ADVFD: BOOSTING VISUAL GENERATION VIA ADVERSARIAL FRÉCHET DISTANCE LOSS

Anonymous authors
Paper under double-blind review

ABSTRACT

Fréchet distance has recently emerged as an effective distribution-level objective for generator post-training, complementing the conventional sample-level diffusion and flow-matching losses. However, directly optimizing Fréchet objectives can cause *Fréchet hacking*. The target metrics keep improving, but visual quality and Fréchet alignment in other feature spaces may stagnate or deteriorate. We attribute this failure to the static pretrained feature spaces used by existing Fréchet losses. These feature spaces provide incomplete and fixed views of the differences between real and generated distributions. To address this limitation, we propose *Adversarial Fréchet Distance* (AdvFD), which complements the static representation targets in FD-Loss with a calibrated adversarially learned representation. AdvFD augments the original static Fréchet objective with a learnable representation that adversarially maximizes the Fréchet discrepancy between real and generated samples, while the generator minimizes the same discrepancy in the resulting adaptive feature space. To prevent the adversarial representation from trivially increasing the objective through feature amplification, we further introduce real-feature whitening, which normalizes its scale and covariance geometry and stabilizes the min–max optimization. Extensive experiments show that AdvFD consistently improves one-step generator post-training across both JiT and pMF backbones and across different model scales.

1 INTRODUCTION

Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2020; Rombach et al., 2022) have become a dominant approach to high-fidelity visual generation. Flow-based methods (Lipman et al., 2022; Liu et al., 2022a; Albergo et al., 2025) provide a closely related view by transporting a simple prior toward the data distribution. Their objectives are defined over samples from the data distribution and the corresponding perturbation paths. In practice, the training process asks the model to predict the target noise, score, or velocity at sampled intermediate states. It does not explicitly minimize the discrepancy between their distribution and the real data distribution.

To move beyond purely sample-level supervision, FD-loss (Yang et al., 2026a) introduces a distribution-matching objective for generator post-training. It minimizes the Fréchet distance between real and generated feature distributions extracted by pretrained encoders, such as Inception (Szegedy et al., 2015), SigLIP (Zhai et al., 2023), and MAE (He et al., 2022), thereby aligning their first- and second-order statistics without requiring paired targets. Although FD-Loss performs well, its target metrics can keep improving as training continues, while visual quality and Fréchet scores from other encoders may remain unchanged or even worsen. As shown in Figure 1 (a), post-training with FD-Loss introduces pronounced visual artifacts. Figure 2 (b) quantifies the same issue: FD-r-Inception, the representation used for training, improves by 29.4%, while FD-r-CLIP which is not used for training, worsens by 8.5%.

The Limits of Static Representation Targets. Why can the optimized metrics improve while visual quality and Fréchet scores from other encoders worsen? The issue arises because FD-Loss constrains the generator through predefined pretrained representations. Since each encoder captures only particular aspects of image semantics, structure, and texture, discrepancies outside its representation scope receive weak penalty. As optimization proceeds, the generator can become increasingly specialized to the selected feature spaces, continuing to reduce the target FD while preserving errors

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

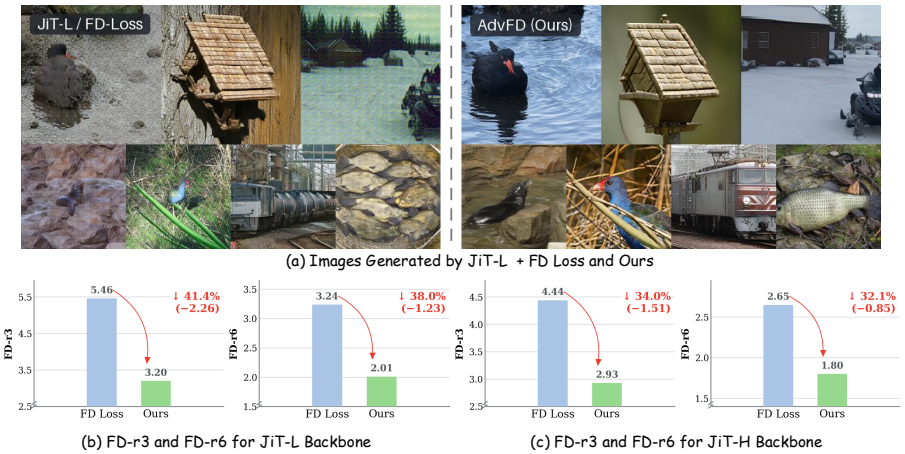


Figure 1: **AdvFD mitigates Fréchet hacking in one-step generation. Top:** Compared with JiT-L trained using the static FD loss, AdvFD generates cleaner and more coherent images under the same 1-NFE sampling budget. **Bottom:** AdvFD consistently reduces FD-r3 and FD-r6 across JiT-L and JiT-H, with relative improvements of 41.4%/38.0% and 34.0%/32.1%, respectively. These results demonstrate improved perceptual quality and generalization across evaluation representations. Lower is better for all metrics.

that remain visible to humans or other encoders. Figure 2 (a) provides a direct example: a learned universal perturbation introduces visible high-frequency artifacts while reducing Inception FID. We refer to this behavior as *Fréchet hacking*. Adding more frozen encoders may broaden the supervision, but the comparison spaces remain static. This motivates learning an adversarial representation that evolves with the generated distribution.

To address this issue, we propose *Adversarial Fréchet Distance* (AdvFD), which complements the static encoder set in FD-Loss with a learnable representation. This representation is optimized to expose residual discrepancies between the real and generated distributions, while the generator continues to minimize a distribution-level Fréchet objective introduced by this representation. However, directly maximizing Fréchet distance with a trainable encoder can lead to trivial feature amplification and unstable optimization. We therefore introduce real-feature whitening to normalize the scale and covariance geometry of the adaptive feature space, yielding a dynamic yet stable representation target. AdvFD can be readily applied to one-step generators and consistently improves both training and cross-encoder Fréchet metrics.

We conduct extensive experiments on ImageNet class-conditional generation across multiple generators. As shown in Figure 1, AdvFD effectively reduces visual artifacts caused by optimizing static representation targets and consistently improves both the optimized Fréchet metrics and those computed with feature encoders not used during training. Our contributions are summarized as follows:

- We identify *Fréchet hacking*, where optimizing static Fréchet objectives improves target metrics but can degrade visual quality and alignment under other representations.
- We propose *Adversarial Fréchet Distance* (AdvFD), which complements static FD targets with an adversarially learned representation that exposes residual real–generated discrepancies.
- We introduce real-feature whitening to stabilize adversarial representation learning. AdvFD consistently improves generation quality across backbones and model scales, outperforming FD-loss and alternative remedies.

2 Related Work

2.1 Fréchet Distance in Visual Generation

Fréchet Distance as Evaluation. Fréchet-based distances have become standard tools for evaluating image generation (Brock et al., 2018; Karras et al., 2019; Dhariwal & Nichol, 2021; Peebles &

Xie, 2023; Ma et al., 2024; Li & He, 2026; Zheng et al., 2025). Among them, the Fréchet Inception Distance (FID) (Heusel et al., 2017) compares real and generated distributions through the first- and second-order statistics of features extracted from a pretrained Inception network (Szegedy et al., 2015). While FID is simple and widely correlated with visual quality, it is fundamentally tied to the Inception feature space and is affected by finite-sample estimation and the Gaussian approximation of feature distributions (Bińkowski et al., 2018; Jayasumana et al., 2024). These limitations have motivated evaluation protocols that replace the underlying feature encoder or adopt alternative distributional discrepancies (Jayasumana et al., 2024; Yang et al., 2026a; Berthet et al., 2026).

Fréchet Distance as Training. Prior work has incorporated feature-statistic discrepancies into adversarial objectives for GAN training. McGAN (Mroueh et al., 2017) matches feature means and covariances through norm-based IPMs, Fisher GAN (Mroueh & Sercu, 2017) normalizes a scalar critic by its second moment, MMD-GAN (Li et al., 2017) maximizes kernel MMD through a constrained encoder, and Fréchet GAN (Doan et al., 2020) directly computes Fréchet distance on discriminator features. Despite their different formulations, these methods are all developed for GANs and jointly learn the feature representation and generator from scratch. More recently, FD-Loss (Yang et al., 2026a) uses frozen pretrained encoders to provide a stable Fréchet training objective for post-training diffusion models. However, once these fixed representations become optimization targets, the generator can over-optimize their feature statistics while improving much more slowly under held-out representations, a phenomenon we term *Fréchet hacking*. AdvFD extends Fréchet-based training to diffusion models by retaining the stable pretrained representations and introducing an adaptive branch that evolves with the generator. This allows the training objective to continuously expose residual distributional mismatches that are missed by fixed encoders, thereby mitigating Fréchet hacking and achieving stronger performance under both training and held-out representations.

2.2 Mitigating Objective Hacking in Generative Training

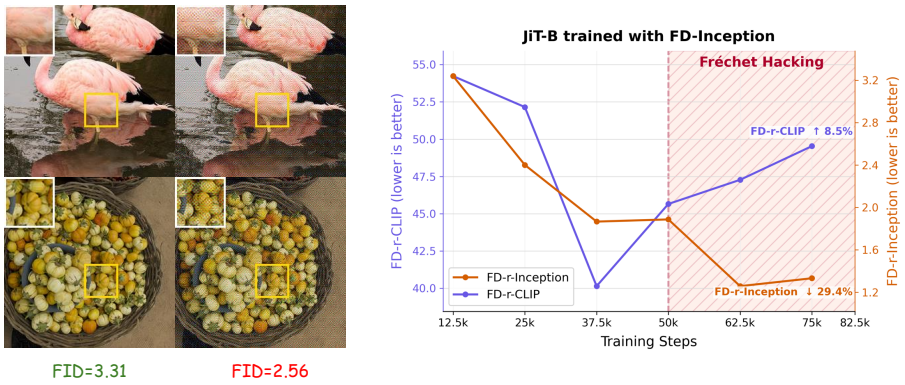
Objective hacking arises when a generator over-optimizes an imperfect proxy, improving the measured objective by exploiting unpenalized artifacts rather than improving the intended visual quality (Pan et al., 2022; Gao et al., 2023). Existing mitigation strategies can be broadly grouped into pretraining-stage and post-training approaches. During pretraining-stage distribution matching, prior works alleviate the limitations of incomplete supervision through adaptive feedback, including patch-level critics, pretrained-feature discriminators, teacher-based distribution matching, and adversarial diffusion distillation (Isola et al., 2017; Sauer et al., 2021; Kumari et al., 2022; Yin et al., 2024b;a; Sauer et al., 2024b;a; Lu et al., 2025). In post-training alignment, models directly optimize learned aesthetic, preference, or alignment rewards, which can lead to reward over-optimization; recent methods mitigate this by calibrating reward confidence, constraining reward optimization, or adversarially updating reward models against generated samples (Kim et al., 2024; Zhai et al., 2025; Mao et al., 2026). While these methods reduce proxy exploitation through external critics, rewards, or distillation signals, AdvFD targets Fréchet hacking in pretrained feature-space distribution matching itself by calibrating the representation space.

3 Fréchet Hacking under Static Representation Matching

Feature-space distribution matching provides a direct training objective for visual generation by comparing real and generated distributions without paired reconstruction targets. FD-Loss (Yang et al., 2026a) implements this idea by minimizing the Fréchet distance between their feature distributions, thereby aligning first- and second-order statistics without teacher distillation. Results show that directly optimizing such representation-space distribution distances can effectively improve generator post-training. However, this success rests on an implicit assumption:

- **Current Assumption:** *Aligning generated and real samples in a static pretrained feature space could improve visual quality.*

However, this assumption does not always hold. The generator is supervised only by the static feature spaces, so it mainly corrects the discrepancies that these representations can detect. Differences they fail to capture may remain weakly constrained, allowing important visual structures and seman-



(a) Visible noise lowers Inception FID but degrades image quality

(b) Fréchet Hacking Emerges During FD-Inception Training

Figure 2: **Fréchet hacking under a static Inception representation. Left:** A universal learned perturbation optimized solely for Inception FID introduces visible high-frequency artifacts while reducing FID from 3.31 to 2.56, directly revealing an exploitable blind direction of the static representation. **Right:** The same failure mode emerges during JiT-B post-training: from 50k-75k steps, FD-r-Inception decreases by 29.4%, whereas FD-r-CLIP increases by 8.5%. Together, these results show that optimizing a static Fréchet objective can improve the target metric while degrading visual quality and distributional alignment in a feature space not used for training.

tic details to degrade during training. As a result, the optimized FD may continue to decrease even when visual quality and alignment in other feature spaces, such as CLIP, stagnate or worsen.

We illustrate the limitations of using the static Inception representation in Figure 2. **Left:** We freeze a pretrained pMF-B generator and learn a universal perturbation solely by minimizing its Inception FID. Although the learned noise introduces clearly visible artifacts, it reduces FID from 3.31 to 2.56, demonstrating that the static representation contains directions along which visual quality can deteriorate while the target metric improves. **Right:** The same limitation appears during FD-Inception post-training: from 50k to 75k steps, FD-r-Inception decreases by 29.4%, whereas FD-r-CLIP increases by 8.5%. These results show that a static feature space can overlook perceptually relevant discrepancies and provide a misleading optimization signal.

From Static to adversarial representations. These observations motivate extending the representation scope of Fréchet training beyond a fixed encoder set. Although adding more frozen encoders can broaden feature coverage, it also increases computational and memory costs while remaining limited to predefined representations. More importantly, a static ensemble may still overlook discrepancies that emerge as the generator adapts to the training objectives. We therefore introduce a learnable representation that is updated using the current real and generated distributions. The representation enlarges their Fréchet discrepancy to expose residual mismatches, while the generator minimizes the same discrepancy in the resulting feature space. As the generated distribution evolves, the representation adapts accordingly, providing dynamic and complementary supervision beyond the fixed encoders.

4 Method

4.1 Adaptive Fréchet Loss

Let p denote the real data distribution, q_θ the distribution induced by the generator G_θ, and $\mathcal{F}$ a set of frozen visual representations, including SigLIP (Zhai et al., 2023), MAE (He et al., 2022), and Inception (Szegedy et al., 2015). Following FD-Loss, we define the static Fréchet objective as

$$D_{\text{static}}\left(p, q_\theta\right) = \sum_{\phi \in \mathcal{F}} \lambda_\phi D_{\text{FD}}^{\phi}\left(p, q_\theta\right), \tag{1}$$

where λ_ϕ controls the contribution of each representation. These frozen feature spaces provide stable distribution-level supervision but may overlook residual discrepancies as the generated distribution

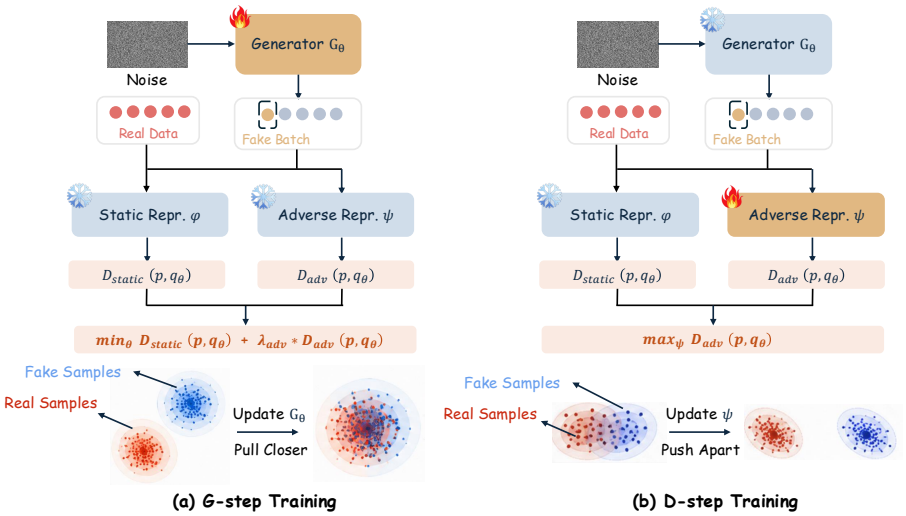


Figure 3: **Overview of our adaptive training.** (a) In the **G-step**, the generator is updated to minimize the static Fréchet objective together with the adaptive Fréchet discrepancy, while both the static and adversarial representations are frozen. (b) In the **D-step**, the generator is frozen, and the adversarial representation is updated to maximize the Fréchet discrepancy between real and generated distributions. The two steps alternately improve the generator and adapt the representation to discrepancies missed by the fixed encoder.

evolves. We therefore introduce a trainable representation ψ_ω, initialized from a pretrained visual encoder, to adapt the comparison space to the current generator.

At iteration t, the current representation ψ_{ω_t} defines an adaptive Fréchet term that complements the static objective. Its calibrated form is introduced below. With ω_t fixed, the generator minimizes

$$\mathcal{L}_t(\theta; \omega_t) = D_{\text{static}}(p, q_\theta) + \lambda_{\text{adv}} D_{\text{adv}}(p, q_\theta; \omega_t). \tag{2}$$

AdvFD alternates between two updates. The generator reduces the discrepancy measured by the current representation, while the adversarial representation increases D_{adv} to reveal remaining differences between the real and generated distributions. This forms a GAN-like game in feature space. The complete training procedure is summarized in Algorithm 1 and Figure 3.

G-step: updating the generator. As shown in Figure 3 (a), the frozen representations in $\mathcal{F}$ and the adversarial representation ψ_{ω_t} are fixed during the G-step. Gradients are propagated only through the generated samples, and the generator is updated as

$$\theta_{t+1} = \theta_t - \eta_{\text{G}}\ \nabla_\theta \left[D_{\text{static}}(p, q_\theta) + \lambda_{\text{adv}} D_{\text{adv}}(p, q_\theta; \omega_t) \right]\big|_{\theta=\theta_t}, \tag{3}$$

where η_{G} is the generator learning rate. The static term maintains alignment in the pretrained feature spaces, while the adaptive term reduces the residual discrepancy exposed by the current learned representation.

D-step: updating the adversarial representation. As shown in Figure 3 (b), the updated generator is frozen during the D-step, and generated samples are detached from the computational graph. We first compute

$$g_t = \nabla_\omega D_{\text{adv}}\big(X_r,\ \text{sg}\big(G_{\theta_{t+1}}(Z)\big); \omega\big)\Big|_{\omega=\omega_t} \tag{4}$$

The representation increment can be equivalently characterized as the solution to the locally constrained response problem

$$\Delta_t = \underset{\|\Delta\|_2 \le \eta_{\text{D}}\tau}{\arg\max} \left\{ \langle g_t, \Delta \rangle - \frac{1}{2\eta_{\text{D}}} \|\Delta\|_2^2 \right\}. \tag{5}$$

This objective maximizes the first-order increase in the adaptive discrepancy while penalizing large departures from the current representation. Its closed-form solution motivates the gradient clipping

used in our D-step:

$$\omega_{t+1} = \omega_t + \eta_{\mathrm{D}} \operatorname{clip}_\tau (g_t), \qquad \operatorname{clip}_\tau(g) = g \min\left(1, \frac{\tau}{\|g\|_2}\right). \tag{6}$$

Here, η_{D} is the representation learning rate and τ is the gradient-norm clipping threshold. We implement this bounded response with AdamW (Loshchilov & Hutter, 2017) under the same clipping threshold. Each D-step selects a bounded local response that increases the real–generated discrepancy around the current representation. The G-step and D-step are applied alternately throughout post-training.

Preventing Feature-Scale Explosion. Directly increasing raw FD in a trainable feature space admits a trivial scale direction. In particular,

$$D_{\mathrm{FD}}^{c\psi_\omega}(p, q_\theta) = c^2 D_{\mathrm{FD}}^{\psi_\omega}(p, q_\theta), \qquad c > 0. \tag{7}$$

Thus, the adversarial representation could increase raw FD simply by enlarging its feature norm, without exposing additional differences between the real and generated distributions.

We remove this coordinate degeneracy through real-feature whitening. Let μ_p^ψ and Σ_p^ψ denote the mean and covariance of the real features under the current representation ψ_ω. We define

$$\bar{\psi}_\omega(x) = \left(\psi_\omega(x) - \mu_p^\psi\right)\left(\Sigma_p^\psi + \epsilon I\right)^{-1/2}, \tag{8}$$

where $\epsilon > 0$ regularizes low-variance directions. In the full-rank population setting with $\epsilon = 0$, affine-equivalent representations are mapped to whitened representations that differ only by an orthogonal transformation, under which FD is invariant. Appendix E provides the detailed derivation.

The adaptive discrepancy used in the G-step and D-step is then defined as

$$D_{\mathrm{adv}}(p, q_\theta; \omega) = D_{\mathrm{FD}}^{\bar{\psi}_\omega}(p, q_\theta). \tag{9}$$

It measures generated-feature deviations relative to the scale and covariance geometry of the real distribution. Under exact whitening, a common rescaling of ψ_ω is canceled by the corresponding change in the real-feature statistics. Therefore, D_{adv} cannot be increased solely through global feature rescaling. Appendix E analyzes the regularized form used in practice. We also describe the details of the training procedure in Section A.

4.2 Relation to GANs

Following the variational view of adversarial training, both GANs and AdvFD optimize a discrepancy through a nested inner–outer problem:

$$\min_\theta \sup_{u \in \mathcal{U}} \Delta_u(p, q_\theta). \tag{10}$$

For a static generated distribution q_θ, the inner optimization adapts u to expose the remaining mismatch between p and q_θ. The outer optimization then updates the generator to reduce the discrepancy identified by the inner player. As q_θ evolves, the inner player is updated accordingly, allowing both methods to provide an adaptive distribution-matching signal.

The key distinction is what the inner optimization learns. In WGAN, the ground cost $c_X(x, y)$ in the original input space is fixed before training. The critic $f(x)$ then assigns a scalar score to each sample and is optimized to give real and generated samples different scores, yielding the Kantorovich dual formulation:

$$W_1(p, q_\theta) = \sup_{\|f\|_{\mathrm{Lip}} \le 1} \left[\mathbb{E}_{x\sim p}[f(x)] - \mathbb{E}_{x\sim q_\theta}[f(x)]\right]. \tag{11}$$

The critic changes these scalar scores, but it does not change the distance between samples or the transport cost used to define the problem. AdvFD uses a different mechanism. It does not assign a scalar score to each sample; instead, the representation network ψ_ω maps each sample to a feature vector. For real and generated samples, AdvFD computes the mean and covariance of these feature vectors and compares the corresponding moment-matched Gaussian distributions using a fixed closed-form W_2^2 expression:

$$D_{\mathrm{FD}}^{\psi_\omega}(p, q_\theta) = W_2^2\Big(\mathcal{N}\big(\mu_p^\psi, \Sigma_p^\psi\big), \mathcal{N}\big(\mu_q^\psi, \Sigma_q^\psi\big)\Big) \tag{12}$$

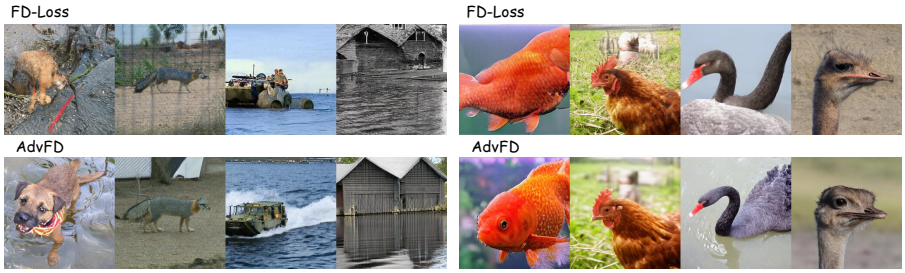


Figure 4: **Qualitative comparison between FD-Loss and AdvFD on ImageNet** 256×256**.** We show representative samples generated by JiT-L (left) and pMF-L (right). Compared with FD-Loss, AdvFD produces cleaner textures, more coherent object structures, and fewer visible artifacts.

where μ_p^ψ and Σ_p^ψ are the mean and covariance of $\psi_\omega(x)$ for real samples, and μ_q^ψ and Σ_q^ψ are the corresponding statistics for generated samples. Learning ψ_ω therefore changes which differences between real and generated samples are reflected in these feature statistics, while the formula used to compare the statistics remains unchanged. In other words, **WGAN uses a fixed geometry with a learned transport potential, whereas AdvFD uses a learned geometry with a fixed Gaussian transport functional.**

5 Experiment

5.1 Settings

Datasets and Training Protocol. We evaluate AdvFD on class-conditional image generation using ImageNet-1K (Russakovsky et al., 2015) at a resolution of 256×256. Specifically, we consider JiT (Li & He, 2026) and pixel MeanFlow (pMF) (Lu et al., 2026) as generators, including both B-, L- and H-scale variants. Following the post-training protocol of FD-loss (Yang et al., 2026a), all methods are initialized from the same pretrained checkpoints and optimized on the ImageNet-1K training set. Unless otherwise specified, AdvFD uses the same static Fréchet representations as its FD-loss baseline and introduces only the additional adversarial representation. During evaluation, all one-step models generate 50,000 samples using identical sampling settings to ensure a fair comparison.

Evaluation Metrics. We report FID, FD-r6, and FD-r3. Following the evaluation protocol of FD-loss (Yang et al., 2026a), FID measures distributional discrepancy in the Inception feature space, while FD-r6 averages normalized Fréchet distance ratios across six visual representations: Inception (Szegedy et al., 2015), ConvNeXt (Liu et al., 2022b), DINOv2 (Oquab et al., 2023), MAE (He et al., 2022), SigLIP (Zhai et al., 2023), and CLIP (Radford et al., 2021). Since the SIM (**S**igLIP+**I**nception+**M**AE) objective used for training includes SigLIP, MAE, and Inception, we additionally report FD-r3 over the remaining three encoders, namely ConvNeXt, DINOv2, and CLIP. FD-r3 therefore evaluates whether improvements in the optimized feature spaces transfer to representations not used during training. Lower values indicate better distribution matching.

5.2 Main Results on C2I

Quantitative Results. Tables 1 and 4 compare AdvFD with representative generative models on ImageNet 256×256. AdvFD consistently improves upon FD-loss across all JiT and pMF scales (B/L/H) while preserving one-step generation. On JiT-B, it reduces FID/FD-r6/FD-r3 from 1.00/5.53/8.45 to 0.79/3.92/6.03. On JiT-L, AdvFD achieves 0.73/2.01/3.20, with relative reductions of 38.0% and 41.4% in FD-r6 and FD-r3, respectively. These gains persist on JiT-H, improving 0.75/2.65/4.44 to 0.72/1.80/2.93. AdvFD also improves all three metrics across pMF-B/L/H, reaching 0.74/1.74/2.50 on pMF-H compared with 0.77/1.89/2.69 for FD-loss. The consistent FD-r3 reductions indicate that these gains extend to representations excluded from training, supporting generalization across feature spaces, generator architectures, and model scales.

Table 1: **Class-conditional generation on ImageNet** 256×256**.** We compare the original models, FD-loss, and AdvFD on JiT and pMF at B, L, and H scales

Method	NFE	#Params	FID ↓	FD-r6 ↓	FD-r3 ↓
JiT-B	$50\times2\times2^{\dagger}$	131M	3.71	15.65	15.06
+ FD-loss	1	131M	1.00	5.53	8.45
+ AdvFD (Ours)	1	131M	**0.79**	**3.92**	**6.03**
JiT-L	$50\times2\times2^{\dagger}$	459M	2.59	10.73	10.27
+ FD-loss	1	459M	0.77	3.24	5.46
+ AdvFD (Ours)	1	459M	**0.73**	**2.01**	**3.20**
JiT-H	$50\times2\times2^{\dagger}$	953M	1.97	7.66	9.07
+ FD-loss	1	953M	0.75	2.65	4.44
+ AdvFD (Ours)	1	953M	**0.72**	**1.80**	**2.93**
pMF-B	1	118M	3.31	13.70	11.82
+ FD-loss	1	118M	0.85	3.50	4.49
+ AdvFD (Ours)	1	118M	**0.81**	**3.32**	**4.22**
pMF-L	1	410M	2.72	9.09	7.62
+ FD-loss	1	410M	0.78	2.09	2.72
+ AdvFD (Ours)	1	410M	**0.77**	**1.89**	**2.57**
pMF-H	1	935M	2.29	6.87	6.09
+ FD-loss	1	935M	0.77	1.89	2.69
+ AdvFD (Ours)	1	935M	**0.74**	**1.74**	**2.50**

Table 2: **Alternative anti-hacking strategies and whitening ablation.** All experiments use JiT-B with the SIM FD objective. Lower is better for all metrics.

Method	FID ↓	FD-r6 ↓	FD-r3 ↓
FD-loss	1.00	5.53	8.45
FD-loss + PatchGAN	0.90	5.57	8.90
FD-loss + DMD	1.77	5.90	6.75
AdvFD	**0.79**	**3.92**	**6.03**

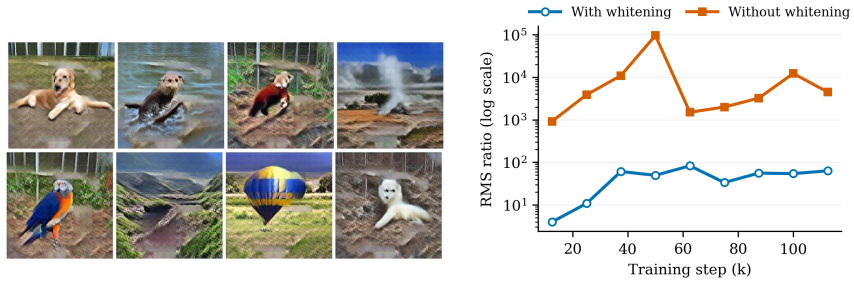
Method	Branch	FID ↓	FD-r6 ↓	FD-r3 ↓
FD-loss	–	1.00	5.53	8.45
AdvFD	None	10.69	58.50	40.54
	Static	13.49	28.32	21.66
	Adaptive	**0.79**	**3.92**	**6.03**

Qualitative Results. Figure 4 provides qualitative comparisons on representative JiT-L and pMF-L backbones. Across both architectures, AdvFD produces more coherent object structures, cleaner textures, and fewer local artifacts than FD-Loss. These qualitative improvements are consistent with the quantitative gains, demonstrating that AdvFD generalizes across different generators.

5.3 Ablation Study

Ablation on Anti-Hacking Strategies. We compare AdvFD with several alternative remedies for mitigating Fréchet hacking, including external PatchGAN (Isola et al., 2017) and DMD-style (Yin et al., 2024b) guidance. Motivated by ASD (Wei et al., 2024) and VSD (Wang et al., 2023), DMD actually adopts a GAN-like adversarial learning scheme, where a learnable critic is introduced to provide adaptive feedback against artifacts overlooked by the fixed distillation objective. These methods introduce additional realism signals outside the original FD objective, but they do not necessarily improve Fréchet alignment when evaluated with feature encoders not used for training. As shown in Table 2 (a), external PatchGAN even degrades FD-r6 and FD-r3 with the FD-loss baseline, while DMD hurts FID and brings limited gains on held-out FD-r3. In contrast, AdvFD with an Inception-initialized adversarial representation substantially improves FD-r6 and FD-r3, suggesting that calibrated adversarial representations better expose blind spots of static FD objectives than external score-based remedies.

Ablation on Whitening Location. Table 2(b) shows that whitening the adversarial representation is essential for stable training. Without whitening, FID, FD-r6, and FD-r3 deteriorate to 10.69, 58.50, and 40.54, respectively. Whitening only the static representations offers limited stabilization because the trainable branch remains unconstrained. In contrast, whitening the adversarial representation



(a) Degraded Samples from Training without Whitening

(b) Whitening Prevents Adversarial Feature Norm Inflation

Figure 5: **Effect of feature whitening on adversarial feature stability.** (a) Omitting whitening leads to severe degradation in generated samples. (b) The ratio between the pre-whitening RMS norms of the adaptive and frozen-reference features over training, shown on a logarithmic scale. Real-feature whitening removes the direct incentive to increase the adaptive Fréchet discrepancy through global feature scaling, thereby substantially suppressing feature-norm inflation in practice.

Table 3: **Ablation on adversarial representation backbones and initialization.** We keep the static training objective fixed to SIM FD and vary the backbone and initialization of the adversarial representation ψ_ω. "Pretrained" denotes initialization from the corresponding pretrained visual encoder, while "Random" uses the same architecture without loading pretrained weights. Lower is better for all metrics. Best results are shown in **bold**, and second-best results are underlined.

Method	Adv. Repr.	Initialization	FID ↓	FD-r6 ↓	FD-r3 ↓
FD-loss (Baseline)	–	–	1.00	5.53	8.45
AdvFD	SigLIP	Pretrained	0.89	4.55	7.40
AdvFD	MAE	Pretrained	0.90	4.73	7.42
AdvFD	Inception	Random	0.85	5.69	7.63
AdvFD	Inception	Pretrained	**0.79**	**3.92**	**6.03**

suppresses feature-scale inflation and improves all metrics over FD-Loss. Figure 5 further supports this finding: panel (a) shows severe sample degradation without whitening, while panel (b) shows that the pre-whitening RMS norm of adaptive features grows by orders of magnitude relative to the frozen reference. With whitening, this ratio remains lower and more stable, indicating that whitening limits Fréchet discrepancy inflation through global feature scaling.

Ablation on adversarial representation backbones and initialization. Table 3 varies the backbone and initialization of ψ_ω while keeping the static SIM-FD objective unchanged. SigLIP and MAE are adapted with rank-16 LoRA (Hu et al., 2021) due to memory constraints, whereas Inception is fully fine-tuned. Pretrained Inception performs best across all metrics, providing a well-conditioned feature space that lets the D-step target residual real–generated discrepancies. A randomly initialized encoder must first learn basic visual features and more readily increases FD through low-level shortcuts. SigLIP and MAE may be less sensitive to local textures and artifacts due to patchification and global aggregation. This does not contradict Section 3, which identifies Inception as the most exploitable space: a frozen encoder retains a fixed blind subspace, whereas the D-step adapts the comparison geometry as q_θ evolves, reweighting exploitable directions.

6 Conclusion

We presented **AdvFD**, a calibrated adversarial representation training framework that improves the robustness of Fréchet-based distribution matching under static representation targets. By adaptively exposing residual discrepancies overlooked by static feature spaces, AdvFD improves one-step ImageNet generation across different backbones while preserving the same 1-NFE sampling cost. AdvFD consistently enhances both the optimized Fréchet metrics and those measured in feature spaces not used during training, demonstrating improved robustness beyond the target representation. We hope this work encourages further exploration of adaptive comparison spaces for reliable distribution-level training and its extension to larger-scale, multimodal, and video generation.

AI USE STATEMENT

We used GPT-5.6 solely to improve the grammar, clarity, and readability of the manuscript. We did not use generative AI for any other research or manuscript preparation tasks. All AI-assisted edits were reviewed by the authors to ensure that they accurately preserved the intended technical meaning. The authors take full responsibility for the final content of this paper.

REPRODUCIBILITY STATEMENT

The proposed method is described in Section 4. Details of the datasets, preprocessing procedures, evaluation protocols, and experimental settings are provided in Section 5 and Appendix H. These details are included to facilitate reproduction of the reported results.

REFERENCES

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

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 214–223. JMLR.org, 2017.

Quentin Berthet, Yu-Han Wu, Clement Crepy, Romuald Elie, Klaus Greff, and Michael Eli Sander. Mind: Monge inception distance for generative models evaluation. *arXiv preprint arXiv:2605.06797*, 2026.

Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Mingyang Deng, He Li, Tianhong Li, Yilun Du, and Kaiming He. Generative modeling via drifting, 2026. URL https://arxiv.org/abs/2602.04770.

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

Khoa D Doan, Saurav Manchanda, Fengjiao Wang, Sathiya Keerthi, Avradeep Bhowmik, and Chandan K Reddy. Image generation via minimizing fr\'echet distance in discriminator feature space. *arXiv preprint arXiv:2003.11774*, 2020.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pp. 10835–10866. PMLR, 2023.

Zhengyang Geng, Yiyang Lu, Zongze Wu, Eli Shechtman, J Zico Kolter, and Kaiming He. Improved mean flows: On the challenges of fastforward generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 30467–30476, 2026.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

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

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

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9307–9315, 2024.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

Kyuyoung Kim, Jongheon Jeong, Minyong An, Mohammad Ghavamzadeh, Krishnamurthy Dvijotham, Jinwoo Shin, and Kimin Lee. Confidence-aware reward optimization for fine-tuning text-to-image models. *arXiv preprint arXiv:2404.01863*, 2024.

Nupur Kumari, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Ensembling off-the-shelf models for gan training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10651–10662, 2022.

Xingjian Leng, Jaskirat Singh, Yunzhong Hou, Zhenchang Xing, Saining Xie, and Liang Zheng. Repa-e: Unlocking vae for end-to-end tuning with latent diffusion transformers, 2025. URL https://arxiv.org/abs/2504.10483.

Chun-Liang Li, Wei-Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. Mmd gan: Towards deeper understanding of moment matching network. *Advances in neural information processing systems*, 30, 2017.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36115–36125, 2026.

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization, 2024. URL https://arxiv.org/abs/2406.11838.

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

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

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11966–11976. IEEE, 2022b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Yanzuo Lu, Yuxi Ren, Xin Xia, Shanchuan Lin, Xing Wang, Xuefeng Xiao, Andy J Ma, Xiaohua Xie, and Jian-Huang Lai. Adversarial distribution matching for diffusion distillation towards efficient image and video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16818–16829, 2025.

Yiyang Lu, Susie Lu, Qiao Sun, Hanhong Zhao, Zhicheng Jiang, Xianbang Wang, Tianhong Li, Zhengyang Geng, and Kaiming He. One-step latent-free image generation with pixel mean flows. *arXiv preprint arXiv:2601.22158*, 2026.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pp. 23–40. Springer, 2024.

Weijia Mao, Hao Chen, Zhenheng Yang, and Mike Zheng Shou. The image as its own reward: Reinforcement learning with adversarial reward for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5999–6009, 2026.

Youssef Mroueh and Tom Sercu. Fisher gan. *Advances in neural information processing systems*, 30, 2017.

Youssef Mroueh, Tom Sercu, and Vaibhava Goel. Mcgan: Mean and covariance feature matching gan. In *International conference on machine learning*, pp. 2527–2535. PMLR, 2017.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Alexander Pan, Kush Bhatia, and Jacob Steinhardt. The effects of reward misspecification: Mapping and mitigating misaligned models. *arXiv preprint arXiv:2201.03544*, 2022.

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

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Sucheng Ren, Qihang Yu, Ju He, Xiaohui Shen, Alan Yuille, and Liang-Chieh Chen. Flowar: Scale-wise autoregressive image generation meets flow matching, 2024. URL `https://arxiv.org/abs/2412.15205`.

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*, pp. 10684–10695, 2022.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger. Projected gans converge faster. *Advances in Neural Information Processing Systems*, 34:17480–17492, 2021.

Axel Sauer, Frederic Boesel, Tim Dockhorn, Andreas Blattmann, Patrick Esser, and Robin Rombach. Fast high-resolution image synthesis with latent adversarial diffusion distillation. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024a.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024b.

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

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

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction, 2024. URL https://arxiv.org/abs/2404.02905.

Shuai Wang, Zhi Tian, Weilin Huang, and Limin Wang. Ddt: Decoupled diffusion transformer, 2025. URL https://arxiv.org/abs/2504.05741.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in neural information processing systems*, 36:8406–8441, 2023.

Min Wei, Jingkai Zhou, Junyao Sun, and Xuesong Zhang. Adversarial score distillation: When score distillation meets gan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8131–8141, 2024.

Ge Wu, Shen Zhang, Ruijing Shi, Shanghua Gao, Zhenyuan Chen, Lei Wang, Zhaowei Chen, Hongcheng Gao, Yao Tang, Jian Yang, Ming-Ming Cheng, and Xiang Li. Representation entanglement for generation: Training diffusion transformers is much easier than you think, 2025. URL https://arxiv.org/abs/2507.01467.

Jiawei Yang, Zhengyang Geng, Xuan Ju, Yonglong Tian, and Yue Wang. Representation fr\'echet loss for visual generation. *arXiv preprint arXiv:2604.28190*, 2026a.

Jiawei Yang, Tianhong Li, Lijie Fan, Yonglong Tian, and Yue Wang. Latent denoising makes good tokenizers, 2026b. URL https://arxiv.org/abs/2507.15856.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models, 2025. URL https://arxiv.org/abs/2501.01423.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024b.

Qihang Yu, Qihao Liu, Ju He, Xinyang Zhang, Yang Liu, Liang-Chieh Chen, and Xi Chen. Autoregressive image generation with masked bit modeling, 2026. URL https://arxiv.org/abs/2602.09024.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think, 2025. URL https://arxiv.org/abs/2410.06940.

Kevin Zhai, Utsav Singh, Anirudh Thatipelli, Souradip Chakraborty, Anit Kumar Sahu, Furong Huang, Amrit Singh Bedi, and Mubarak Shah. Mira: Towards mitigating reward hacking in inference-time alignment of t2i diffusion models. *arXiv preprint arXiv:2510.01549*, 2025.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2510.11690*, 2025.

Algorithm 1 AdvFD Training Procedure

Require: Pretrained generator G_θ, frozen representations $\mathcal{F}$, pretrained adversarial representation ψ_ω, real distribution p, adaptive weight λ_{adv}, learning rates η_{G} and η_{D}, clipping threshold τ, and training iterations T

for $t = 0, \ldots, T-1$ **do**
  Sample a real batch $X_r \sim p$ and noise $Z \sim p_Z$
  Generate $X_g \leftarrow G_{\theta_t}(Z)$

  G-step: update the generator
  Freeze the frozen representations $\mathcal{F}$ and adversarial representation ψ_{ω_t}
  Compute $D_{\text{static}} \leftarrow \sum_{\phi \in \mathcal{F}} \lambda_\phi D^{\phi}_{\text{FD}}(X_r, X_g)$
  Compute $D_{\text{adv}} \leftarrow D^{\bar{\psi}_{\omega_t}}_{\text{FD}}(X_r, X_g)$ using real-feature whitening
  $\theta_{t+1} \leftarrow \theta_t - \eta_{\text{G}} \nabla_\theta \left[D_{\text{static}} + \lambda_{\text{adv}} D_{\text{adv}} \right]$

  D-step: update the adversarial representation
  Freeze $G_{\theta_{t+1}}$ and detach the generated samples
  $X_g^{\text{det}} \leftarrow \text{sg}\big[G_{\theta_{t+1}}(Z)\big]$
  Recompute $D_{\text{adv}} \leftarrow D^{\bar{\psi}_{\omega_t}}_{\text{FD}}(X_r, X_g^{\text{det}})$
  $g_t \leftarrow \nabla_\omega D_{\text{adv}}$
  $\omega_{t+1} \leftarrow \omega_t + \eta_{\text{D}}\, g_t \min\left(1, \frac{\tau}{\|g_t\|_2}\right)$
end for
return post-trained generator G_{θ_T}

A Detailed Training Procedure

Algorithm 1 summarizes the alternating training procedure of AdvFD. During the G-step, both the static and adversarial representations are frozen, while the generator is updated to minimize the corresponding Fréchet discrepancies. This step pulls the generated distribution closer to the real distribution in the current feature spaces, as illustrated in Figure 3 (a). During the D-step, the generator is frozen, and the adversarial representation is updated to maximize the Fréchet discrepancy, effectively pushing the real and generated feature distributions apart to expose residual differences that are not captured by the static representations, as shown in Figure 3 (b). By alternating between pulling the distributions together and adaptively pushing them apart, AdvFD provides the generator with a continuously evolving distribution-level training signal.

B System Comparison

As shown in Table 4, AdvFD consistently improves upon FD-loss across all six JiT and pMF variants on FID, FD-r6, and FD-r3, while retaining single-evaluation generation. JiT-H with AdvFD achieves the lowest FID of 0.72 among the compared generative models, whereas pMF-H with AdvFD obtains the best FD-r6 and FD-r3 scores of 1.74 and 2.50, respectively. Both models outperform strong baselines, including BAR-L, REPA-E, and RAE-XL, across all three metrics. The improvements also extend to representations excluded from training: AdvFD reduces FD-r3 from 4.44 to 2.93 on JiT-H and from 2.69 to 2.50 on pMF-H compared with FD-loss. These results demonstrate consistent gains across model families and scales, with improvements that generalize beyond the optimized representation set.

C Human Study

We conduct a human preference study using 1,000 randomly sampled images and 10 evaluators with diverse expertise. As shown in Figure 6, our method is consistently preferred over the corresponding FD-Loss baselines across both model families and scales. Specifically, our method achieves preference rates of 71.2% and 63.9% on JiT-L and JiT-H, respectively, and 60.1% and 64.6% on pMF-L and pMF-H, exceeding 50% in all four comparisons.

Table 4: **Class-conditional generation on ImageNet** 256×256**.** We compare AdvFD with discrete-space, latent-space, and pixel-space generative models. FD-loss and AdvFD use the SIM representations, including SigLIP, Inception, and MAE. FID and FD-r6 follow the FD-loss evaluation protocol, while FD-r3 excludes the SIM representations used during training to evaluate generalization beyond the optimized representation set. Lower is better for all metrics. †For methods using interval CFG, we report the full-CFG NFE upper bound following FD-Loss.

Method	NFE	Space	#Params	FID ↓	FD-r6 ↓	FD-r3 ↓
reference (real images)						
50k validation images	N/A	N/A	N/A	1.68	1.00	1.00
discrete-space models						
VAR-d30 (Tian et al., 2024)	10×2	discrete	2B	1.97	6.70	6.77
BAR-L (Yu et al., 2026)	$256\times2\times4$	discrete	1.1B	**1.01**	**3.57**	**3.35**
latent-space models, multi-step without semantic distillation						
SiT-XL/2 (Ma et al., 2024)	250×2	latent	675M	2.12	8.44	9.20
MAR-L (Li et al., 2024)	$256\times2\times100$	latent	478M	1.80	6.68	7.32
FlowAR-H (Ren et al., 2024)	$50\times2^{\dagger}$	latent	1.9B	1.68	6.13	6.16
MAR-H (Li et al., 2024)	$256\times2\times100$	latent	942M	1.56	5.61	6.31
MAR-L, DeTok (Yang et al., 2026b)	$256\times2\times100$	latent	478M	**1.39**	**5.49**	**6.05**
with semantic distillation						
REG (Wu et al., 2025)	$250\times2^{\dagger}$	latent	685M	1.54	4.64	5.15
SiT-XL/2-REPA (Yu et al., 2025)	$250\times2^{\dagger}$	latent	675M	1.42	5.45	6.05
LightningDiT (Yao et al., 2025)	250×2	latent	675M	1.42	4.57	5.02
DDT-XL (Wang et al., 2025)	250×2	latent	675M	1.26	5.70	6.38
REPA-E (Leng et al., 2025)	$250\times2^{\dagger}$	latent	676M	1.17	**3.04**	**3.33**
RAE-XL (Zheng et al., 2025)	$50\times2^{\dagger}$	latent	839M	**1.16**	3.26	3.92
latent-space models, one-step						
Drift-L (latent) (Deng et al., 2026)	1	latent	463M	1.53	10.92	11.32
iMF-XL (Geng et al., 2026)	1	latent	610M	1.82	8.39	8.72
iMF-XL (Geng et al., 2026)	2	latent	610M	1.61	7.48	**7.79**
pixel-space models, multi-step backbones						
JiT-B (Li & He, 2026)	$50\times2\times2^{\dagger}$	pixel	131M	3.71	15.65	15.06
+ FD-loss (Yang et al., 2026a)	1	pixel	131M	1.00	5.53	8.45
+ AdvFD (Ours)	1	pixel	131M	**0.79**	**3.92**	**6.03**
JiT-L (Li & He, 2026)	$50\times2\times2^{\dagger}$	pixel	459M	2.59	10.73	10.27
+ FD-loss (Yang et al., 2026a)	1	pixel	459M	0.77	3.24	5.46
+ AdvFD (Ours)	1	pixel	459M	**0.73**	**2.01**	**3.20**
JiT-H (Li & He, 2026)	$50\times2\times2^{\dagger}$	pixel	953M	1.97	7.66	9.07
+ FD-loss (Yang et al., 2026a)	1	pixel	953M	0.75	2.65	4.44
+ AdvFD (Ours)	1	pixel	953M	**0.72**	**1.80**	**2.93**
pixel-space models, one-step						
Drift-L (pixel) (Deng et al., 2026)	1	pixel	465M	1.43	10.51	11.18
pMF-B (Lu et al., 2026)	1	pixel	118M	3.31	13.70	11.82
+ FD-loss (Yang et al., 2026a)	1	pixel	118M	0.85	3.50	4.49
+ AdvFD (Ours)	1	pixel	118M	**0.81**	**3.32**	**4.22**
pMF-L (Lu et al., 2026)	1	pixel	410M	2.72	9.09	7.62
+ FD-loss (Yang et al., 2026a)	1	pixel	410M	0.78	2.09	2.72
+ AdvFD (Ours)	1	pixel	410M	**0.77**	**1.89**	**2.57**
pMF-H (Lu et al., 2026)	1	pixel	935M	2.29	6.87	6.09
+ FD-loss (Yang et al., 2026a)	1	pixel	935M	0.77	1.89	2.69
+ AdvFD (Ours)	1	pixel	935M	**0.74**	**1.74**	**2.50**

D Finite-Horizon Boundedness of Representation Adaptation

We briefly show that the finite, gradient-clipped AdamW (Loshchilov & Hutter, 2017) updates used in AdvFD remain within a bounded neighborhood of their initialization. Let $\tilde{g}^{(k)} = \mathrm{clip}_{\tau}(g^{(k)})$, such that $\|\tilde{g}^{(k)}\|_2 \le \tau$. Starting from zero-initialized moments, the bias-corrected first moment of AdamW is a convex combination of the clipped gradients and therefore satisfies $\|\hat{m}^{(k)}\|_2 \le \tau$. Since $\hat{v}^{(k)}$ is elementwise nonnegative and $\varepsilon_{\mathrm{A}} > 0$, the preconditioned update direction is bounded as $\|u^{(k)}\|_2 \le B_{\mathrm{A}} \coloneqq \tau/\varepsilon_{\mathrm{A}}$.

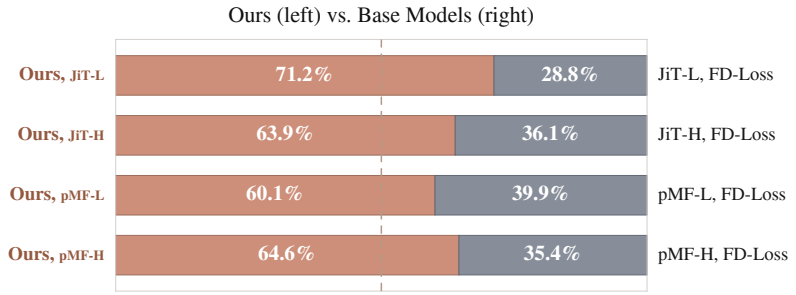


Figure 6: Human preference comparison between our method and the corresponding FD-Loss baselines. Orange and gray bars indicate preference rates for our method and the baselines, respectively. The dashed line marks equal preference at 50%.

For the AdamW update $\omega^{(k+1)} = (1 - \eta_{\mathrm{D}}\lambda_{\mathrm{wd}})\omega^{(k)} + \eta_{\mathrm{D}}u^{(k+1)}$, where $0 \le \eta_{\mathrm{D}}\lambda_{\mathrm{wd}} \le 1$, unrolling the recursion gives, for every $k \le K$,

$$\left\|\omega^{(k)} - \omega^{(0)}\right\|_2 \le \rho_K^{\mathrm{AdamW}} := \begin{cases} \left[1 - (1 - \eta_{\mathrm{D}}\lambda_{\mathrm{wd}})^K\right]\left(\left\|\omega^{(0)}\right\|_2 + \dfrac{B_{\mathrm{A}}}{\lambda_{\mathrm{wd}}}\right), & \lambda_{\mathrm{wd}} > 0, \\ K\eta_{\mathrm{D}}B_{\mathrm{A}}, & \lambda_{\mathrm{wd}} = 0. \end{cases} \tag{13}$$

Thus, the first K iterates lie in a compact reachable set. Assuming compact input support and continuity of the representation network, the regularized adaptive Fréchet discrepancy is continuous on this set and therefore attains a finite maximum.

This result establishes only the finite-horizon boundedness of the implemented optimizer trajectory. It does not eliminate the scale degeneracy of raw Fréchet distance, which is addressed by real-feature whitening.

E Derivation and Affine Invariance of Real-Feature Whitening

This appendix provides the detailed justification for the real-feature whitening used in Section 4.1. We first explain why the adaptive discrepancy should be invariant to invertible affine reparameterizations, then derive real-feature whitening as a solution to this requirement, and finally quantify the approximation error introduced by covariance regularization.

Why affine invariance is required. Let $\psi_\omega(x) \in \mathbb{R}^d$ denote the row-vector output of the adversarial representation. An invertible affine reparameterization

$$\psi'_\omega(x) = \psi_\omega(x)A + b, \qquad A \in \mathrm{GL}(d), \qquad b \in \mathbb{R}^d, \tag{14}$$

changes the origin, scale, and axes of the feature coordinates without changing the information represented by ψ_ω. An adaptive discrepancy should therefore assign the same value to affine-equivalent representations:

$$\mathcal{D}\left(p, q_\theta; \psi_\omega A + b\right) = \mathcal{D}\left(p, q_\theta; \psi_\omega\right). \tag{15}$$

Raw Fréchet distance does not satisfy this requirement because it depends on the Euclidean coordinates through the feature mean and covariance. In particular, feature scaling is obtained by setting $A = cI$ and $b = 0$. For any $c > 0$, the corresponding feature statistics satisfy

$$\mu_r^{c\psi} = c\mu_r^{\psi}, \qquad \Sigma_r^{c\psi} = c^2\Sigma_r^{\psi}, \qquad r \in \{p, q_\theta\}, \tag{16}$$

and hence

$$D_{\mathrm{FD}}^{c\psi_\omega}\left(p, q_\theta\right) = c^2 D_{\mathrm{FD}}^{\psi_\omega}\left(p, q_\theta\right). \tag{17}$$

Whenever the original discrepancy is nonzero,

$$\sup_{c>0} D_{\mathrm{FD}}^{c\psi_\omega}\left(p, q_\theta\right) = +\infty. \tag{18}$$

Thus, increasing raw FD can be achieved by continually expanding the feature coordinates, without exposing any additional difference between the represented distributions. Equation 15 removes this coordinate-dependent direction from the adaptive objective.

Real-feature whitening as a solution. Let

$$\mu_p^{\psi} = \mathbb{E}_{x\sim p}\left[\psi_\omega(x)\right], \qquad \Sigma_p^{\psi} = \mathrm{Cov}_{x\sim p}\left[\psi_\omega(x)\right] \tag{19}$$

denote the population mean and covariance of the real features. We consider a standardized representation of the form

$$\bar{\psi}_\omega(x) = \left(\psi_\omega(x) - \mu_p^{\psi}\right) W_\omega. \tag{20}$$

Centering removes the dependence on the coordinate origin. To remove the scale and covariance geometry, we require the transformed real covariance to be identity:

$$W_\omega^\top \Sigma_p^{\psi} W_\omega = I. \tag{21}$$

When $\Sigma_p^{\psi} \succ 0$, every solution of Equation 21 can be written as

$$W_\omega = \left(\Sigma_p^{\psi}\right)^{-1/2} R, \qquad R^\top R = I. \tag{22}$$

The remaining matrix R is an arbitrary orthogonal rotation. Because FD is invariant to a common orthogonal transformation, all such solutions induce the same discrepancy. We therefore select the symmetric whitener $R = I$:

$$\bar{\psi}_\omega(x) = \left(\psi_\omega(x) - \mu_p^{\psi}\right) \left(\Sigma_p^{\psi}\right)^{-1/2}. \tag{23}$$

The resulting real-feature statistics are

$$\mu_p^{\bar{\psi}} = 0, \qquad \Sigma_p^{\bar{\psi}} = I. \tag{24}$$

It remains to verify that this construction satisfies Equation 15. Under the affine transformation in Equation 14, the real-feature statistics become

$$\mu_p' = \mu_p^{\psi} A + b, \qquad \Sigma_p' = A^\top \Sigma_p^{\psi} A. \tag{25}$$

Whitening the transformed representation using its own real statistics gives

$$\begin{aligned} \bar{\psi}'_\omega(x) &= \left(\psi'_\omega(x) - \mu_p'\right) \left(\Sigma_p'\right)^{-1/2} && (26)\\ &= \left(\psi_\omega(x) - \mu_p^{\psi}\right) A \left(A^\top \Sigma_p^{\psi} A\right)^{-1/2} && (27)\\ &= \bar{\psi}_\omega(x) Q_A, && (28) \end{aligned}$$

where

$$Q_A = \left(\Sigma_p^{\psi}\right)^{1/2} A \left(A^\top \Sigma_p^{\psi} A\right)^{-1/2}. \tag{29}$$

This matrix is orthogonal because

$$Q_A^\top Q_A = \left(A^\top \Sigma_p^{\psi} A\right)^{-1/2} \left(A^\top \Sigma_p^{\psi} A\right) \left(A^\top \Sigma_p^{\psi} A\right)^{-1/2} = I. \tag{30}$$

Thus, affine-equivalent raw representations become orthogonally equivalent after real-feature whitening. Since FD is invariant to a common orthogonal transformation,

$$D_{\mathrm{FD}}^{\bar{\psi}'_\omega}\left(p, q_\theta\right) = D_{\mathrm{FD}}^{\bar{\psi}_\omega}\left(p, q_\theta\right), \tag{31}$$

which proves Equation 15. For the scaling transformation $A = cI$ with $c > 0$, $Q_A = I$, so the whitened features themselves remain unchanged. Consequently, the adversarial representation cannot increase the whitened discrepancy merely by expanding its global coordinate scale.

Under exact whitening, the adaptive discrepancy can be written as

$$D_{\mathrm{adv}}\left(p, q_\theta; \omega\right) = \left\|\mu_q^{\bar{\psi}}\right\|_2^2 + \mathrm{Tr}\left[I + \Sigma_q^{\bar{\psi}} - 2\left(\Sigma_q^{\bar{\psi}}\right)^{1/2}\right], \tag{32}$$

where $\mu_q^{\bar{\psi}}$ and $\Sigma_q^{\bar{\psi}}$ are the mean and covariance of the whitened generated features.

Approximation error introduced by covariance regularization. Exact whitening assumes population statistics and $\Sigma_p^\psi \succ 0$. In practice, the real-feature statistics are estimated from a minibatch, and the empirical covariance may be ill-conditioned or rank deficient. We therefore use

$$\bar{\psi}_{\omega,\epsilon}(x) = \left(\psi_\omega(x) - \hat{\mu}_p^\psi\right)\left(\hat{\Sigma}_p^\psi + \epsilon I\right)^{-1/2}, \qquad \epsilon > 0. \tag{33}$$

The ridge makes the inverse covariance square root well defined and bounds its operator norm:

$$\left\|\left(\hat{\Sigma}_p^\psi + \epsilon I\right)^{-1/2}\right\|_2 \le \epsilon^{-1/2}. \tag{34}$$

To quantify the deviation from exact whitening, let

$$\hat{\Sigma}_p^\psi = U\,\mathrm{diag}\left(\lambda_1, \ldots, \lambda_d\right) U^\top. \tag{35}$$

The covariance of the regularized-whitened real features is

$$\Sigma_{p,\epsilon}^{\bar{\psi}} = \left(\hat{\Sigma}_p^\psi + \epsilon I\right)^{-1/2} \hat{\Sigma}_p^\psi \left(\hat{\Sigma}_p^\psi + \epsilon I\right)^{-1/2} \tag{36}$$

$$= U\,\mathrm{diag}\left(\frac{\lambda_1}{\lambda_1+\epsilon}, \ldots, \frac{\lambda_d}{\lambda_d+\epsilon}\right) U^\top. \tag{37}$$

Exact whitening would map every nonzero covariance eigenvalue to one. With regularization, the error along the i-th eigenvector is

$$\left|1 - \frac{\lambda_i}{\lambda_i+\epsilon}\right| = \frac{\epsilon}{\lambda_i+\epsilon}. \tag{38}$$

Therefore, on any covariance subspace satisfying $\lambda_i \ge \lambda_{\min} > 0$,

$$\left\|\Sigma_{p,\epsilon}^{\bar{\psi}} - I\right\|_2 \le \frac{\epsilon}{\lambda_{\min}+\epsilon} \le \frac{\epsilon}{\lambda_{\min}}. \tag{39}$$

The relative change in the whitening multiplier is similarly bounded by

$$1 - \frac{(\lambda_i+\epsilon)^{-1/2}}{\lambda_i^{-1/2}} = 1 - \sqrt{\frac{\lambda_i}{\lambda_i+\epsilon}} \le \frac{\epsilon}{2\lambda_i}. \tag{40}$$

Thus, the regularized transformation differs from exact whitening by $\mathcal{O}(\epsilon/\lambda_{\min})$ on well-conditioned covariance directions. Directions with very small empirical eigenvalues incur a larger approximation error but are intentionally regularized to prevent unstable inverse scaling.

We use $\epsilon = 10^{-3}$ in all experiments. Equation 38 gives a covariance error below 1% for directions with $\lambda_i \ge 0.1$ and below 0.1% for directions with $\lambda_i \ge 1$. Because $\epsilon > 0$ breaks exact affine invariance, the implemented transformation should be viewed as a regularized approximation to the canonicalization derived above. Exact affine invariance is recovered in the full-rank limit as $\epsilon \to 0$.

Deviation from detached EMA statistics. The cancellation above further assumes that whitening uses the real-feature statistics of the current ψ_ω. We estimate them with an EMA that is detached from the D-step gradient, so an instantaneous rescaling of ψ_ω is not cancelled within the same step and is corrected only once the EMA catches up. The resulting deviation is governed by the drift of Σ_p^ψ over that timescale, which is small in our setting because $\eta_D \le 2 \times 10^{-6}$ and ψ_ω is updated only every two generator steps. Together with the $O(\epsilon/\lambda_{\min})$ term above, invariance therefore holds up to a controlled approximation rather than exactly, which is consistent with the bounded residual growth in Figure 5 (b).

F Transport-Based Relation to GANs

We provide a transport-based interpretation of the relation between conventional adversarial training, WGAN, and AdvFD. The central distinction is whether the inner optimization learns a function under a predefined distribution geometry or learns the geometry in which the distributions are compared.

From GAN discrimination to distribution transport. The original GAN (Goodfellow et al., 2020) optimizes

$$\min_{\theta} \sup_{D} \left[\mathbb{E}_{x\sim p} \log D(x) + \mathbb{E}_{x\sim q_\theta} \log\left(1 - D(x)\right) \right]. \tag{41}$$

For a static generated distribution q_θ, the optimal discriminator is

$$D^\star(x) = \frac{p(x)}{p(x) + q_\theta(x)}. \tag{42}$$

Substituting $D^\star$ into Equation 41 gives

$$\sup_{D} V\left(D; p, q_\theta\right) = -\log 4 + 2D_{\mathrm{JS}}\left(p \| q_\theta\right). \tag{43}$$

Thus, the discriminator can be analytically eliminated, and the resulting objective measures distributional overlap through the Jensen–Shannon divergence. This formulation does not introduce a cost describing how probability mass should move between distinct sample locations.

WGAN under a fixed transport geometry. WGAN (Arjovsky et al., 2017) instead begins by fixing a ground cost $d_{\mathcal{X}}(x, y)$ between samples in the input space. This ground cost defines the Wasserstein-1 transport problem

$$W_1\left(p, q_\theta\right) = \inf_{\gamma \in \Pi(p, q_\theta)} \mathbb{E}_{(x,y)\sim\gamma}\left[d_{\mathcal{X}}(x, y)\right], \tag{44}$$

where $\Pi(p, q_\theta)$ denotes the set of couplings whose marginals are p and q_θ. The coupling γ specifies how probability mass is transported, while $d_{\mathcal{X}}$ specifies the cost of moving it.

Once $d_{\mathcal{X}}$ is fixed, the resulting optimal-transport problem can be expressed through the Kantorovich–Rubinstein dual:

$$W_1\left(p, q_\theta\right) = \sup_{\|f\|_{\mathrm{Lip}} \le 1} \left[\mathbb{E}_{x\sim p} f(x) - \mathbb{E}_{x\sim q_\theta} f(x)\right]. \tag{45}$$

The WGAN critic f is therefore a scalar dual potential for a transport problem whose sample representation and ground cost have already been specified. The critic optimization solves the dual problem but does not change how distances between samples are defined.

In summary, WGAN follows the structure

$$\underbrace{d_{\mathcal{X}}(x, y)}_{\text{fixed transport geometry}} \quad \Longrightarrow \quad \underbrace{\sup_{\|f\|_{\mathrm{Lip}} \le 1} \left[\mathbb{E}_p f - \mathbb{E}_{q_\theta} f\right]}_{\text{learned dual transport potential}}. \tag{46}$$

Fréchet distance as Gaussian transport. AdvFD begins from a different transport functional. For a representation $\psi_\omega : \mathcal{X} \to \mathbb{R}^d$, define the corresponding pushforward feature distributions as

$$P_\omega = (\psi_\omega)_{\#} p, \qquad Q_{\theta,\omega} = (\psi_\omega)_{\#} q_\theta. \tag{47}$$

Let

$$\mathcal{G}_2(\nu) = \mathcal{N}\left(\mu_\nu, \Sigma_\nu\right) \tag{48}$$

denote the Gaussian distribution with the same mean and covariance as ν. For a static representation, the Fréchet discrepancy is the squared Wasserstein-2 distance between the two Gaussian moment models:

$$\begin{aligned} D_{\mathrm{FD}}^{\psi_\omega}\left(p, q_\theta\right) &= W_2^2\left(\mathcal{G}_2(P_\omega), \mathcal{G}_2(Q_{\theta,\omega})\right) \\ &= \left\|\mu_p^\psi - \mu_q^\psi\right\|_2^2 \\ &\quad + \mathrm{Tr}\left[\Sigma_p^\psi + \Sigma_q^\psi - 2\left((\Sigma_p^\psi)^{1/2}\Sigma_q^\psi(\Sigma_p^\psi)^{1/2}\right)^{1/2}\right]. \end{aligned} \tag{49}$$

Equivalently,

$$D_{\mathrm{FD}}^{\psi_\omega}\left(p, q_\theta\right) = \inf_{\gamma \in \Pi(\mathcal{G}_2(P_\omega), \mathcal{G}_2(Q_{\theta,\omega}))} \mathbb{E}_{(z,z')\sim\gamma}\left[\|z - z'\|_2^2\right]. \tag{50}$$

Unlike WGAN, this Gaussian transport problem has a closed-form optimal value once ψ_ω is fixed. The mean term in Equation 49 measures the cost of translating the center of the generated feature distribution, while the covariance term measures the cost of aligning its scale and orientation with those of the real feature distribution.

Table 5: **Ablation on adversarial loss weight.** We vary the adversarial loss weight λ_{adv} under the same JiT-B post-training setting. The FD-loss baseline and all AdvFD variants are trained with the SIM FD objective. FD-r3 excludes the SIM representations used for training. Lower is better for all metrics.

Method	λ_{adv}	FID ↓	FD-r6 ↓	FD-r3 ↓
FD-loss baseline	–	1.00	5.53	8.45
AdvFD	0.01	0.83	4.51	7.34
AdvFD	0.05	0.80	4.68	6.28
AdvFD	0.10	**0.79**	**3.92**	**6.03**
AdvFD	0.20	0.87	5.46	6.74

AdvFD as learned transport geometry. AdvFD does not learn a dual potential for one fixed transport problem. Instead, it optimizes the representation that determines the feature distributions participating in the Gaussian transport problem. Let Ψ denote the set of representations reachable by the D-step from the initialization ψ_{ω_0} under the clipping constraint of 4. We have the optimization goal:

$$\sup_{\psi_\omega \in \Psi} D_{\mathrm{FD}}^{\psi_\omega}(p, q_\theta) = \sup_{\psi_\omega \in \Psi} W_2^2\big(\mathcal{G}_2(P_\omega),\, \mathcal{G}_2(Q_{\theta,\omega})\big). \tag{51}$$

Each representation induces a feature-space comparison cost

$$d_{\psi_\omega}(x, y) = \|\psi_\omega(x) - \psi_\omega(y)\|_2\,, \tag{52}$$

and changes the mean and covariance structures used in Equation 49. Maximizing over ψ_ω therefore searches for a feature geometry in which the remaining second-order mismatch between p and q_θ is most pronounced.

For each fixed ψ_ω, the Gaussian transport problem is solved in closed form. The remaining representation optimization generally has no analytic solution and is performed through the D-step. The generator then minimizes the resulting discrepancy through the G-step. Hence, AdvFD follows the structure

$$\underbrace{\bar{\psi}_{\omega_t} \longrightarrow \bar{\psi}_{\omega_{t+1}}}_{\text{adversely updated feature geometry}} \quad \text{applied to} \quad \underbrace{W_2^2\big(\mathcal{G}_2(P_\omega), \mathcal{G}_2(Q_{\theta,\omega})\big)}_{\text{fixed closed-form Gaussian transport functional}} \tag{53}$$

The contrast with WGAN can therefore be summarized as

$$\begin{aligned} &\text{WGAN:} && \text{fixed transport geometry} + \text{learned dual transport potential},\\ &\text{AdvFD:} && \text{learned feature geometry} + \text{fixed Gaussian transport functional}. \end{aligned} \tag{54}$$

WGAN learns the solution to an optimal-transport problem under a predefined sample-space geometry. AdvFD instead learns the geometry in which a predefined second-order transport objective is evaluated.

G More Experiment Results

Ablation on Adversarial Loss Weight. Table 5 studies the effect of the adaptive loss weight λ_{adv}. All nonzero weights improve over the FD-loss baseline, confirming that the adversarial representation provides complementary supervision. However, a small weight such as 0.01 underutilizes the exposed discrepancies and brings relatively limited gains, particularly on the FD-r3 metric. Increasing λ_{adv} to 0.05 or 0.10 yields substantially better results, with 0.10 achieving the best FD-r6 and FD-r3. A larger weight of 0.20 degrades all metrics, as the adaptive term begins to dominate the stable static FD objective and may introduce overly strong or conflicting gradients that disrupt the original post-training process. These results suggest that a moderate adaptive loss weight provides the best balance between preserving static distribution alignment and correcting its representation blind spots.

Wall-clock-aligned comparison. Table 6 shows that the gains of AdvFD cannot be attributed to additional computation or merely adding another representation branch. When the adversarial representation is frozen, it becomes an additional static target that cannot adjust to the evolving generated distribution, thereby overemphasizing a fixed feature geometry and substantially degrading all

Table 6: **Wall-clock-aligned comparison.** AdvFD (Frozen) retains the additional representation branch but freezes its parameters, isolating the effect of adversarial representation learning. FD-loss is additionally trained for N steps to match the total wall-clock cost of AdvFD.

Method	Adaptive Repr.	Train Steps	Wall-Clock	FID ↓	FD-r6 ↓	FD-r3 ↓
FD-loss	–	T	C	1.00	5.53	8.45
AdvFD (Frozen)	Frozen	T	C_{AdvFD}	1.62	11.02	8.71
FD-loss	–	$T+N$	C_{AdvFD}	1.71	7.70	8.63
AdvFD	Trainable	T	C_{AdvFD}	**0.79**	**3.92**	**6.03**

metrics. Likewise, extending FD-loss to match the wall-clock cost of AdvFD provides no improvement, indicating that longer optimization of the same static objectives is insufficient. In contrast, AdvFD achieves an FID of 0.79, FD-r6 of 3.92, and FD-r3 of 6.03 under the same wall-clock budget, confirming that the gains arise from adaptively updating the representation to expose residual real–generated discrepancies.

H Implementation and Evaluation Details

H.1 Implementation Details

Training Settings. We conduct all experiments on ImageNet-1K at a resolution of 256×256. We initialize the post-training stage from the released base checkpoints and keep the original pMF and JiT generator architectures. Unless otherwise stated, the main experiments use the FD-SIM objective together with the FD-Adv objective. The training hyperparameters are summarized in Table 7.

The pMF models use 2D rotary position embeddings, learned positional embeddings, and disable the auxiliary velocity head. JiT uses 2D rotary position embeddings, learned positional embeddings, and the legacy time convention. Gradient checkpointing is enabled for the large-scale experiments. The generator weights are tracked with an EDM-style model EMA, while the final released-checkpoint evaluation uses the online generator weights.

At each training iteration, class labels are sampled uniformly from the 1000 ImageNet classes and one-step images are generated with `num_sampling_steps`=1. The FD-SIM objective matches the feature distributions extracted by SigLIP, MAE, and InceptionV3. The corresponding feature target resolutions are 224, 224, and 299, respectively, and all three FD terms use unit weight. The representation models are frozen with respect to their parameters, but gradients are propagated through the generated images.

The FD reference statistics are precomputed from ImageNet images after center cropping and tensor conversion. During training, the FD feature statistics are initialized from 50,000 generated samples using a queue fill batch size of 256. The running FD statistics are then updated with an exponential moving average using $\beta = 0.999$. This EMA is independent of the EMA used for generator weights. The generated features are gathered across GPUs before computing their mean and covariance, and the covariance trace term is evaluated using the symmetric eigendecomposition implemented by `--fd_eigvalsh`. The normalized FD loss is defined as

$$\mathcal{L}_{\mathrm{FD}}^{\mathrm{norm}} = \frac{\mathcal{L}_{\mathrm{FD}}}{\mathrm{sg}(\mathcal{L}_{\mathrm{FD}}) + 0.01},$$

where $\mathrm{sg}(\cdot)$ denotes stop-gradient.

For FD-Adv training, a trainable copy of the selected representation backbone is updated jointly with the generator. The FD-Adv critic uses AdamW with a learning rate of 2×10^{-6} for pMF and 1×10^{-6} for JiT, as summarized in Table 7. In the SIM+Adv recipes, the critic is updated every two generator steps. The adversarial term starts at step 1,000 and is linearly warmed up for 4,000 steps. Real-feature gradients are detached, and the adversarial FD uses real-reference whitening with $\epsilon = 10^{-3}$. The FD-Adv feature statistics are maintained with a separate EMA using $\beta = 0.99$. The queue-size, EMA, and backbone ablations use the same batch-size and optimizer settings, while the shorter ablation runs use 50 training epochs.

Table 7: Training hyperparameters for the 256×256 experiments.

Hyperparameter	pMF	JiT
Generator		
Initialization	`pMF-B/L/H.pth`	`JiT-B/L/H.pth`
Global batch size	1024	1024
Per-GPU batch size	$1024/G$ (128 with 8 GPUs)	$1024/G$ (128 with 8 GPUs)
Training length	100×1250 steps	100×1250 steps
Generator optimizer	AdamW	AdamW
Generator learning rate	1×10^{-6}	1×10^{-5}
Adam betas	$(0.9, 0.95)$	$(0.9, 0.95)$
Weight decay	0	0
Learning-rate warm-up	5 epochs	5 epochs
Learning-rate schedule	cosine	cosine
Sampling steps	1	1
Model-weight EMA	EDM	EDM
Main FD Objective		
Main FD representation	SigLIP + MAE + Inception	SigLIP + MAE + Inception
FD queue size	50,000	50,000
FD queue fill batch size	256	256
FD-statistics EMA β	0.999	0.999
Covariance solver	`eigvalsh`	`eigvalsh`
Adversarial Representation		
FD-Adv backbone	pretrained Inception	pretrained Inception
FD-Adv learning rate	2×10^{-6}	1×10^{-6}
FD-Adv weight	0.05	0.10
FD-Adv steps per update	1	1
FD-Adv update frequency	every 2 generator steps	every 2 generator steps
FD-Adv start step	1,000	1,000
FD-Adv warm-up steps	4,000	4,000
FD-Adv gradient clipping	1.0	1.0
FD-Adv real-feature gradient	detached	detached
FD-Adv-statistics EMA β	0.99	0.99
FD-Adv whitening	enabled, $\epsilon = 10^{-3}$	enabled, $\epsilon = 10^{-3}$

H.2 Evaluation Details

We evaluate each checkpoint using 50,000 generated images. The released-checkpoint evaluation uses an evaluation batch size of 128 images per GPU and evaluates the online generator weights. The model-specific sampling configurations are:

- pMF-B: CFG $= 8.5$, interval $[0.1, 0.7]$, and noise scale 1.0;
- pMF-L: CFG $= 7.0$, interval $[0.2, 0.7]$, and noise scale 1.0;
- pMF-H: CFG $= 7.0$, interval $[0.2, 0.6]$, and noise scale 2.0;
- JiT-B: CFG $= 3.0$ and interval $[0.1, 1.0]$;
- JiT-L: CFG $= 2.4$ and interval $[0.1, 1.0]$;
- JiT-H: CFG $= 2.2$ and interval $[0.1, 1.0]$.

The default evaluation suite contains InceptionV3, ConvNeXt, DINOv2-L, MAE-L, SigLIP-SO400M, and CLIP-L. Generated images are converted to the $[0, 1]$ range before feature extraction and resized to the encoder-specific target resolution. For each representation space, we compute the raw Fréchet distance between the generated-image statistics and the corresponding ImageNet reference statistics.

We report the normalized representation Fréchet distance

$$\text{FD-r}_m = \frac{\text{FD}_m}{\text{valFD}_m},$$

where the validation-set normalizers are 1.68 for Inception, 56.87 for ConvNeXt, 14.19 for DINOv2, 0.04 for MAE, 0.60 for SigLIP, and 5.60 for CLIP. We report three aggregate metrics. First, FDr-6 is the arithmetic mean across all six representation spaces:

$$\text{FD-r6} = \frac{1}{6}\left(\text{FDr}_{\text{Inception}} + \text{FDr}_{\text{ConvNeXt}} + \text{FDr}_{\text{DINOv2}} + \text{FDr}_{\text{MAE}} + \text{FDr}_{\text{SigLIP}} + \text{FDr}_{\text{CLIP}}\right).$$

Second, FD-r3 is computed over the DINOv2, CLIP, and ConvNeXt representation spaces:

$$\text{FD-r3} = \frac{1}{3}\left(\text{FDr}_{\text{DINOv2}} + \text{FDr}_{\text{CLIP}} + \text{FDr}_{\text{ConvNeXt}}\right).$$

Thus, FDr-3 excludes the Inception, SigLIP, and MAE representation spaces.

I LIMITATIONS

Our current evaluation focuses on class-conditional ImageNet generation at 256×256 resolution. Although AdvFD consistently improves multiple generator backbones and model scales, its effectiveness on higher-resolution, text-conditioned, and video generation remains to be validated. In addition, AdvFD introduces extra training-time computation for updating the adversarial representation and computing the whitening transformation, while leaving the inference architecture and 1-NFE sampling cost unchanged.