Enhancing Diffusion Model Guidance through Calibration and Regularization

Seyed Alireza Javid
UC San Diego
sajavid@ucsd.edu

Amirhossein Bagheri
Politecnico di Milano
amirhossein.bagheri@mail.polimi.it

Nuria González-Prelcic
UC San Diego
ngprelcic@ucsd.edu

Abstract

Classifier-guided diffusion models have emerged as a powerful approach for conditional image generation, but they suffer from overconfident predictions during early denoising steps, causing the guidance gradient to vanish. This paper presents two complementary contributions to enhance classifier-guided diffusion models. First, we introduce a differentiable Smooth Expected Calibration Error (Smooth ECE) loss that improves classifier calibration with minimal fine-tuning, achieving approximately a 3% improvement in Fréchet Inception Distance (FID) scores. Second, we propose enhanced sampling guidance methods that operate on off-the-shelf classifiers without requiring retraining. Our approach includes: (1) tilted sampling that leverages batch-level information to control outlier influence, (2) adaptive entropy-regularized sampling to maintain diversity, and (3) a novel divergence-regularized sampling method that adds a class-aware, mode-covering correction that strengthens movement toward the target class while maintaining exploration. Theoretical analysis reveals that our methods effectively combine enhanced target direction guidance with controlled diversity exploration, mitigating gradient vanishing. Experimental results on ImageNet demonstrate that our best divergence-guided sampling achieves an FID of 2.13 while maintaining competitive precision and recall metrics. Our methods provide a practical solution for improving conditional generation quality without the computational overhead of classifier and diffusion model retraining. *Code is available at:* `https://github.com/ajavid34/guided-info-diffusion`.

1 Introduction

Denoising diffusion probabilistic models (DDPMs) have achieved state-of-the-art results in unconditional image generation, producing high-quality and diverse images by progressively reversing a noising process Ho et al. [2020], Nichol and Dhariwal [2021], Song et al. [2020a]. These models are built on a solid probabilistic foundation, which allows for stable training and scalability across a wide range of datasets. A key strength of DDPMs is their adaptability: by incorporating external information like class labels or text embeddings, they can be extended to conditional image generation.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Structured Probabilistic Inference & Generative Modeling: Probabilistic Inference in the Era of Large Foundation Models.

Conditional image generation enables the creation of images that adhere to specific constraints, such as class labels or textual descriptions Ramesh et al. [2021], Rombach et al. [2022], Liu et al. [2024]. DDPMs have proven to be highly effective in this area, particularly for generating class-conditioned images of exceptional quality Ho et al. [2020], Dhariwal and Nichol [2021], Nichol and Dhariwal [2021]. The theoretical underpinnings of diffusion models, similar to score-based generative models, allow for this conditional extension through Bayes' theorem without the need for retraining the entire model Song et al. [2023], Song and Ermon [2019], Song et al. [2020b]. One of the most effective techniques for conditional generation involves using an independent, noise-aware classifier to guide the reverse diffusion process Dhariwal and Nichol [2021], Sohl-Dickstein et al. [2015], Ma et al. [2023]. This classifier directs the generation process toward the desired output by estimating the gradient of the log-probability of the target label with respect to the noisy input, all without requiring retraining of the generative model itself.

Despite these advances, a critical limitation of classifier-guided diffusion models is that they suffer from overconfident predictions during early denoising steps, causing the guidance gradient to vanish Ma et al. [2023], Zheng et al. [2022]. Our main contributions to this paper are as follows:

- We propose a calibration-aware classifier finetuning loss based on the Smooth Expected Calibration Error (Smooth ECE) Naeini et al. [2015]. This loss improves calibration and achieves better downstream performance while introducing only minimal fine-tuning overhead.
- We introduce an entropy-regularized guidance method that prevents premature overconfidence, promoting diversity during sampling while preserving class relevance.
- We propose a divergence regularized guidance method, which enhances mode coverage and preserves fidelity to the target class distribution.
- We design a tilted-loss-based batch-aware sampling strategy, which leverages information across generated samples to balance quality and diversity without additional complexity.

We conduct experimental analysis on ImageNet 128×128, demonstrating that our proposed methods outperform existing classifier-guided diffusion approaches in terms of Fréchet inception distance (FID) Heusel et al. [2017], precision, and recall.

2 Classifier design

As introduced by Ma et al. [2023], the ECE (details in Appendix C.1) has a direct connection to the FID. Motivated by this connection, we define a differentiable Huber-like calibration loss termed **Smooth ECE**, which operates over confidence bins and applies a smooth absolute error at the individual sample level. The loss is formulated as

$$\mathcal{L}_{\mathrm{ECE}} = \frac{1}{n} \sum_{b=1}^{B} \sum_{i:\hat{p}^{(i)} \in \mathcal{B}_b} \sqrt{\left(\hat{p}^{(i)} - a^{(i)}\right)^2 + \beta}. \tag{1}$$

where $\hat{p}^{(i)} = \max_y p_\phi(y \mid x^{(i)})$ is the predicted confidence for sample i, and $a^{(i)} = \mathbb{I}[\hat{y}^{(i)} = y^{(i)}]$ is the correctness of the prediction, where $\hat{y}^{(i)} = \arg\max_c p_\phi(c \mid x^{(i)})$. The term $\beta > 0$ is a smoothing constant that ensures differentiability. The interval $\mathcal{B}_b = \left(\frac{b-1}{B}, \frac{b}{B}\right]$ defines the b-th confidence bin, and each sample contributes only to the bin corresponding to its confidence level. The loss is computed per sample and aggregated over all bins to form the final scalar value. The relationship between smooth ECE and Huber loss is explained in Appendix C.1.

Remark 1. Our method differs from Meta-Calibration Bohdal et al. [2023] by directly introducing smoothness into the ECE loss via a Huber-like function, avoiding the need for soft binning or differentiable ranking. Unlike their meta-learning framework, we apply our smooth ECE loss directly during training without outer-loop optimization. This results in a simpler, more efficient, and plug-and-play approach.

3 Sampling guidance

We proposed a calibration-aware classifier design that leverages the ECE. However, fine-tuning classifiers can be costly and time-consuming. In what follows, we focus on off-the-shelf classifier-guidance methods that operate directly on existing checkpoints and require no additional fine-tuning. Since off-the-shelf classifiers are typically not robust to Gaussian noise and are not time-dependent Ma et al. [2023], we select $\widehat{x}_0(t) = \left(\widehat{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta\left(\widehat{x}_t, t\right)\right)/\sqrt{\bar{\alpha}_t}$ as the classifier input.

3.1 Regularized sampling

The sampling guidance from the classifier during the denoising steps of a diffusion model is helpful for increasing both the diversity and quality of the generated image Dhariwal and Nichol [2021]. However, a critical flaw emerges in this process. Even when an image is only partially generated and still lacks fine-grained features, a standard classifier can often classify it with excessively high confidence Zheng et al. [2022]. This causes the classifier's predicted distribution for the noisy image to prematurely converge to a one-hot distribution. As a result, the conditional gradient guidance vanishes early in the process, and the conditional generation degrades into a less effective unconditional one for the remaining steps.

To address this limitation, Zheng et al. [2022] proposed entropy-constrained training and scalar weight based entropy on the classifier guidance. While they show some improvement compared to the baseline, their scheme requires costly training of a classifier from scratch and cannot be generalized to off-the-shelf classifiers. As Figure 1 shows, the top 25% most confident samples maintain near-perfect confidence throughout most of the sampling process, demonstrating the overconfidence problem that leads to less effective guidance. While the average confidence increases more gradually, we can still see the overconfidence problem for most of the denoising process. This motivates our analysis of two general methods for more robust classifier guidance.

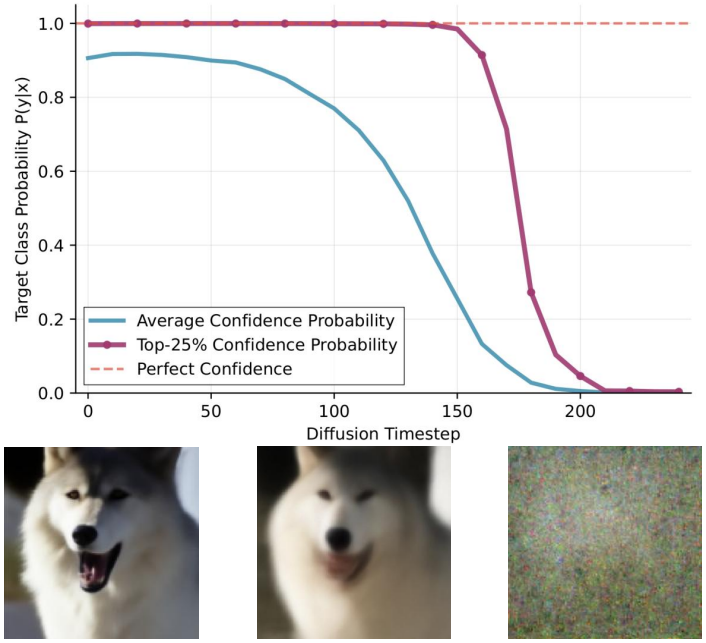


Figure 1: The visualization of the denoising sampling process. Overconfidence in classifier-guided diffusion sampling without regularization is mostly effective in the last steps where the probability is close to one.

3.1.1 Entropy-regularized sampling

Ma et al. [2023] demonstrated that the classifier guidance can be interpreted as the gradient of the joint $E_{\tau_1}(x, y)$ and marginal energy $E_{\tau_2}(x)$ difference as follows:

$$\begin{aligned}\log p_{\tau_1,\tau_2}(y \mid x) &= \log \frac{\exp(\tau_1 f_y(x))}{\sum_{i=1}^{N} \exp(\tau_2 f_i(x))} = \tau_1 f_y(x) - \log \sum_{i=1}^{N} \exp(\tau_2 f_i(x)) \\ &:= -E_{\tau_1}(x, y) + E_{\tau_2}(x),\end{aligned} \tag{2}$$

$$\nabla_x \log p_{\tau_1,\tau_2}(y \mid x) = \nabla_x \left(\tau_1 f_y(x) - \log \left(\sum_{i=1}^{N} \exp(\tau_2 f_i(x)) \right) \right) := -\nabla_x E_{\tau_1}(x, y) + \nabla_x E_{\tau_2}(x), \tag{3}$$

where τ_1 and τ_2 represent joint and marginal temperatures, respectively. Based on the discussion in Section 3.1, we rewrite the regularized guidance as

$$\mathcal{S}_{\text{entropy}}(x, y) := \log p_{\tau_1,\tau_2}(y|x) + \lambda_t H(p(\cdot|x)), \tag{4}$$

where $H(p(\cdot|x)) = -\sum_{i=1}^{N} p(i|x) \log p(i|x)$ is the entropy of the distribution, $p(i|x) = \frac{\exp(f_i(x))}{\sum_{j=1}^{N} \exp(f_j(x))}$, and $\lambda(t) \geq 0$ is the adaptive entropy regularization weight.

Proposition 1. *The gradient of the entropy-regularized score can be written as*

$$\nabla_x \mathcal{S}_{entropy}(x, y) = \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x) \nabla_x f_i(x) \tag{5}$$

$$- \lambda_t \sum_{i=1}^{N} p(i|x)[\log p(i|x) + 1] \left[\nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x) \nabla_x f_j(x) \right]. \tag{6}$$

We observe that the entropy gradient promotes diversity by driving the distribution away from deterministic solutions, which balances exploration in high-confidence regions. Specifically, the entropy term $\nabla_x H(p(\cdot|x))$ contributes a weighted sum of gradients from all classes, where each class's contribution is modulated by $p(i|x)\,[\log p(i|x) + 1]$. This encourages the model to maintain uncertainty with a regularization during sampling, effectively balancing exploitation of the target class with exploration of the probability landscape.

However, a fundamental limitation of this approach is that the entropy regularization is class-agnostic. The sum over the gradient of all classes in the regularization term which is independent of the target class, might overshoot the gradient. This indiscriminate diversity enhancement can lead to suboptimal gradient updates, where the model is encouraged to explore directions that increase entropy but potentially move away from the target class manifold. Consequently, while the method successfully prevents overconfidence, it may compromise the fidelity and class-consistency of the generated images, particularly in later denoising steps where precise class alignment becomes crucial.

3.1.2 Divergence-regularized sampling

Based on the mentioned limitation of the entropy approach, we introduce the f-divergence regularized guidance as

$$\mathcal{S}_{\mathcal{D}}(x, y) := \log p_{\tau_1,\tau_2}(y|x) - \alpha D_f(q_y(\cdot) \| p(\cdot|x)), \tag{7}$$

where $D_f(q\|p) = \sum_i q(i) f\left(\frac{p(i)}{q(i)}\right)$ is an f-divergence with generator function $f : \mathbb{R}_+ \to \mathbb{R}$, $q_y(\cdot)$ is a target distribution (discussed further in Appendix F.3), and $\alpha > 0$ is the divergence weight. We use $q_y(i) = (1 - \epsilon)\frac{1}{N} + \epsilon \mathbb{I}_{i=y}$ with $\epsilon = 0.1$, where the uniform component $(1 - \epsilon)/N > 0$ ensures $q_y(i) > 0$ for all i. This smoothing is essential: if q_y were one-hot, the divergence $D_f(q_y\|p)$ would be infinite unless p is also one-hot (which never occurs during sampling). The uniform component prevents divergence blow-up while ϵ provides target class emphasis. Different choices of f lead to different divergences with distinct theoretical properties and practical implications for guidance.

Assumption 1. The generator function $f : \mathbb{R}_+ \to \mathbb{R}$ satisfies:

1. f is differentiable on $(0, \infty)$,
2. $f(1) = 0$,
3. The target distribution satisfies $q_y(i) > 0$ for all i to ensure finite divergence.

Proposition 2. *The gradient of the f-divergence regularized score is*

$$\begin{aligned}\nabla_x \mathcal{S}_{\mathcal{D}}(x, y) = \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x)\, \nabla_x f_i(x) \\ - \alpha \sum_{i=1}^{N} w_f(q_y(i), p(i|x))\, g_i(x),\end{aligned} \tag{8}$$

where $p(i|x) = \frac{\exp(f_i(x))}{\sum_j \exp(f_j(x))}$, $g_i(x) = \nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x)\, \nabla_x f_j(x)$, $p_{\tau_2}(i|x) = \frac{\exp(\tau_2 f_i(x))}{\sum_j \exp(\tau_2 f_j(x))}$ *and* $w_f(q, p) = p\, f'\left(\frac{p}{q}\right)$.

We now examine three specific f-divergences, each offering distinct theoretical properties and practical trade-offs for diffusion guidance.

3.1.3 Reverse KL Divergence: Mode-Covering Guidance

For reverse Kullback-Leibler divergence with $f(t) = -\log(t)$, we have $f'(t) = -1/t$, which gives $w_f(q, p) = -q$. This yields a particularly interpretable gradient form.

Corollary 1 (Reverse KL gradient)**.** *The gradient of the reverse KL divergence regularized score is:*

$$\begin{aligned}\nabla_x \mathcal{S}_{RKL}(x, y) =& \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x) \nabla_x f_i(x) \\ &+ \alpha \sum_{i=1}^{N} q_y(i) \left[\nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x) \nabla_x f_j(x) \right].\end{aligned} \tag{9}$$

Reverse KL divergence exhibits two fundamental properties that are particularly advantageous for our guidance objective [Polyanskiy and Wu, 2025, Bishop and Nasrabadi, 2006]. First, its *mode-covering* behavior ensures that the model distribution $p(\cdot|x)$ maintains non-zero probability mass wherever the target distribution $q_y(\cdot)$ has support, effectively preventing the exclusion of any viable modes during sampling. Second, its *zero-avoiding* characteristic imposes a severe penalty when $p(\cdot|x) \to 0$ while $q_y(\cdot) \geq 0$, which mathematically manifests as the divergence approaching infinity. This asymmetric penalty structure is crucial for diversity preservation. In the context of diffusion guidance, these properties ensure that the generated samples maintain coverage over all classes in the target distribution.

To gain deeper insight into the guidance mechanism, we decompose the reverse KL gradient into interpretable components.

Lemma 1 (Reverse KL decomposition)**.** *The negative of reverse KL gradient can be decomposed as:*

$$-\nabla_x D_{KL}(q_y \| p(\cdot|x)) = \underbrace{\sum_{i=1}^{N} q_y(i) \nabla_x f_i(x)}_{\textit{Target direction}} - \underbrace{\sum_{j=1}^{N} p(j|x) \nabla_x f_j(x)}_{\textit{Current direction}}. \tag{10}$$

This decomposition reveals that the reverse KL regularization pulls the sample toward the target class distribution q_y while simultaneously pushing away from the current prediction $p(\cdot|x)$. This dual mechanism prevents premature convergence to overconfident predictions.

To understand the guidance direction more concretely, we conduct a closed-form analysis in mixed-Gaussian scenarios, which provides insight into how the method operates on structured data manifolds.

Proposition 3 (Gaussian mixture analysis)**.** *Let $X \sim \mathcal{P}$ be a random variable defined on $\mathbb{R}^d$, with density function $f(x) = \sum_{k=1}^{K} b_k f_k(x)$, where $f_k(x)$ is a normal density with mean μ_k and covariance matrix Σ_k, and $b_k > 0$ with $\sum_{k=1}^{K} b_k = 1$. Define the posterior as $w_k(x) := p(k|x) = \frac{b_k f_k(x)}{\sum_{j=1}^{K} b_j f_j(x)}$. Then:*

$$\nabla_x \mathcal{S}_{RKL}(x, y) = \sum_{k=1}^{K} f_k(x) \Sigma_k^{-1} (\mu_k - x)\, \Gamma_k(x, y, \alpha, \epsilon), \tag{11}$$

where the weight function is:

$$\Gamma_k(x, y, \alpha, \epsilon) = \tau_1 \mathbb{I}_{k=y} - \tau_2\, p_{\tau_2}(k|x) + \alpha\big(q_y(k) - w_k(x)\big), \tag{12}$$

and $w_k(x)$ is the posterior distribution. Here $p_{\tau_2}(k|x)$ is the tempered posterior over components, e.g. $p_{\tau_2}(k|x) \propto \exp(\tau_2\, \ell_k(x))$ with $\ell_k(x) = \log b_k + \log f_k(x)$, and ϵ enters only through q_y when you choose a smoothed target.

Remark 2. When all covariances are identity matrices ($\Sigma_k = \sigma^2 I$) and the target distribution follows $q_y(i) = (1 - \epsilon)\frac{1}{N} + \epsilon \mathbb{I}_{i=y}$, the guidance is proportional to:

$$\frac{1}{\sigma^2}\left[\underbrace{(\tau_1 + \alpha\epsilon)(\mu_y - x)}_{\text{Enhanced target direction}} + \alpha\frac{1-\epsilon}{K}\underbrace{\sum_{k \neq y}(\mu_k - x)}_{\text{Diversity directions}} - \underbrace{\sum_{k=1}^{K}(\tau_2 p_{\tau_2}(k|x) + \alpha p(k|x))(\mu_k - x)}_{\text{Current distribution pull}}\right]. \tag{13}$$

The first term pulls toward the target class center with amplified strength, the second term uniformly pulls toward all non-target class centers to maintain mode coverage, and the last term acts as an adaptive correction that prevents overshooting and maintains stability.

Figure 2 demonstrates that throughout the entire sampling process, the gradient maps maintain significant activity across multiple regions, never collapsing to a single point. This confirms the mode-covering property of reverse KL divergence. Additional visualizations are provided in Appendix F.

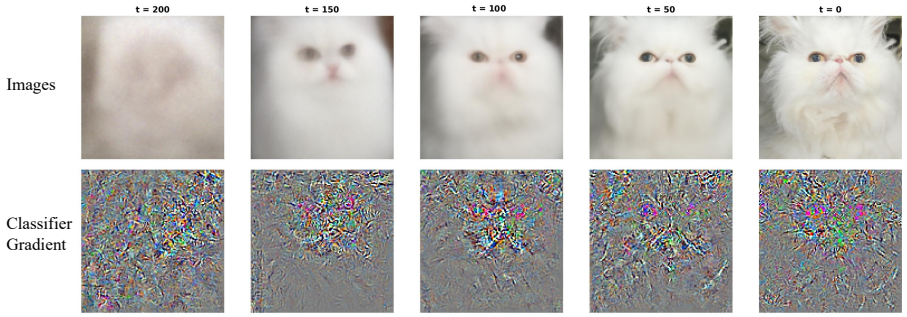


Figure 2: The visualization of intermediate sampling pictures and classifier gradient figures.

3.1.4 Forward KL Divergence: Mode-Seeking Guidance

In contrast to reverse KL, forward KL divergence with $f(t) = t\log(t)$ exhibits mode-seeking behavior, penalizing $p(\cdot|x)$ for placing mass where $q_y(\cdot)$ has little support. For this divergence, we have $w_f(q, p) = p[\log(p/q) + 1]$.

Corollary 2 (Forward KL gradient)**.** *The gradient of the forward KL divergence regularized score is:*

$$\begin{aligned}\nabla_x S_{FKL}(x,y) = \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x) \nabla_x f_i(x) \\ - \alpha \sum_{i=1}^{N} p(i|x) \left[\log \frac{p(i|x)}{q_y(i)} + 1 \right] \\ \times \left[\nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x) \nabla_x f_j(x) \right].\end{aligned}$$

This formulation prioritizes precision over coverage. When q_y is concentrated on a single mode, forward KL strongly penalizes $p(\cdot|x)$ for exploring other modes, potentially producing sharper but less diverse samples compared to reverse KL. The logarithmic weighting term $\log(q_y(i)/p(i|x))$ becomes increasingly negative for modes where $p(\cdot|x)$ exceeds q_y, creating a strong repulsive force away from non-target regions. The numerical results also confirm this in Section 4 as this method has the highest precision and lowest recall.

3.1.5 Jensen-Shannon Divergence: Balanced Guidance

Jensen-Shannon divergence offers a symmetric alternative that balances mode-seeking and mode-covering behaviors through an implicit mixture distribution $m = \frac{1}{2}(q_y + p(\cdot|x))$. It can be expressed as:

$$D_{\mathrm{JS}}(q_y \| p(\cdot|x)) = \frac{1}{2} D_{\mathrm{KL}}(q_y \| m) + \frac{1}{2} D_{\mathrm{KL}}(p(\cdot|x) \| m), \tag{14}$$

where $m(i) = \frac{1}{2}(q_y(i) + p(i|x))$.

Corollary 3 (Jensen-Shannon gradient)**.** *The gradient of the Jensen-Shannon divergence regularized score is:*

$$\begin{aligned}\nabla_x S_{JS}(x,y) = \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x) \nabla_x f_i(x) \\ - \alpha \sum_{i=1}^{N} p(i|x) \log \frac{2p(i|x)}{q_y(i) + p(i|x)} \\ \times \left[\nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x) \nabla_x f_j(x) \right].\end{aligned}$$

where $g_i(x) = \nabla_x f_i(x) - \sum_j p(j|x) \nabla_x f_j(x)$.

The symmetric nature of JS divergence provides several advantages. Unlike forward KL (which heavily penalizes p for spreading beyond q_y) and reverse KL (which heavily penalizes p for missing modes in q_y), JS divergence applies moderate penalties in both directions through the mixture m. The weighting term $(q_y(i) - p(i|x))/m(i)$ is bounded and becomes small when $q_y(i) \approx p(i|x)$, providing smooth gradient dynamics. This may offer a favorable trade-off between maintaining target class fidelity and preserving sample diversity, as we verify empirically in Section 4.

The overall algorithm for f-divergence guided sampling is provided in Algorithm 1.

3.2 Tilted sampling

Since we are using batch sampling, we can leverage the information across the samples generated in each batch. This batch-aware sampling could lead to better adjustments for outlier-generated images. However, we do not want to add extra complexity to the sampling

Algorithm 1 DDPM f-Divergence Guided Sampling

Require: Diffusion model $\mathcal{D}_\theta$, classifier f, class label y, temperatures τ_1, τ_2, divergence weight α, target bias ϵ, target distribution q_y, classifier guidance scale γ_t, divergence type $\mathcal{D}$.

$\hat{x}_T \sim \mathcal{N}(0, I)$
for $t = T, \ldots, 1$ **do**
  $\mu, \epsilon_\theta(\hat{x}_t, y, t) \leftarrow \mathcal{D}_\theta(\hat{x}_t, y, t)$
  $\hat{x}_0(t) \leftarrow (\hat{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\hat{x}_t, y, t))/\sqrt{\bar{\alpha}_t}$ ▷ Predicted clean sample
  $p \leftarrow \text{softmax}(f(\hat{x}_0(t)))$ ▷ Current distribution
  Compute $g \leftarrow \nabla_{\hat{x}_0(t)} \mathcal{S}_\mathcal{D}(x, y)$
  $\hat{x}_{t-1} \sim \mathcal{N}(\mu + \gamma_t g, \sigma_t)$
end for
return $\hat{x}_0$

process. We utilize the loss introduced in Li et al. [2023] for coefficient adjustment in our sampling process:

$$\mathcal{S}_{tilted}(t; x, y) := \frac{1}{t} \log \left(\frac{1}{N} \sum_{i \in [N]} e^{t \, \log p_{\tau_1, \tau_2}(y|x)} \right). \tag{15}$$

where y is the target class we want to generate. Following Li et al. [2023], the derivative will have a coefficient which is a distribution over samples. The new derivative is $g^i_{new} = w^i(t; \theta) \, \nabla_{x_t} \log p_{\tau_1, \tau_2}(y|x)|_{x_t = \mu}$ (details in the Appendix C.2). This formulation allows us to control the diversity of generated samples by tuning the parameter t. When t is large and positive, the process gives more weight to high-probability samples and pays less attention to outliers, which improves sample quality but reduces diversity. In contrast, when t is negative, the process emphasizes low-probability or outlier samples, which increases diversity but may reduce overall sample quality. In this way, the hyperparameter t provides a direct trade-off between quality and diversity.

4 Experiments

Smooth ECE: We evaluate the effectiveness of the Smooth ECE regularizer (for details see Appendix E.1) when applied during fine-tuning. As shown in Table 1, incorporating the Smooth ECE regularization leads to a reduction of 0.2 in the FID score, corresponding to a relative improvement of approximately 3%. These results show that the Smooth ECE regularizer not only improves the quality of generated samples but could also be useful as a regularization method during the main training phase (to be investigated in the future work). Moreover, the observed increase in precision indicates a positive shift towards more class-consistent samples, while recall remains competitive, showing that the regularization does not excessively constrain diversity.

Table 1: Comparison of the baseline DDPM, classifier-guided diffusion, and our Smooth ECE-guided diffusion. All models are sampled for 250 DDPM steps, and we generate 10k ImageNet 128×128 samples for evaluation. Lower is better for FID; higher is better for Precision and Recall. FID values are averaged over three runs.

Method	**Classifier**	**FID** $\downarrow$	**Precision** $\uparrow$	**Recall** $\uparrow$
Dhariwal and Nichol [2021]	Basic fine-tuned	6.15	0.77	**0.68**
Diffusion + ECE (ours)	ECE fine-tuned	**5.94**	**0.79**	0.66

Sampling Guidance: We explore three different sampling strategies to improve inference: (1) *adaptive entropy-regularized sampling*; and (2) *divergence-regularized sampling*; (3) *tilted sampling*, which incorporates batch-level information using a tilted loss. As shown in Table 2, all three approaches outperform the baseline guided by ResNet-50 Ma et al. [2023]. Tilted sampling (with $t = -0.2$) slightly reduces the FID from 5.335 to 5.281, demonstrating

the value of leveraging batch-level guidance. Entropy-regularized sampling also leads to modest improvements in both FID and recall. As we anticipated, this method has the best recall value by incorporating the class-agnostic regularization. The best performance comes from divergence-guided sampling, which achieves the lowest FID score of 5.118. The divergence-regularized method attains the lowest FID while maintaining strong recall and precision, indicating that balancing class consistency with exploration enhances both quality and coverage.

Table 2: The comparison of the baseline DDPM diffusion and our regularized guided sampling. All models are sampled for 250 DDPM steps. We generate 10k ImageNet 128×128 samples for evaluation. FID values are averaged over three runs.

Method	**Classifier**	**FID** $\downarrow$	**Precision** $\uparrow$	**Recall** $\uparrow$
Ma et al. [2023]	ResNet-50	5.34	**0.78**	0.67
Diffusion tilted guided ($t = -0.2$) (ours)	ResNet-50	5.28	0.77	0.68
Diffusion adaptive entropy guided (ours)	ResNet-50	5.30	0.77	**0.69**
RKL guided (ours)	ResNet-50	**5.12**	**0.78**	0.68

Lastly, Table 3 benchmarks our best-performing approaches against representative state-of-the-art methods on ImageNet 128×128 with the matched sampling steps, without retraining. All three f-divergence methods substantially outperform the baseline [Ma et al., 2023], with

Table 3: The comparison of the baseline DDPM diffusion and our divergence guided samplings. All models are sampled for 250 DDPM steps. We generate 50k ImageNet 128×128 samples for evaluation.

Method	**Classifier**	**FID** $\downarrow$	**Precision** $\uparrow$	**Recall** $\uparrow$
Dhariwal and Nichol [2021]	Basic fine-tuned	2.97	0.78	0.59
Zheng et al. [2022]	Entropy aware classifier	2.68	0.80	0.56
Ho and Salimans [2022]	-	2.43	-	-
Ma et al. [2023]	ResNet-50	2.37	0.77	0.60
FKL guided (ours)	ResNet-50	2.33	0.78	0.61
RKL guided (ours)	ResNet-50	2.29	0.78	0.61
JS guided (ours)	ResNet-50	2.27	0.77	**0.62**
Ma et al. [2023]	ResNet-101	2.19	0.79	0.58
FKL guided (ours)	ResNet-101	2.17	**0.80**	0.59
RKL guided (ours)	ResNet-101	2.14	0.79	0.59
JS guided (ours)	ResNet-101	**2.13**	0.79	0.60

Jensen-Shannon divergence achieving the best performance (FID of 2.27 with ResNet-50 and **2.13 with ResNet-101**), establishing a new result without any need to retrain the classifier or diffusion model.

The consistent ordering JS > RKL > FKL > Baseline can be explained by examining how each divergence balances precision and recall. Forward KL's mode-seeking behavior strongly penalizes $p(\cdot|x)$ for placing mass outside q_y's support, forcing the model to concentrate on high-confidence regions. While this yields the highest precision (0.80 with ResNet-101), it comes at the cost of the lowest recall (0.59), confirming that aggressive mode-seeking sacrifices diversity. The logarithmic weighting term $\log(q_y(i)/p(i|x))$ in Equation (2) creates strong repulsive forces that prematurely collapse the sampling distribution.

Reverse KL exhibits the opposite behavior: its mode-covering property (Lemma 1) maintains broader support, yielding better recall (0.59) while slightly sacrificing precision (0.79). However, the linear weighting $q_y(i)$ in Equation (9) can lead to overshooting when q_y is diffuse, as the regularization pulls toward *all* classes weighted by q_y rather than adaptively moderating based on current prediction quality.

Jensen-Shannon achieves the best balance through its mixture distribution $m(i) = \frac{1}{2}(q_y(i) + p(i|x))$. The symmetric weighting $(q_y(i) - p(i|x))/m(i)$ in Equation (15) provides *adaptive* corrections that strengthen when p deviates from q_y but moderate when they align. This

dynamic penalty structure achieves the highest recall (0.60) while maintaining competitive precision (0.79), confirming that JS's balanced approach to mode-seeking and mode-covering is optimal for diffusion guidance. The mixture m acts as a stabilizing reference that prevents both premature collapse (FKL's failure mode) and excessive exploration (RKL's tendency). ResNet-101 also improves our best result from 2.27 to 2.13 FID, while maintaining consistent relative improvements over the baseline (4-8% FID reduction).

5 Conclusion

This work addresses critical limitations in classifier-guided diffusion through calibration-aware design and regularized sampling guidance. Our key contributions include a differentiable Smooth ECE loss achieving 3% FID improvement with minimal overhead, three sampling methods with divergence guidance achieving the best FID of 2.13, and theoretical analysis providing mathematical grounding for observed behaviors. This work demonstrates that principled f-divergence regularization can substantially improve classifier-guided diffusion without model retraining, achieving FID of 2.13 on ImageNet 128×128. Our theoretical framework reveals how different divergence choices (RKL, FKL, JS) trade off mode coverage and precision, with Jensen-Shannon's balanced penalties proving empirically optimal. These contributions offer practical, plug-and-play enhancements for conditional generation in deployed systems where diffusion models cannot be retrained with more complex approaches.

References

S Mussafer Ali and Samuel D Silvey. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society: Series B (Methodological)*, 28(1):131–142, 1966.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223, 2017.

Armin Behnamnia, Gholamali Aminian, Alireza Aghaei, Chengchun Shi, Vincent YF Tan, and Hamid R Rabiee. Log-sum-exponential estimator for off-policy evaluation and learning. In *Forty-second International Conference on Machine Learning*.

Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

Ondrej Bohdal, Yongxin Yang, and Timothy Hospedales. Meta-calibration: Learning of model calibration using differentiable expected calibration error. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL `https://openreview.net/forum?id=R2hUure38l`.

Giuseppe C Calafiore, Stephane Gaubert, and Corrado Possieri. Log-sum-exp neural networks and posynomial models for convex and log-log-convex data. *IEEE transactions on neural networks and learning systems*, 31(3):827–838, 2019.

Imre Csiszár. On information-type measure of difference of probability distributions and indirect observations. *Studia Sci. Math. Hungar.*, 2:299–318, 1967.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Adji Bousso Dieng, Dustin Tran, Rajesh Ranganath, John Paisley, and David Blei. Variational inference via χ upper bound minimization. *Advances in Neural Information Processing Systems*, 30, 2017.

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

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

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

Peter J Huber. Robust statistics. In *International encyclopedia of statistical science*, pages 1248–1251. Springer, 2011.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2): 183–233, 1999.

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

Bahjat Kawar, Roy Ganz, and Michael Elad. Enhancing diffusion-based image synthesis with robust classifier guidance. *arXiv preprint arXiv:2208.08664*, 2022.

Diederik Kingma and Ruiqi Gao. Understanding diffusion objectives as the elbo with simple data augmentation. *Advances in Neural Information Processing Systems*, 36:65484–65516, 2023.

Jongmin Lee, Wonseok Jeon, Byung-Jun Lee, Joelle Pineau, and Kee-Eung Kim. Optidice: Offline policy optimization via stationary distribution correction estimation. In *International Conference on Machine Learning*, pages 6120–6130, 2021.

Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. Tilted empirical risk minimization. *arXiv preprint arXiv:2007.01162*, 2020.

Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. On tilted losses in machine learning: Theory and applications. *Journal of Machine Learning Research*, 24(142):1–79, 2023. URL `http://jmlr.org/papers/v24/21-1095.html`.

Yingzhen Li and Richard E Turner. Rényi divergence variational inference. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024.

Jiajun Ma, Tianyang Hu, Wenjia Wang, and Jiacheng Sun. Elucidating the design space of classifier-guided diffusion generation. *arXiv preprint arXiv:2310.11311*, 2023.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Ofir Nachum, Mohammad Norouzi, Kelvin Xu, and Dale Schuurmans. Trust-pcl: An off-policy trust region method for continuous control. In *International Conference on Learning Representations*, 2017.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29, 2015.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

Yury Polyanskiy and Yihong Wu. *Information theory: From coding to learning*. Cambridge university press, 2025.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

Igal Sason and Sergio Verdú. f-divergence inequalities. *IEEE Transactions on Information Theory*, 62(11):5973–6006, 2016.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

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

Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum likelihood training of score-based diffusion models. In *Advances in Neural Information Processing Systems*, volume 34, pages 1415–1428, 2021.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202, pages 31999–32015, 2023.

Christopher KI Williams and David Barber. Bayesian classification with gaussian processes. *IEEE Transactions on pattern analysis and machine intelligence*, 20(12):1342–1351, 1998.

Yilun Xu, Gabriele Corso, Tommi Jaakkola, Arash Vahdat, and Karsten Kreis. Disco-diff: Enhancing continuous diffusion models with discrete latents. *arXiv preprint arXiv:2407.03300*, 2024.

Guangcong Zheng, Shengming Li, Hui Wang, Taiping Yao, Yang Chen, Shouhong Ding, and Xi Li. Entropy-driven sampling and training scheme for conditional diffusion generation. In *European Conference on Computer Vision*, pages 754–769. Springer, 2022.

A Related Works

Other improved guidance approaches: Zheng et al. [2022] study the vanishing-guidance problem in classifier-guided DDPMs and propose an entropy-based scheme that weights the classifier gradient by a function of predictive entropy during sampling. This promotes exploration and helps preserve structural fidelity, yielding empirical gains in conditional generation. However, the approach leaves the guidance direction unchanged and is not supported by a general theoretical analysis for arbitrary classifiers. As we discussed in Section 3.1, the absence of theoretical analysis and the limited effectiveness of this method for general classifiers motivated our work, which directly regularizes the sampling objective. Kawar et al. [2022] proposed to harness the perceptually aligned gradients phenomenon by utilizing robust classifiers to guide a diffusion process, while this method showed a minor improvement compared to Dhariwal and Nichol [2021], the limited contribution could not surpass our second baseline Ma et al. [2023].

Classifier-free guidance: Rather than relying on an external classifier, Ho and Salimans [2022] proposed classifier-free guidance in which they train a single denoiser to predict diffusion noise both *with* a condition and *without* it. The gap between these two predictions approximates the conditional score, which can be used to steer sampling. In practice, guidance is applied by $\epsilon^*_\theta(x_t, y, t) = \epsilon_\theta(x_t, y, t) + (s-1)\Big[\epsilon_\theta(x_t, y, t) - \epsilon_\theta(x_t, \varnothing, t)\Big]$, where $s \geq 1$ is the guidance scale and $\epsilon_\theta(x_t, \varnothing, t)$ denotes the unconditional prediction learned by randomly dropping the condition during training. This simple setup avoids an auxiliary classifier and typically improves on-condition fidelity in generated samples.

Beyond classifier guidance: The diffusion modeling landscape has seen significant advances across multiple fronts. Score-based generative models Song et al. [2020b] provide an alternative mathematical framework using stochastic differential equations (SDEs), enabling continuous-time formulations and advanced sampling techniques like probability flow ordinary differential equation (ODE). Denoising diffusion implicit models (DDIM) Song et al. [2020a] introduced deterministic sampling that dramatically reduces the number of denoising steps required. Latent diffusion models Rombach et al. [2022] operate in compressed latent spaces, significantly reducing computational costs while maintaining generation quality. EDM Karras et al. [2022] unified various diffusion formulations and introduced improved training and sampling strategies. Consistency models Song et al. [2023] enable single-step generation through distillation from pre-trained diffusion models. DiT Peebles and Xie [2023] replaced U-Net architectures with transformers, achieving state-of-the-art results on class-conditional ImageNet generation. More recent diffusion approaches have reported improved FID scores by reformulating diffusion objectives or augmenting the latent space. Kingma and Gao [2023] reinterpret common diffusion losses as ELBO variants with monotonic weighting, achieving state-of-the-art FID on ImageNet. Xu et al. [2024] propose DisCo-Diff, which introduces discrete latents to simplify the noise-to-data mapping, further improving image quality. While effective, both methods require retraining diffusion models from scratch and often rely on longer sampling chains (nearly doubling the diffusion steps), making them more computational demanding in practice.

Tilted losses and robust optimization: Inspired by the log-sum-exponential operator with applications in multinomial linear regression, naive Bayes classifiers, tilted empirical risk, and log-sum-exponential off-policy estimator, [Calafiore et al., 2019, Murphy, 2012, Williams and Barber, 1998, Li et al., 2020, Behnamnia et al.], we propose the tilted sampling. The tilted loss framework introduced by Li et al. [2020, 2023] provides a principled approach for controlling the influence of outliers through a temperature parameter t, enabling weighted gradient computation where sample contributions are modulated by their relative importance. This framework extends beyond simple outlier control, offering connections to distributionally robust optimization (DRO) and risk-sensitive learning. In the context of machine learning, tilted losses have been successfully applied to federated learning for handling heterogeneous data distributions, reinforcement learning for risk-aware policy optimization, and meta-learning for improved generalization across tasks. The temperature parameter t in tilted losses creates a spectrum between average-case ($t \to 0$) and worst-case ($t \to \infty$) optimization, with negative values emphasizing easy examples—a property we exploit in our diffusion guidance to down-weight overconfident classifier predictions.

f-divergences: f-divergences Csiszár [1967], Ali and Silvey [1966] provide a unified framework for comparing probability distributions, encompassing KL divergence, Hellinger distance, and total variation as special cases Sason and Verdú [2016]. In generative modeling, f-GANs Nowozin et al. [2016] showed that different GAN formulations minimize different f-divergences, with forward KL causing mode-dropping and reverse KL encouraging mode-covering Arjovsky et al. [2017]—insights directly motivating our guidance design.

Variational inference has explored f-divergence alternatives beyond forward KL Jordan et al. [1999], including Rényi Li and Turner [2016] and χ divergences Dieng et al. [2017], demonstrating that divergence choice affects underfitting-overfitting trade-offs. Similar trade-offs appear in RL policy optimization Nachum et al. [2017], Lee et al. [2021], where divergence selection impacts exploration-exploitation balance.

In diffusion models, prior work analyzed f-divergences for *training objectives* Song et al. [2021], Kingma and Gao [2023]. Our work is the first systematic exploration of f-divergences for

classifier-guided sampling, where divergence acts as a regularizer rather than training objective. We provide: (1) rigorous gradient derivations (Proposition 2), (2) closed-form Gaussian analysis (Proposition 3), and (3) empirical demonstration that Jensen-Shannon divergence achieves superior precision-recall balance. The finding that symmetric JS outperforms both forward and reverse KL challenges conventional wisdom that mode-covering (reverse KL) is universally optimal for generation, suggesting balanced penalties are preferable when diversity and fidelity are equally valued.

B Background

In this section, we provide a brief overview of the formulation of Gaussian diffusion models as discussed in Ho et al. [2020] and the concept of classifier guidance referenced in Dhariwal and Nichol [2021].

B.1 Gaussian diffusion models

Diffusion models consist of a series of time-dependent components that implement both forward and reverse processes. We define data distribution as $x_0 \sim q(x_0)$ and a Markovian noising process q which gradually adds noise to the data to produce noised samples $\{x_t\}_{t=1}^T$. This process is defined as the forward process, and for a given forward variance β_t, is defined as

$$q\left(x_t \mid x_{t-1}\right) := \mathcal{N}\left(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t \mathbf{I}\right). \tag{16}$$

Using $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=0}^{t} \alpha_s$ this can also be written as

$$q\left(x_t \mid x_0\right) = \mathcal{N}\left(x_t; \bar{\alpha}_t x_0, \left(1-\bar{\alpha}_t\right)\right). \tag{17}$$

The Bayes' theorem will show that the posterior $q\left(x_{t-1} \mid x_t, x_0\right)$ has a Gaussian distribution with the following mean and variance:

$$\tilde{\mu}_t\left(x_t, x_0\right) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} x_0 + \frac{\sqrt{\alpha_t}\left(1-\bar{\alpha}_{t-1}\right)}{1-\bar{\alpha}_t} x_t, \tag{18}$$

$$\tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t, \tag{19}$$

$$q\left(x_{t-1} \mid x_t, x_0\right) = \mathcal{N}\left(x_{t-1}; \tilde{\mu}\left(x_t, x_0\right), \tilde{\beta}_t \mathbf{I}\right). \tag{20}$$

In the reverse process, clean samples are gradually generated from noisy samples. This process is defined as

$$p_\theta\left(\widehat{x}_{t-1} \mid \widehat{x}_t\right) = \mathcal{N}\left(\widehat{x}_{t-1}; \mu_\theta\left(\widehat{x}_t, t\right), \sigma_t\right), \tag{21}$$

where $\mu_\theta\left(\widehat{x}_t, t\right)$ is derived from removing the diffusion estimated $\epsilon_\theta\left(\widehat{x}_t, t\right)$ from the noisy samples $\widehat{x}_t$: $\mu_\theta\left(\widehat{x}_t, t\right) = \frac{1}{\sqrt{\alpha_t}}\left(\widehat{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta\left(\widehat{x}_t, t\right)\right)$, and σ_t denotes the reverse process variance. To train this model so that $p(x_0)$ approximates the true data distribution $q(x_0)$, the following variational lower bound (L_{vlb}), is optimized

$$L_{\text{vlb}} := \sum_{t=0}^{T} L_t, \tag{22}$$

$$L_0 := -\log p_\theta\left(x_0 \mid x_1\right), \tag{23}$$

$$L_{t-1} := D_{KL}\left(q\left(x_{t-1} \mid x_t, x_0\right) \| p_\theta\left(x_{t-1} \mid x_t\right)\right), \tag{24}$$

$$L_T := D_{KL}\left(q\left(x_T \mid x_0\right) \| p\left(x_T\right)\right), \tag{25}$$

where $D_{KL}(p(x)\|q(x)) = \int_{\mathcal{X}} p(x)\log(p(x)/q(x))\mathrm{d}x$ is the KL-divergence between two distribution $p(x)$ and $q(x)$. In practice, Ho et al. [2020] proposed a simplified training objective that predicts the Gaussian noise added at step t as

$$L_{\text{simple}} := \mathbb{E}_{t\sim[1,T],\, x_0\sim q(x_0),\, \epsilon\sim\mathcal{N}(0,\mathbf{I})}\left[\, \|\epsilon - \epsilon_\theta(x_t, t)\|_2^2 \,\right]. \tag{26}$$

This parameterization trains the model as a noise predictor $\epsilon_\theta(x_t, t)$ rather than directly regressing the mean $\mu_\theta(\hat{x}_t, t)$, which is equivalent up to a constant factor in the variational objective and is empirically easier to optimize.

B.2 Classifier guidance diffusion models

Classifier guidance Dhariwal and Nichol [2021] can be applied in the reverse process. This can guide the sampling trajectory toward regions of higher classifier likelihood and improve fidelity and class-consistency. We start with a diffusion model with an unconditional reverse noising process $p_\theta(x_{t-1} \mid x_t)$. We condition this on a label y, according to

$$p_{\theta,\phi}(x_t \mid x_{t+1}, y) \propto p_\theta(x_t \mid x_{t+1})\, p_\phi(y \mid x_t). \tag{27}$$

$\log p_\phi(y \mid x_t)$ can be approximated using a Taylor expansion around $x_t = \mu$ as

$$\begin{aligned} \log p_\phi(y \mid x_t) &\approx \log p_\phi(y \mid x_t)|_{x_t=\mu} + (x_t - \mu)\, \nabla_{x_t} \log p_\phi(y \mid x_t)|_{x_t=\mu} \\ &= (x_t - \mu)\, g + C_1. \end{aligned} \tag{28}$$

Here, $g = \nabla_{x_t} \log p_\phi(y \mid x_t)|_{x_t=\mu}$, and C_1 is a constant. This gives

$$\begin{aligned} \log\left(p_\theta(x_t \mid x_{t+1})\, p_\phi(y \mid x_t)\right) &\approx -\frac{1}{2}(x_t - \mu)^T \Sigma^{-1}(x_t - \mu) + (x_t - \mu)\, g + C_2 \\ &= \log p(z) + C_3, z \sim \mathcal{N}(\mu + \Sigma g, \Sigma). \end{aligned} \tag{29}$$

It is thus established that the conditional transition operator can be approximated by a Gaussian distribution analogous to its unconditional counterpart, with the mean shifted by Σg, where g denotes the classifier gradient. In practice, this gradient is computed as $g = \nabla_{\widehat{x}_t} \log\left(p(y \mid \widehat{x}_t)\right)$, which we obtain by taking the gradient of the log-probability with respect to the noisy input. Specifically, for a classifier f that outputs logits, we compute this as the gradient of the log-softmax $\nabla_{\widehat{x}_t} \log\left(\text{softmax}\left(f_y(\widehat{x}_t)\right)\right)$, where $f_y(\widehat{x}_t)$ denotes the logit corresponding to class y.

C Supplementary Notes

C.1 Smooth ECE and Huber Loss

The standard ECE measures the similarity between predicted confidences and empirical accuracies, typically using an absolute error term:

$$\mathcal{L}_{\text{ECE}}^{(i)} = \left|\hat{p}^{(i)} - a^{(i)}\right|, \tag{30}$$

where $\hat{p}^{(i)} = \max_c p_\phi(c \mid x^{(i)})$ denotes the model's predicted confidence, and $a^{(i)}$ indicates whether the prediction was correct.

This formulation, while simple and interpretable, is not differentiable at zero and may lead to instability or optimization challenges when used as a training loss.

To address this, we introduce a smoothed version:

$$\mathcal{L}_{\text{SmoothECE}}^{(i)} = \sqrt{\left(\hat{p}^{(i)} - a^{(i)}\right)^2 + \beta}, \tag{31}$$

where $\beta > 0$ is a small constant that controls the degree of smoothing.

This formulation closely resembles the **Huber loss**, a well-known technique for robust regression that combines the benefits of both ℓ_1 and ℓ_2 losses Huber [2011]. The Huber loss is defined as

$$\mathcal{L}_\delta(r) = \begin{cases} \frac{1}{2} r^2 & \text{if } |r| \le \delta, \\ \delta\left(|r| - \frac{1}{2}\delta\right) & \text{otherwise,} \end{cases} \tag{32}$$

where $r = \hat{p}^{(i)} - a^{(i)}$ is the calibration residual. The Huber loss behaves quadratically near zero (like ℓ_2) and linearly in the tails (like ℓ_1), which makes it robust to outliers while maintaining differentiability around the minimum.

- **Quadratic behavior near zero:** When $|\hat{p}^{(i)} - a^{(i)}| \ll \sqrt{\beta}$, we can use a Taylor expansion:

$$\sqrt{r^2 + \beta} \approx \sqrt{\beta} + \frac{1}{2\sqrt{\beta}} r^2 + \mathcal{O}(r^4), \tag{33}$$

  which behaves like a scaled ℓ_2 loss with a constant offset.

- **Linear behavior for large residuals:** When $|\hat{p}^{(i)} - a^{(i)}| \gg \sqrt{\beta}$, we get:

$$\sqrt{r^2 + \beta} \approx |r| + \frac{\beta}{2|r|} + \mathcal{O}\left(\frac{1}{|r|^3}\right), \tag{34}$$

  which converges to ℓ_1 behavior as $|r|$ increases.

Therefore, the Smooth ECE loss serves as a fully differentiable, Huber-like alternative to the standard ECE, with the added benefit of avoiding discontinuities in gradients.

C.2 Tilted sampling & Gradient weight

We want to use the information from all samples in a batch in a smart way. For this, we apply the *tilted loss* introduced in Li et al. [2023]. This loss has a parameter t which controls how much we focus on certain samples. By changing t, we can either put more weight on common (high-probability) samples, or on rare (outlier) samples.

The tilted loss is defined as

$$\mathcal{L}(t;\theta) := \frac{1}{t} \log\left(\frac{1}{N}\sum_{i=1}^{N} e^{t\,\ell(f(x_i),y;\theta)}\right) \tag{35}$$

where $\ell(f(x_i), y; \theta)$ is any loss function.

In our case, we replace the loss with the log-likelihood score. This gives

$$\mathcal{S}_{tilted}(t;x,y) := \frac{1}{t} \log\left(\frac{1}{N}\sum_{i=1}^{N} e^{t\ \log p_{\tau_1,\tau_2}(y|x_i)}\right) \tag{36}$$

Gradient form: Taking the gradient with respect to the x gives

$$\nabla_x \mathcal{S}_{tilted}(t;x,y) = \frac{\sum_{i=1}^{N} e^{t \log p_{\tau_1,\tau_2}(y|x_i)}\, \nabla_x \log p_{\tau_1,\tau_2}(y|x_i)}{\sum_{j=1}^{N} e^{t \log p_{\tau_1,\tau_2}(y|x_j)}} \tag{37}$$

This can be written as a weighted sum of per-sample gradients:

$$\nabla_x \mathcal{S}_{tilted}(t;x,y) = \sum_{i=1}^{N} w^i(t;x,y)\, \nabla_x \log p_{\tau_1,\tau_2}(y|x_i) \tag{38}$$

with weights

$$w^i(t;x,y) := \frac{e^{t \log p_{\tau_1,\tau_2}(y|x_i)}}{\sum_{j=1}^{N} e^{t \log p_{\tau_1,\tau_2}(y|x_j)}} \tag{39}$$

The weights $w^i(t;x,y)$ form a normalized distribution over the batch. If $t > 0$, high-probability samples get larger weights (mode-seeking). If $t < 0$, low-probability samples get larger weights (outlier-seeking). If $t = 0$, all samples have equal weight and follow the regular guidance.

Applying to classifier guidance: The tilted loss was first defined for parameter updates, but we can use the same idea for classifier gradients with respect to the sample x. This gives the tilted classifier gradient

$$g^i_{\text{tilted}} = w^i(t;x,y)\ \ \nabla_x \log p_{\tau_1,\tau_2}(y|x_i)|_{x_t=\mu} \tag{40}$$

As we know from Eq. 3 we achieve

$$g^i_{\text{tilted}} = w^i(t;x,y)\ \ [-\nabla_x E_{\tau_1}(x,y) + \nabla_x E_{\tau_2}(x)]|_{x_t=\mu} \tag{41}$$

Sampling step: The modified sampling rule becomes

$$z^i \sim \mathcal{N}\big(\mu + \Sigma g^i_{\text{tilted}},\ \Sigma\big) \tag{42}$$

In this way, the hyperparameter t directly controls the trade-off between *quality* (when $t > 0$, we push towards high-likelihood samples) and *diversity* (when $t < 0$, we encourage exploring lower-likelihood samples).

C.3 Introduction to f-divergences

f-divergences [Csiszár, 1967, Ali and Silvey, 1966] provide a general framework for measuring dissimilarity between probability distributions. In this work we adopt the discrete convention

$$D_f(q\|p) \ = \ \sum_{i=1}^{N} q(i)\, f\!\left(\frac{p(i)}{q(i)}\right), \tag{43}$$

where $f : \mathbb{R}_+ \to \mathbb{R}$ is convex and satisfies $f(1) = 0$. Convexity implies $D_f(q\|p) \geq 0$ with equality iff $p = q$.

Different choices of f recover classical divergences; Table 4 summarizes the divergences used.

Table 4: f-divergences used in this work with their generator functions.

Divergence	$f(t)$	$f'(t)$	Notes
Reverse KL ($D_{\mathrm{KL}}(q\|p)$)	$-\log t$	$-1/t$	penalizes $p\to 0$ when $q>0$
Forward KL ($D_{\mathrm{KL}}(p\|q)$)	$t\log t$	$\log t + 1$	penalizes $p>0$ when $q\to 0$
Jensen–Shannon	$-(t+1)\log\frac{t+1}{2} + t\log t$	$\log\frac{2t}{t+1}$	symmetric, bounded
Squared Hellinger	$(\sqrt{t}-1)^2$	$1-\frac{1}{\sqrt{t}}$	H is a metric (not H^2)

Gradient computation. Let $p(\cdot|x)$ denote the classifier output over N classes and q_y a fixed target distribution. For (43), define

$$w_f(q,p) \ := \ p\, f'\!\left(\frac{p}{q}\right). \tag{44}$$

Using $\nabla_x p(i|x) = p(i|x)\nabla_x \log p(i|x)$, we obtain

$$\nabla_x D_f(q_y\|p(\cdot|x)) \ = \ \sum_{i=1}^{N} w_f\big(q_y(i), p(i|x)\big)\, \nabla_x \log p(i|x), \tag{45}$$

and with the softmax identity $\nabla_x \log p(i|x) \ = \ \nabla_x f_i(x) - \sum_j p(j|x)\nabla_x f_j(x)$, this yields Proposition 2.

Properties and intuition. Most f-divergences are asymmetric, while JS and Hellinger are symmetric. The choice of f determines the weighting $w_f(q_y(i), p(i|x))$ in (45): for reverse KL, $w_f(q,p) = -q$ (non-vanishing as $p \to 0$); for forward KL, $w_f(q,p) = p\big(\log\frac{p}{q} + 1\big)$ (vanishes as $p \to 0$); for squared Hellinger, $w_f(q,p) = p - \sqrt{pq}$ (damped by the geometric mean term). These weightings help explain the different empirical precision–recall behaviors observed in Section 4.

C.4 Squared Hellinger Divergence

For completeness, we also evaluated squared Hellinger distance, defined by $f(t) = (\sqrt{t}-1)^2$, with derivative $f'(t) = 1 - \frac{1}{\sqrt{t}}$. Under our convention $t = \frac{p}{q}$, the corresponding weight is $w_f(q,p) = pf'(p/q) = p - \sqrt{pq}$.

Corollary 4 (Squared Hellinger gradient)**.** *The gradient of the squared Hellinger divergence regularized score is:*

$$\begin{aligned}\nabla_x \mathcal{S}_{H^2}(x,y) = \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x) \nabla_x f_i(x) \\ + \alpha \sum_{i=1}^{N} \sqrt{q_y(i) p(i|x)} \left[\nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x) \nabla_x f_j(x) \right].\end{aligned} \tag{46}$$

Unlike KL-type divergences, the squared Hellinger distance is a true metric satisfying the triangle inequality. Its square root weighting $\sqrt{q_y(i)p(i|x)}$ provides more moderate gradient corrections compared to the logarithmic terms in KL divergences or the linear terms in JS divergence.

Empirical evaluation on ImageNet 128×128 with ResNet-101 yielded FID = 2.15, (Table 5). Interestingly, Hellinger achieves the *highest precision* among all divergences, tied with FKL at 0.80, but exhibits the *lowest recall* at 0.58. This precision-recall trade-off reveals Hellinger's implicit mode-seeking behavior despite being a symmetric metric.

The square root weighting creates a natural damping effect: when $p(i|x) \gg q_y(i)$, the gradient $\sqrt{q_y(i)p(i|x)}$ grows sublinearly, strongly discouraging the model from placing mass in low-target regions. Conversely, when $p(i|x) \ll q_y(i)$, the gradient also grows slowly, providing weak encouragement to explore under-represented modes. This *bidirectional damping* results in conservative, high-fidelity generation that prioritizes precision over diversity.

While Hellinger outperforms the baseline (FID 2.19 → 2.15), it underperforms relative to RKL (2.14) and JS (2.13), and achieves the same recall as the baseline (0.58). This suggests that the bounded nature of Hellinger distance [always in $[0, 1]$ for probability distributions] limits its effectiveness for diffusion guidance, where stronger gradient signals are beneficial in early denoising steps. The metric's symmetric penalization of both under-coverage and over-extension may be suboptimal for conditional generation, where maintaining mode coverage is often more critical than preventing overshoot.

Table 5: Comparison of Squared Hellinger divergence result on ImageNet 128×128 with 50k samples.

Method	**Classifier**	**FID ↓**	**Precision ↑**	**Recall ↑**
Ma et al. [2023]	ResNet-101	2.19	0.79	0.58
Hellinger guided	ResNet-101	2.15	0.80	0.58

D Mathematical proofs

D.1 Proof of Proposition 1

Proof. The original guidance gradient is

$$\nabla_x \log p_{\tau_1,\tau_2}(y|x) = \tau_1 \nabla_x f_y(x) - \frac{\sum_{i=1}^{N} \exp(\tau_2 f_i(x)) \tau_2 \nabla_x f_i(x)}{\sum_{i=1}^{N} \exp(\tau_2 f_i(x))} \tag{47}$$

$$= \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x) \nabla_x f_i(x). \tag{48}$$

For the entropy gradient

$$\nabla_x H(p(\cdot|x)) = -\nabla_x \sum_{i=1}^{N} p(i|x) \log p(i|x) \tag{49}$$

$$= -\sum_{i=1}^{N} \left[\nabla_x p(i|x) \log p(i|x) + p(i|x) \nabla_x \log p(i|x) \right]. \tag{50}$$

Using the fact that $\nabla_x \log p(i|x) = (\nabla_x f_i(x) - \sum_j p(j|x)\nabla_x f_j(x))$, and $\nabla_x p(i|x) = p(i|x)\nabla_x \log p(i|x)$ we write

$$\nabla_x H(p(\cdot|x)) = -\sum_{i=1}^{N} p(i|x)[\log p(i|x) + 1]\nabla_x \log p(i|x) \tag{51}$$

$$\nabla_x H(p(\cdot|x)) = -\sum_{i=1}^{N} p(i|x)[\log p(i|x) + 1] \left[\nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x)\nabla_x f_j(x)\right] \tag{52}$$

Combining them results to

$$\nabla_x \mathcal{S}_{\text{entropy}}(x, y) = -\nabla_x E_{\tau_1}(x, y) + \nabla_x E_{\tau_2}(x) + \lambda(t)\nabla_x H(p(\cdot|x)) \tag{53}$$

$$= \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x)\nabla_x f_i(x) - \lambda(t) \sum_{i=1}^{N} p(i|x)[\log p(i|x) + 1] \left[\nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x)\nabla_x f_j(x)\right]. \tag{54}$$

□

The additional $+1$ term in $[\log p(i|x) + 1]$ ensures proper normalization of the gradient, while the negative sign indicates that the gradient opposes concentration toward any single mode.

D.2 Proof of Proposition 2

Proof. Let $D_f(q_y\|p) = \sum_i q_i\, f\left(\frac{p_i}{q_i}\right)$, where $p_i := p(i|x),\ q_i := q_y(i)$. Taking the gradient with respect to x, and noting that q_i is constant:

$$\begin{aligned} \nabla_x D_f &= \sum_i q_i\, f'\left(\frac{p_i}{q_i}\right) \nabla_x \left(\frac{p_i}{q_i}\right) \\ &= \sum_i f'\left(\frac{p_i}{q_i}\right) \nabla_x p_i. \end{aligned} \tag{55}$$

Using $\nabla_x p_i = p_i\, \nabla_x \log p_i$ and $\nabla_x \log p_i = \nabla_x f_i(x) - \sum_j p_j\, \nabla_x f_j(x) =: g_i(x)$, we obtain

$$\nabla_x D_f = \sum_i p_i\, f'\left(\frac{p_i}{q_i}\right) g_i(x) = \sum_i w_f(q_i, p_i)\, g_i(x). \tag{56}$$

Combining with $\nabla_x \log p_{\tau_1,\tau_2}(y|x) = \tau_1 \nabla_x f_y(x) - \tau_2 \sum_i p_{\tau_2}(i|x)\, \nabla_x f_i(x)$ and the negative sign in $\mathcal{S}_{\mathcal{D}} = \log p_{\tau_1,\tau_2}(y|x) - \alpha D_f(q_y\|p)$ gives Proposition 2. □

D.3 Proof of Corollary 1

Proof. Recall Proposition 2 gives

$$\nabla_x \mathcal{S}(x, y) = \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x)\, \nabla_x f_i(x) - \alpha \nabla_x D_f(q_y\|p(\cdot|x)), \tag{57}$$

and for an f-divergence of the form $D_f(q\|p) = \sum_i q(i) f\left(\frac{p(i)}{q(i)}\right)$,

$$\nabla_x D_f(q_y\|p(\cdot|x)) = \sum_{i=1}^{N} w_f\big(q_y(i), p(i|x)\big)\, g_i(x), \qquad w_f(q, p) = p\, f'\left(\frac{p}{q}\right), \tag{58}$$

where $g_i(x) = \nabla_x \log p(i|x)$.

For reverse KL, $D_{\mathrm{KL}}(q_y\|p) = \sum_i q_y(i) \log \frac{q_y(i)}{p(i|x)}$ corresponds to $f(t) = -\log t$ with $t = \frac{p}{q}$. Hence $f'(t) = -1/t$ and

$$w_f\big(q_y(i), p(i|x)\big) = p(i|x)\, f'\left(\frac{p(i|x)}{q_y(i)}\right) = p(i|x)\left(-\frac{q_y(i)}{p(i|x)}\right) = -\,q_y(i). \tag{59}$$

Therefore,

$$\nabla_x D_{\mathrm{KL}}(q_y\|p(\cdot|x)) = \sum_{i=1}^{N} \big(-\,q_y(i)\big)\, g_i(x) = -\sum_{i=1}^{N} q_y(i)\, g_i(x), \tag{60}$$

and the regularizer contribution in the score is $-\alpha\nabla_x D_{\mathrm{KL}}(q_y\|p) = +\alpha \sum_i q_y(i) g_i(x)$. Substituting this into Proposition 2 yields Corollary 1. □

D.4 Proof of Corollary 2

Proof. Forward KL is $D_{\mathrm{KL}}(p\|q_y) = \sum_i p(i|x) \log \frac{p(i|x)}{q_y(i)}$. With our f-divergence convention $D_f(q\|p) = \sum_i q(i) f\Big(\frac{p(i)}{q(i)}\Big)$, this corresponds to $f(t) = t \log t$ (again with $t = \frac{p}{q}$). Indeed, $\sum_i q(i)\, t \log t = \sum_i p(i) \log \frac{p(i)}{q(i)}$. Then $f'(t) = \log t + 1$, and the associated weight is

$$w_f\big(q_y(i), p(i|x)\big) = p(i|x)\Big(\log \frac{p(i|x)}{q_y(i)} + 1\Big). \tag{61}$$

Hence,

$$\nabla_x D_{\mathrm{KL}}(p\|q_y) = \sum_{i=1}^{N} p(i|x)\Big(\log \frac{p(i|x)}{q_y(i)} + 1\Big)\, g_i(x), \qquad g_i(x) = \nabla_x \log p(i|x), \tag{62}$$

and plugging into Proposition 2 gives the regularizer term $-\alpha\nabla_x D_{\mathrm{KL}}(p\|q_y) = -\alpha \sum_i p(i|x)\big(\log \frac{p(i|x)}{q_y(i)} + 1\big) g_i(x)$, which is exactly Corollary 2. □

D.5 Proof of Corollary 3

Proof. The Jensen–Shannon divergence is

$$D_{\mathrm{JS}}(q_y\|p) = \frac{1}{2} D_{\mathrm{KL}}(q_y\|m) + \frac{1}{2} D_{\mathrm{KL}}(p\|m), \qquad m(i) = \frac{q_y(i) + p(i|x)}{2}. \tag{63}$$

Since q_y is fixed, only $p(\cdot|x)$ (and thus m) depends on x. Differentiate D_{JS} w.r.t. $p(i|x)$ (treating p as free variables first). A direct computation gives the partial derivative

$$\frac{\partial}{\partial p(i)} D_{\mathrm{JS}}(q_y\|p) = \frac{1}{2} \log \frac{p(i)}{m(i)}. \tag{64}$$

(Indeed, the $q_y\|m$ term contributes $-\frac{q_y(i)}{4m(i)}$, the $p\|m$ term contributes $\frac{1}{2}\big(\log \frac{p(i)}{m(i)} + 1\big) - \frac{p(i)}{4m(i)}$, and the rational terms cancel because $q_y(i) + p(i) = 2m(i)$.)

Therefore, by the chain rule,

$$\nabla_x D_{\mathrm{JS}}(q_y\|p(\cdot|x)) = \sum_{i=1}^{N} \frac{1}{2} \log \frac{p(i|x)}{m(i)}\, \nabla_x p(i|x) = \sum_{i=1}^{N} \frac{p(i|x)}{2} \log \frac{p(i|x)}{m(i)}\, g_i(x), \tag{65}$$

where $\nabla_x p(i|x) = p(i|x) g_i(x)$ and $g_i(x) = \nabla_x \log p(i|x)$. Since $\frac{p(i|x)}{m(i)} = \frac{2p(i|x)}{p(i|x)+q_y(i)}$, this can be written as

$$\nabla_x D_{\mathrm{JS}}(q_y\|p(\cdot|x)) = \sum_{i=1}^{N} \frac{p(i|x)}{2} \log \frac{2p(i|x)}{p(i|x) + q_y(i)}\, g_i(x). \tag{66}$$

Substituting into Proposition 2 and applying the $-\alpha$ factor in the score yields Corollary 3. □

D.6 Proof of Lemma 1

Proof. By definition,

$$D_{\mathrm{KL}}(q_y\|p(\cdot|x)) = \sum_{i=1}^{N} q_y(i)\log\frac{q_y(i)}{p(i|x)} = \text{const} - \sum_{i=1}^{N} q_y(i)\log p(i|x), \tag{67}$$

so

$$-\nabla_x D_{\mathrm{KL}}(q_y\|p(\cdot|x)) = \sum_{i=1}^{N} q_y(i)\,\nabla_x \log p(i|x). \tag{68}$$

For softmax probabilities $p(i|x) = \frac{e^{f_i(x)}}{\sum_k e^{f_k(x)}}$,

$$\nabla_x \log p(i|x) = \nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x)\nabla_x f_j(x). \tag{69}$$

Multiplying by $q_y(i)$ and summing over i gives

$$\begin{aligned} -\nabla_x D_{\mathrm{KL}}(q_y\|p(\cdot|x)) &= \sum_{i=1}^{N} q_y(i)\nabla_x f_i(x) - \Big(\sum_{i=1}^{N} q_y(i)\Big)\sum_{j=1}^{N} p(j|x)\nabla_x f_j(x) \\ &= \sum_{i=1}^{N} q_y(i)\nabla_x f_i(x) - \sum_{j=1}^{N} p(j|x)\nabla_x f_j(x), \end{aligned} \tag{70}$$

since $\sum_i q_y(i) = 1$. This is the claimed decomposition. □

D.7 Proof of Proposition 3

Proof. Let $\ell_k(x) = \log b_k + \log f_k(x)$. For Gaussian components, $\nabla_x \ell_k(x) = \nabla_x \log f_k(x) = \Sigma_k^{-1}(\mu_k - x)$. Additionally, since $\nabla_x \log f_k(x) = \frac{1}{f_k(x)}\nabla_x f_k(x) \to \nabla_x f_k(x) = f_k(x)\nabla_x \log f_k(x)$. Substituting these into Corollary 1 gives

$$\begin{aligned} \nabla_x \mathcal{S}_{\mathrm{RKL}}(x,y) &= \tau_1 f_y(x)\nabla_x \ell_y(x) - \tau_2 \sum_{k=1}^{K} p_{\tau_2}(k|x) f_k(x)\nabla_x \ell_k(x) \\ &\quad + \alpha \sum_{k=1}^{K} q_y(k)\Big(f_k(x)\nabla_x \ell_k(x) - \sum_{j=1}^{K} f_j(x) w_j(x)\nabla_x \ell_j(x)\Big) \\ &= \sum_{k=1}^{K} f_k(x)\Sigma_k^{-1}(\mu_k - x) \\ &\quad \times \Big[\tau_1 \mathbb{I}_{k=y} - \tau_2 p_{\tau_2}(k|x) + \alpha q_y(k)\Big] \\ &\quad - \alpha\Big(\sum_{k=1}^{K} q_y(k)\Big)\sum_{j=1}^{K} f_j(x) w_j(x)\Sigma_j^{-1}(\mu_j - x). \end{aligned} \tag{71}$$

Since $\sum_k q_y(k) = 1$, collecting the coefficient of each $f_k(x)\Sigma_k^{-1}(\mu_k - x)$ yields $\Gamma_k(x, y, \alpha, \epsilon) = \tau_1 \mathbb{I}_{k=y} - \tau_2 p_{\tau_2}(k|x) + \alpha(q_y(k) - w_k(x))$ as stated. □

D.8 Proof of Corollary 4

Proof. The squared Hellinger distance can be written as

$$D_{H^2}(q_y\|p(\cdot|x)) = \sum_{i=1}^{N}\big(\sqrt{q_y(i)} - \sqrt{p(i|x)}\big)^2 = \sum_i q_y(i) + \sum_i p(i|x) - 2\sum_i \sqrt{q_y(i)p(i|x)}. \tag{72}$$

Since $\sum_i q_y(i) = 1$ and $\sum_i p(i|x) = 1$, this reduces to $D_{H^2}(q_y\|p) = 2 - 2\sum_i \sqrt{q_y(i)p(i|x)}$. Thus,

$$\nabla_x D_{H^2}(q_y\|p(\cdot|x)) = -2\sum_{i=1}^{N} \nabla_x \sqrt{q_y(i)p(i|x)} = -2\sum_{i=1}^{N} \frac{\sqrt{q_y(i)}}{2\sqrt{p(i|x)}}\, \nabla_x p(i|x). \tag{73}$$

Using $\nabla_x p(i|x) = p(i|x)\, g_i(x)$ with $g_i(x) = \nabla_x \log p(i|x)$,

$$\nabla_x D_{H^2}(q_y\|p(\cdot|x)) = -\sum_{i=1}^{N} \sqrt{q_y(i)p(i|x)}\, g_i(x). \tag{74}$$

Finally, for the score $\mathcal{S}_{H^2}(x, y) = \log p_{\tau_1,\tau_2}(y|x) - \alpha D_{H^2}(q_y\|p(\cdot|x))$, we obtain

$$\nabla_x \mathcal{S}_{H^2}(x, y) = \tau_1 \nabla_x f_y(x) - \tau_2 \sum_{i=1}^{N} p_{\tau_2}(i|x) \nabla_x f_i(x) + \alpha \sum_{i=1}^{N} \sqrt{q_y(i)p(i|x)}\, g_i(x), \tag{75}$$

which is the stated corollary (after rewriting $g_i(x)$ via the softmax identity if desired). □

E Implementation details

E.1 Fine-Tuning with ECE Smooth Regularization

To improve the calibration of the classifier without compromising its predictive performance, we incorporate the ECE Smooth loss as a regularization term alongside the standard cross-entropy objective. Specifically, during fine-tuning, the total loss combines the classification loss with the ECE Smooth term. The following algorithm outlines the training procedure used in our implementation

Algorithm 2 ECE Smooth Fine-Tuning

Require: Number of epochs E, batch size B, regularization weight λ, classifier $\mathcal{C}_\phi$, smoothing constant β

for $e = 1, \ldots, E$ **do**
  for each batch $\{(x^{(i)}, y^{(i)})\}_{i=1}^{B}$ **do**
    $\hat{p}^{(i)} \leftarrow \mathcal{C}_\phi(x^{(i)})$ ▷ Predicted probability vector
    $a^{(i)} \leftarrow \mathbb{1}\left(\arg\max_j \hat{p}_j^{(i)} = y^{(i)}\right)$ ▷ Correctness indicator
    Compute classification loss:
      $\mathcal{L}_{\text{CE}} = \frac{1}{B}\sum_{i=1}^{B} -\log \hat{p}_{y^{(i)}}^{(i)}$
    Compute ECE smooth regularizer:
      $\mathcal{L}_{\text{ECE}} = \frac{1}{B}\sum_{i=1}^{B} \sqrt{\left(\max_j \hat{p}_j^{(i)} - a^{(i)}\right)^2 + \beta}$
    Combine losses: $\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda \cdot \mathcal{L}_{\text{ECE}}$
    Update ϕ using gradient of $\mathcal{L}$
  end for
end for
return fine-tuned classifier $\mathcal{C}_\phi$

We fine-tuned the official Dhariwal and Nichol [2021] classifier from here using ECE Smooth regularizer with a regularization weight of $\lambda = 1$ and a smoothing constant of $\beta = 0.0001$. The training was performed for a total of 12000 iterations with $B = 256$ in our setup. We used a single NVIDIA H100 80 GB and the official ILSVRC2012 dataset for fine-tuning.

E.2 Sampling

The off-the-shelf classifiers are the official Pytorch checkpoints at here. We used the (γ_t) schedule and other default settings from Ma et al. [2023] to ensure a fair comparison, and incorporated our proposed methods on top of this baseline to assess their improvements. We

used the diffusion model from Dhariwal and Nichol [2021] without retraining. Through all the experiments, we evaluated the FID score using a batch size of $B = 256$, and the reference dataset batches contain pre-computed statistics over the whole dataset provided here. We also used a single NVIDIA A100 SXM4 40 GB for this stage.

The divergence regularization adds negligible computational overhead. All methods share $\mathcal{O}(B \cdot C)$ complexity with the baseline, where divergence computation (one softmax + one divergence term) adds 50-80% cost to gradient computation. Since classifier gradients represent a small fraction of total sampling time (dominated by diffusion model forward passes), end-to-end overhead is less than 3% on ImageNet 128×128 with 250 DDPM steps. Memory overhead is also minimal (<12 MB for batch size 256).

F Additional experiments

F.1 Tilted sampling

Table 6 presents an ablation study of the TERM tilt parameter t for classifier-guided diffusion sampling. The results demonstrate a non-monotonic relationship between t and generation quality measured by FID. Optimal performance is achieved at $t = -0.2$ (FID = 5.28), improving upon the standard averaging baseline at $t = 0$ (FID = 5.34). Performance degrades at both extremes: positive tilting ($t = 0.1$) yields FID = 5.45, while increasingly negative values deteriorate from FID = 5.30 at $t = -0.1$ to FID = 5.63 at $t = -0.5$.

The optimal performance at mild negative tilting ($t = -0.2$) reveals that classifier guidance benefits from down-weighting high-confidence predictions. When $t < 0$, TERM effectively implements confidence tempering by emphasizing lower-probability samples, preventing the diffusion process from over-optimizing toward potentially spurious classifier modes. This acts as an implicit regularizer that reduces sensitivity to classifier overconfidence and biases. In contrast, positive tilting amplifies already-dominant high-confidence predictions, leading to less diverse generations and potential adversarial artifacts. The degradation at strongly negative values ($t = -0.5$) suggests that excessive emphasis on low-confidence predictions provides overly weak guidance signals. Thus, the sweet spot at $t = -0.2$ balances leveraging classifier knowledge while avoiding over-reliance on its most confident predictions, resulting in more stable and higher-quality image generation.

Table 6: Ablation study of t in tilted guidance with respect to FID. The classifier is the ResNet-50. The diffusion model is from Dhariwal and Nichol [2021]. We generate 10k ImageNet 128×128 samples with 250 DDPM steps for evaluation.

t	0.1	0	−0.1	−0.2	−0.3	−0.5
FID	5.45	5.34	5.30	5.28	5.40	5.63

F.2 Entropy guidance sampling

The hyperparameter here is the adaptive entropy regularization weight λ_t. We first tested a constant value $\lambda_t = 0.1$, which resulted in almost the same FID as the baseline. Therefore, we selected an adaptive approach for the weight as $\lambda_t \in [0.05, 0.2]$. This will start from the greatest value at the start during early denoising steps when the image is heavily corrupted and classifier predictions are unreliable, then gradually decreases to the minimum value. This time-dependent approach aligns with the inherent dynamics of the diffusion process; early steps benefit from exploration (high entropy) while later steps require precision (low entropy) to preserve fine-grained class characteristics. This resulted in the best FID, which was reported in the experiments.

F.3 Divergence guidance sampling

The selection of the target distribution $q_y(\cdot)$ in Equation (7) is critical for balancing class fidelity with mode coverage. A complete uniform distribution $q_y(i) = 1/N$ would be class-

agnostic and detrimental to conditional generation, while a one-hot distribution $q_y(i) = \mathbb{I}_{i=y}$ would eliminate the regularization's diversity-preserving effect. We adopt a parameterized target distribution:

$$q_y(i) = (1 - \epsilon)\frac{1}{N} + \epsilon\, \mathbb{I}_{i=y}, \tag{76}$$

where $\epsilon \in [0, 1]$ controls the bias toward the target class y. This formulation:

- At $\epsilon = 0$: reduces to uniform, maximizing diversity but losing class conditioning
- At $\epsilon = 1$: reduces to one-hot, maximizing class fidelity but losing regularization
- At $\epsilon \in (0, 1)$: balances target class emphasis with exploration of adjacent modes

Based on our experiments (see Table 7), we selected $\epsilon = 0.1$. Table 7 demonstrates the FID analysis for the weight parameters.

Table 7: Ablation study of α in divergence guidance with respect to FID. The classifier is the ResNet-50. The diffusion model is from Dhariwal and Nichol [2021]. We generate 10k ImageNet 128×128 samples with 250 DDPM steps for evaluation.

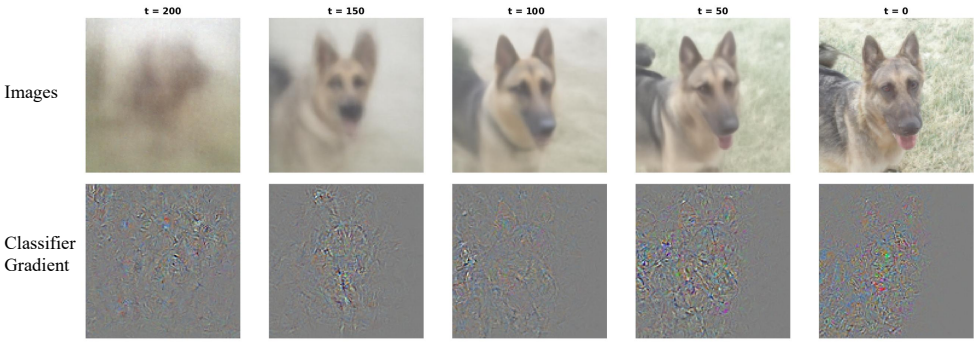
α	0.0	0.05	0.1	0.15	0.1	0.17	0.1
ϵ	0.0	0.1	0.1	0.1	0.05	0.08	0.2
FID	5.34	5.31	5.12	5.29	5.37	5.22	5.41

Thus, we selected $\alpha = 0.1$ for our experiments on Resnet-50. We also evaluated a similar study for JS and forward KL divergences using Resnet-101. Table 8 demonstrates the FID analysis for the weight parameter with $\epsilon = 0.1$.

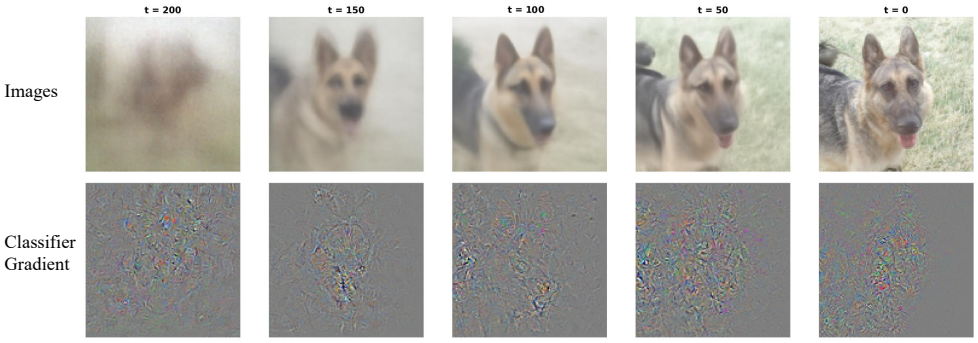
Table 8: Ablation study of α in divergence guidance with respect to FID. The classifier is the ResNet-101. The diffusion model is from Dhariwal and Nichol [2021]. We generate 50k ImageNet 128×128 samples with 250 DDPM steps for evaluation.

α	0.1	0.09	0.08	0.1	0.09	0.08
Divergence	JS	JS	JS	FKL	FKL	FKL
FID	2.16	2.13	2.17	2.20	2.16	2.18

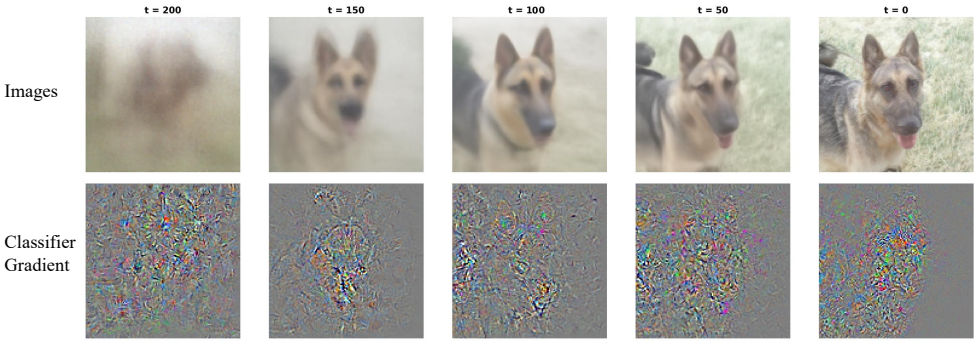
Therefore, $\alpha = 0.09$ was the optimal value for our experiments using Resnet-101. Figure 3 provides a visual comparison of classifier gradient behaviors across three guidance strategies during the diffusion sampling process, with differences in both the gradient evolution and final image quality. The visualization tracks the evolution from heavily noised samples (t=200) to clean images (t=0), revealing distinct patterns in how each method maintains gradient activity and ultimately affects the generated output. In the baseline approach 3a with equal temperatures, the classifier gradients exhibit rapid concentration. The improved baseline 3b shows marginally better gradient preservation but still suffers from premature collapse in the middle stages (t=100 to t=50), producing a final image with comparable quality but subtle differences in texture rendering. In contrast, our reverse KL divergence regularization method 3c demonstrates persistent gradient activity throughout the entire sampling trajectory, culminating in a final image with noticeably sharper details, better color saturation, and more natural texture—particularly visible in the dog's fur and facial features. The gradient maps maintain rich, distributed patterns even in later denoising steps, with visible activity across multiple spatial regions rather than collapsing to isolated points. This sustained gradient diversity validates our theoretical analysis.



(a) Baseline ($\tau_1 = \tau_2 = 1$)



(b) Baseline ($\tau_1 = 1$ and $\tau_2 = 0.5$) Ma et al. [2023]



(c) Reverse KL divergence regularization

Figure 3: The visual comparison of intermediate sampling pictures and classifier gradient figures. The seed is fixed for direct comparison.