-particle blended signal:

   $$z_g = (1 - g_\psi)a + g_\psi b$$

5. **Compute Critic Prediction (q_{pred})**: Pass the same y and t through the critic network $q(y, t; \omega)$.

   $$q_{\text{pred}} = q(y, t; \omega)$$

6. **Update Critic and Gate**: Compute the MSE loss (approximated over the batch):

   $$\mathcal{L} = \mathbb{E}\left[\|q_{\text{pred}} - z_g\|_2^2\right]$$

   Update both network parameters (ω, ψ) by descending this single loss:

   (a) **Update critic**: $\omega \leftarrow \omega - \eta \nabla_\omega \mathcal{L}$
   (b) **Update gate**: $\psi \leftarrow \psi - \eta' \nabla_\psi \mathcal{L}$

Why this works

The $\hat{s}_{\text{CSEM}}$ signal (a) and $\hat{s}_{\text{TWD}}$ signal (b) are complementary.

- $\hat{s}_{\text{CSEM}}$ uses VAE geometry (s_0) and is stable at $t \to 0$ but its variance explodes as $t \to T$. It excels at capturing manifold-tangent features.
- $\hat{s}_{\text{TWD}}$ uses the denoising path $(y - e^{-t}x_0)$ and is stable for large t but its variance explodes as $t \to 0$. It excels at manifold-normal denoising.

The gate $g(y, t; \psi)$ learns to implement the optimal blend $\lambda^*(y, t)$ from the main paper, automatically transitioning from CSEM ($g \to 0$) at small t to Tweedie ($g \to 1$) at large t. The critic $q(y, t; \omega)$ simply learns to predict this optimal blended "teacher" signal. This partitions the task: $\hat{s}_{\text{CSEM}}$ provides geometric structure, $\hat{s}_{\text{TWD}}$ provides denoising, and the critic-gate learns the minimal-variance arbitration between them.

This recipe uses the full stochastic posterior $x_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$, which is **critical**. Using only the mean μ_0 (i.e., $\Sigma_0 = 0$) would collapse the $\hat{s}_{\text{CSEM}}$ signal $s_0(x_0)$ to infinity and break the method. The variance Σ_0 provides the geometric information that CSEM transports.

Scope & Extensions

Conclusion

This recipe provides the missing explicit training procedure for leveraging stochastic VAE posteriors in Critic–and–Gate diffusion training. By **not discarding the covariance**, we preserve the principled task partitioning: CSEM provides manifold-tangent transport, Tweedie provides manifold-normal denoising, and the gate performs variance-optimal arbitration. The result is a data-efficient, few-step sampler that respects VAE-encoded geometry without succumbing to score explosion.