ROAD: MODERN ADVERSARIAL TRAINING FOR SIMPLE ONE-STEP DISTILLATION AT SCALE

Anonymous authors
Paper under double-blind review

ABSTRACT

One-step distillation turns pretrained diffusion models into fast generators, but strong results often require substantial training machinery and compute. Pure adversarial distillation offers a simpler route, yet has struggled at modern text-to-image scale. To address this limitation, we propose ROAD (Representation-grounded One-step Adversarial Distillation), a one-stage framework that distills text-to-image models with one adversarial objective and a lightweight representation-space critic. First, we replace the classical convolutional head with a Lipschitz-controlled Transformer head with token-level text conditioning, which matches the semantic tokens of modern visual encoders. Second, we observed a fixed random feature mixing leads to long-horizon degradation. We attribute it to generator–critic co-adaptation around one realization-specific view, and periodically re-draw the mixing to keep the randomization effective. Third, for our dense and vector-valued logits, we introduce Jacobian R1 which penalizes the full input–logit Jacobian, overcoming pooled R1's limitation of neglecting most of logit sensitivities. A single adversarial objective then suffices: in one step, ROAD reaches 11.03 FID on SDXL and 13.73 on SD3.5-M on COCO-10K, surpassing the multi-step teachers at a fraction of distribution matching's training compute.

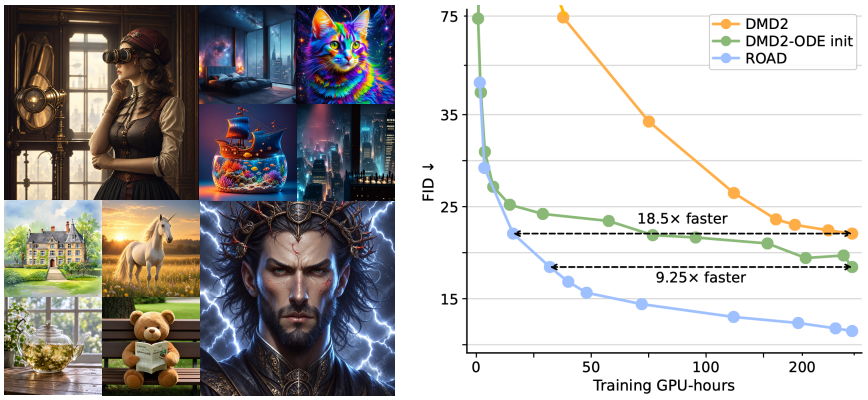


Figure 1: **A single adversarial loss suffices for one-step distillation.** Left: samples from ROAD. Right: one-step SDXL FID vs. training GPU-hours (our runs). DMD2 starts from the base model; DMD2-ODE init (released recipe; Yin et al., 2024a) starts with a regression stage on teacher ODE pairs. ROAD needs no such stage, beats both baselines' best FID early, and keeps improving.

1 INTRODUCTION

Diffusion and flow models have achieved remarkable quality in text-to-image generation, but their iterative sampling remains expensive at inference time. Step distillation therefore aims to compress a many-step generator into one or a few evaluations. Paradoxically, however, the strongest one-step methods often simplify inference by making training substantially more complicated: distribution matching introduces an online score network at base-model scale (Luo et al., 2023; Yin et al., 2024b; Nguyen & Tran, 2024; Zhou et al., 2024), while competitive systems commonly rely on additional

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.

objectives, auxiliary models, or multiple stages (Tab. 2, Sec. D). Thus, *one-step inference has become simple, but one-step training has not*.

Adversarial distillation offers a strikingly simpler alternative. Rather than matching scores or trajectories, it can train a one-step generator directly against real data with a single adversarial objective, without an online score network or repeated teacher queries. Yet this simplicity has not carried over to modern text-to-image models. Pure adversarial training at this scale has struggled with stability and prompt alignment (Sauer et al., 2024b; Yin et al., 2024a). As a result, adversarial losses are typically used only to assist a heavier distillation objective (Sauer et al., 2024b; Zhou et al., 2025c). When adversarial training becomes the main objective, existing methods often recover performance by introducing other sources of complexity, such as teacher-generated supervision, previously distilled generators, or large trainable critics (Sauer et al., 2024a; Chen et al., 2025; Lin et al., 2025). This leaves open a more fundamental question: *is pure adversarial distillation itself unable to scale, or is the critic used to train it the actual bottleneck?*

We investigate the latter possibility. The critic used in adversarial distillation is largely inherited from projected GANs (Sauer et al., 2021; 2023), while the representation space it operates on has changed substantially. Modern visual encoders such as DINOv3 and Perception Encoder already map images into contextual, semantically organized token representations (Siméoni et al., 2026; Bolya et al., 2025). Yet these representations are still read through lightweight convolutional heads; the random feature transform is sampled once and held fixed throughout training; and gradient regularization acts only on a scalar obtained by pooling the dense critic outputs. We find that these seemingly innocuous design choices become increasingly restrictive at modern model scale. They limit how the critic interprets semantic representations, allow its judgments to become increasingly tied to a single feature view during prolonged optimization, and leave individual logit sensitivities insufficiently controlled. This diagnosis leads us to revisit pure adversarial distillation with a modern critic tailored to today's representations and optimization dynamics, making its simple training recipe effective at text-to-image scale. We instantiate this idea in ROAD, **R**epresentation-grounded **O**ne-step **A**dversarial **D**istillation.

Reading modern representations with modern ViT heads. The inherited convolutional readout builds a visual pyramid over fixed neighborhoods, whereas modern token representations call for a head that selects and relates semantic evidence by content. We therefore use ViT heads made viable for adversarial training through Lipschitz controls, while cross-attention to individual prompt tokens improves text alignment. The resulting critic clearly outperforms similar-sized convolutional heads in fidelity and alignment (Sec. 3.1).

Keeping the random mixing random. Projected critics read the frozen features through a random mixing, drawn once and held fixed so that the critic does not rely on a few dominant feature directions (Sauer et al., 2021). This protection, however, does not last over long horizons: training under a fixed draw converges quickly, then oscillates and slowly deteriorates. We attribute this to generator–critic co-adaptation around one persistent view of the representation: directions measured weakly by that view can remain poorly corrected. ROAD periodically redraws the mixing while carrying forward the generator, heads, and optimizer states. This changes the feature view without restarting learning, limits prolonged co-adaptation, and sustains improvement where fixed-mixing runs deteriorate (Sec. 3.2).

Adapting R1 to dense, vector-valued critics. The classical R1 penalty regularizes a scalar critic through its input gradient on real images and is a staple of stable GAN training (Mescheder et al., 2018; Karras et al., 2020b; Kang et al., 2024). ROAD's critic, however, is dense and vector-valued, and the adversarial loss acts on every logit, whereas the customary extension pools the logits into a scalar before applying R1 (Zheng & Yang, 2025; Suvorov et al., 2022; Kang et al., 2023). Pooling keeps only one direction of the critic's output space, so the input–logit Jacobian's sensitivities in every other direction, however large, never enter the penalty. The proposed Jacobian R1 (JR1) instead penalizes every logit's input gradient, i.e., the squared Frobenius norm of the input–logit Jacobian. A single unbiased Hutchinson probe makes it practical with one vector–Jacobian product (Sec. 3.3).

ROAD proves to be a simple, low-cost, and strong one-step distillation method: with one loss and one stage, it reaches 11.03 FID on SDXL and 13.73 on SD3.5-M on COCO-10K. On SDXL, it reaches the best FID of ODE-initialized DMD2 using 1/9.25 of the training GPU-hours (Fig. 1).

2 PRELIMINARIES

Pure adversarial one-step distillation. Given a pretrained text-to-image diffusion model, we initialize a one-step student G_θ from its architecture and weights. The student maps Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ and a prompt $c \sim p_c$ directly to an image at a fixed conditioning timestep, without querying the teacher thereafter. Following Projected GAN, the critic judges images through a frozen pretrained encoder F, whose features are tapped at $L = 4$ depths, $f_\ell(x)$ for $\ell = 1, \dots, L$. The conditional head at level ℓ emits dense N_ℓ per-position logits. Concatenating the logits across all levels gives a vector $D_\phi(x, c) = [D_1(x, c), \dots, D_n(x, c)]^\top \in \mathbb{R}^n$, where $n = \sum_\ell N_\ell$. For real image–prompt pairs $(x, c) \sim p_r$, the critic and generator minimize, respectively,

$$\begin{aligned} \mathcal{L}_D &= \mathbb{E}_{(x,c)\sim p_r}\Big[\sum\nolimits_i \omega_i\, \zeta\big(-D_i(x, c)\big)\Big] + \mathbb{E}_{\epsilon,c}\Big[\sum\nolimits_i \omega_i\, \zeta\big(D_i(G_\theta(\epsilon, c), c)\big)\Big], \\ \mathcal{L}_G &= \mathbb{E}_{\epsilon,c}\Big[\sum\nolimits_i \omega_i\, \zeta\big(-D_i(G_\theta(\epsilon, c), c)\big)\Big], \end{aligned} \tag{1}$$

where $\omega_i = 1/n$ and ζ is the softplus function. These are the logistic critic and non-saturating generator losses (Goodfellow et al., 2014), with the nonlinear loss applied to each logit before uniform averaging. We alternate minimizing $\mathcal{L}_G$ over θ and $\mathcal{L}_D + R_D$ over ϕ, where R_D is the critic gradient penalty defined below and is zero in unregularized configurations.

Random mixing in projected critics. Projected GAN feeds the frozen features through *non-learned random affine maps* (i.e., CCM and CSM) before the trainable multi-scale heads. CCM reorganizes channels within each feature level, and CSM further combines information from different levels along a top-down path. We denote the collection of mixing parameters by M, the critic's measurement basis (subscripts index feature levels; parenthesized superscripts index draws). Schematically,

$$\tilde{f}_\ell = M_\ell^{\mathrm{cc}} f_\ell, \qquad \hat{f}_\ell = M_\ell^{\mathrm{cs}}\big(\hat{f}_{\ell+1} + \tilde{f}_\ell\big), \qquad \hat{f}_L = M_L^{\mathrm{cs}} \tilde{f}_L, \tag{2}$$

where the top-down recursion applies for $\ell = L - 1, \cdots, 1$. In the classical design, these maps are sampled at initialization and then held fixed. By expressing pretrained features in mixed coordinates, the heads base their decisions on combinations of cues across channels and depths. This prevents the discriminator from relying solely on a few dominant feature directions and encourages broader feature use (Sauer et al., 2021).

Pooled R1 regularization. For the vector-valued critic of Eq. (1), the pooled R1 first averages the n logits and applies the classical R1 penalty to the resulting scalar:

$$R_1^{\mathrm{pool}} = \frac{\gamma}{2}\, \mathbb{E}_{(x,c)\sim p_r} \big\| \nabla_x \big(\tfrac{1}{n} \sum\nolimits_i D_i(x, c) \big) \big\|_2^2. \tag{3}$$

This pooled form controls the input sensitivity of the averaged logit. The common practice is to apply this regularization lazily to reduce cost (Karras et al., 2020b; Kang et al., 2023).

3 SCALING PURE ADVERSARIAL DISTILLATION WITH A REPRESENTATION-GROUNDED CRITIC

ROAD trains the one-step student with the dense adversarial losses and the JR1 penalty against a projected critic assembled from three parts: the frozen encoder, the random mixings, which ROAD periodically re-draws, and trainable Transformer heads that attend to the prompt's text tokens and emit dense per-position logits (Fig. 2). The teacher appears only as the student's initialization.

3.1 A TRANSFORMER HEAD FOR SEMANTIC TOKEN REPRESENTATIONS

A natural starting point is to retain the convolutional discriminator design of Projected GAN (Sauer et al., 2021). In our one-step distillation setting, this convolutional (Conv) readout remains trainable but underperforms in both image quality and prompt adherence, and simply increasing its capacity (Deep Conv) further degrades performance, as shown in rows 1 and 2 of Tab. 1, respectively.

We attribute this to a change in what the trainable head is for. A modern ViT backbone such as DINOv3 already maps an image into a contextual, semantic representation space, and every tap returns the same number of tokens. The head need not rebuild the visual pyramid that projected

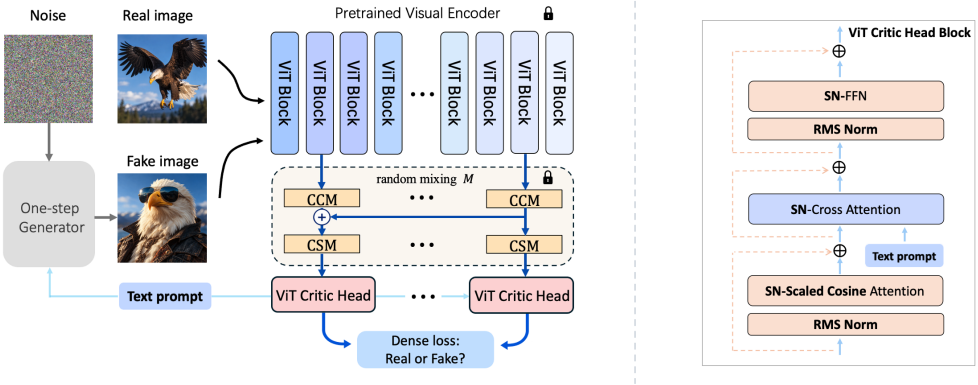


Figure 2: **Overall architecture and critic head's ViT block of ROAD.** Left: generated and real images are encoded into multi-level semantic features, transformed by periodically re-drawn random mixings M, and scored by Transformer critic heads. Right: a Lipschitz-controlled ViT block uses spectral-normalized scaled-cosine self-attention, token-level cross-attention, and an FFN, together with RMSNorm and learnable residual scales.

	Critic head	Param.	FID ↓	CLIP ↑
1	Conv	24.8M	19.02	0.317
2	Deep Conv	54.8M	21.52	0.311
3	Naive Transformer	54.4M	Collapse	–
4	ViTGAN	54.3M	15.40	0.319
5	ViT (ours)	54.3M	14.54	0.331

	Text conditioning	CLIP ↑
6	Unconditional	0.320
7	Miyato-style pooled-text projection	0.322
8	Token-level cross-attention	0.331

Table 1: **Critic head design and text conditioning.** Fixed mixing and pooled R1 are used with 2B generator (SDXL). **Top:** Conv follows the convolutional head of Projected GAN. Raising its capacity does not improve performance, whereas our ViT head substantially improves both fidelity and alignment; the naive Transformer head collapses. **Bottom:** Pooled-text projection provides only a minor gain over the unconditional critic, whereas token-level cross-attention yields a stronger improvement in text–image alignment.

GAN's convolutional heads construct; it has only to decide which tokens matter and how they relate. Convolution aggregates over predetermined neighborhoods with fixed, translation-shared weights, an inductive bias suited to building a hierarchy from raw pixels rather than to reading backbone features: two tokens are combined because they are adjacent, not because their content is related. Self-attention decides instead from the tokens themselves which of them are combined and how strongly, which makes a Transformer head the more promising choice.

Lipschitz control, however, is necessary to make this replacement viable in the adversarial game (Zhou et al., 2019; Miyato et al., 2018). It limits how strongly the critic can amplify changes in its input and thereby constrains abrupt variation in the signal returned to the generator. A naive Transformer head collapses in our setting (Tab. 1, row 3). This failure is consistent with a known difficulty: standard dot-product softmax attention is not globally Lipschitz in its inputs, even when the learned linear maps have bounded spectral norm (Kim et al., 2021).

A Lipschitz-controlled ViT head is therefore introduced, which constrains the gain of every trainable component through three complementary mechanisms. First, *spectral normalization* (Miyato et al., 2018) is applied to every learned linear map in attention, the FFN, and the logit readout, so that its spectral norm is no more than 1. Second, *cosine attention with a clamped temperature* normalizes every query and key before their comparison under a learnable per-head temperature clamped at $s_{\max}$ (Liu et al., 2022). This removes their raw activation magnitudes from the attention logits and bounds the remaining scale explicitly. Third, *each residual branch uses a learnable per-channel scale* initialized to $1/B_\ell$ for a level-ℓ head of depth B_ℓ, preventing the cumulative residual gain from growing excessively with the head depth. On any convex region where queries and keys keep a norm floor before ℓ_2-normalization, these controls make the head's map from mixed tokens to logits Lipschitz with the explicit constant Λ_ℓ of Eq. (7), proved as Lemma A.1 in Sec. A.

With the controls in place, the ViT head trains successfully and substantially improves the quality of generated images (Tab. 1, row 5). ViTGAN (Lee et al., 2022) provides an alternative ViT discriminator design, but remains inferior to our head under the same protocol (Tab. 1, row 4). The

weak prompt adherence long associated with adversarial distillation, however, persists (Sauer et al., 2024b; Yin et al., 2024a). Comparing text-conditioning schemes, we attribute this shortfall to the pooled prompt vector: a single pooled embedding cannot express fine-grained linguistic structure.

Token-level text conditioning via cross-attention. The Transformer head admits token-level cross-attention naturally, and cross-attention provides a direct mechanism for image tokens at different spatial positions to interact differently with individual text tokens. We insert cosine cross-attention into the first block of every head, with subsequent blocks refining the conditioned representation through self-attention and FFN layer. Each level keeps its own text key/value projections, allowing different semantic depths to read the same prompt differently. Under the unified protocol, Miyato-style pooled-text projection provides only a minor improvement over the unconditional critic, whereas token-level cross-attention yields a much stronger gain in CLIP score (Tab. 1, rows 6-8).

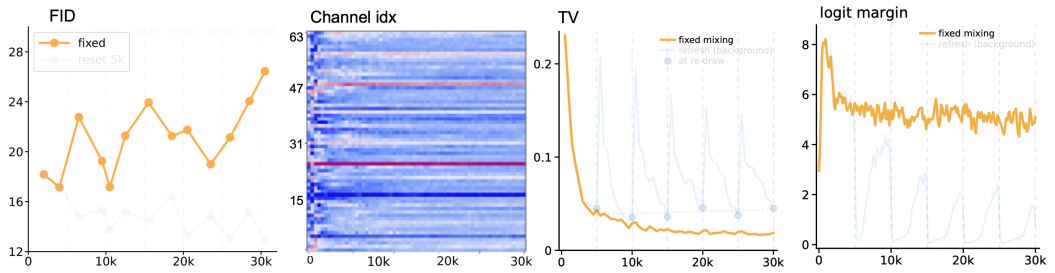
Multi-branch random mixing. Projected GAN's random mixing is tailored to a convolutional pyramid, where CCM is a 1×1 convolution within a level and CSM adds 3×3 convolutions between levels of differing resolution. A ViT backbone has no such pyramid, since all taps share one uniform patch-token grid, so both maps reduce to a plain 1×1 convolution that mixes channels alone. We therefore replace each single-kernel mixing by an average of three size-preserving random convolutions over the patch tokens, with kernels 1×1, 3×3, and 5×5; the summary token is transformed separately by a random linear map. The wider branches let a token's mixed value draw on its spatial neighbors, so evidence is spread across positions as well as channels and depths. Sec. E.4 shows the advantage of this design. These frozen multi-branch maps constitute the mixing M that Sec. 3.2 re-draws, and Sec. B.1 gives its matrix form and analyzes its anisotropy.

3.2 Periodic Mixing Refresh Prevents Critic Co-adaptation

With the Transformer head of Sec. 3.1, training converges quickly to good performance, then begins to oscillate and slowly deteriorate (Fig. 3a). The symptom resembles discriminator overfitting, yet standard augmentation remedies such as DiffAugment (Zhao et al., 2020) and ADA (Karras et al., 2020a) do not materially alleviate this late-stage degradation under fixed mixing (Sec. E.4). We therefore examine other possible causes and finally focus on the random mixing, held fixed throughout training in the classical design. We argue that its anisotropy persistently biases long-horizon optimization toward the same feature directions. The generator and the critic then form a *realization-specific co-adaptation* around this single view, which eventually erodes generation quality.

A persistent critic state. We diagnose the critic with two measurements, Pre-projection Feature Utilization Coverage (PFUC) and the logit margin (Sec. B.3). *PFUC* is the first-order sensitivity of the critic's mean logit to each backbone channel before mixing, expressed in backbone coordinates. It settles after an early phase of training and barely moves thereafter (Fig. 3b): the TV between consecutive profiles decays to a floor (Fig. 3c). The *logit margin* is the batch-mean real logit minus the batch-mean generated logit, measuring how confidently the critic separates the two populations. After the first few thousand steps, the margin stays large and nearly stationary (Fig. 3d), even though FID has already begun its usual oscillation and deterioration. Because the generator is still changing, a critic decision that has stopped moving is not by itself a healthy sign. Together, the panels show a persistent critic state, i.e., hardened per-channel sensitivity profile and stationary logit margin.

Coupling to the fixed mixing. Write the critic as $D_\phi\big(M^{(0)}f(x)\big)$, where $M^{(0)}$ is the training draw. Across training, D_ϕ supplies the generator with an optimization signal shaped by this one realization. Gradually $M^{(0)}$ is absorbed into the critic's parameters and becomes the fixed geometry within which the adversarial game continues to be optimized. Finally, the way the signal uses the features itself hardens, producing the phenomena of Fig. 3b–d. *We test this account with two post-hoc experiments: fresh-draw transfer and a linear probe*. First, we hold one checkpoint fixed and replace only $M^{(0)}$ by independently drawn mixings. Across six checkpoints from 5k to 30k steps, the logit margin is about 5.0 under $M^{(0)}$ but about 0 under a fresh draw, indicating the critic's real–generated separation does not transfer to a fresh measurement basis. Second, at each level, we fit a separate linear classifier to the frozen mixed features to distinguish real from generated samples. From 2.5k steps on, the two populations are harder to separate under $M^{(0)}$ than under a fresh draw, particularly at the shallow levels. This suggests that what still distinguishes generated from real images is least visible under $M^{(0)}$, the view through which the generator was trained. The two diagnostics together show that the critic's separation is specific to the pairing $D_\phi \circ M^{(0)}$.



(a) FID vs. training steps (b) PFUC vs. training steps (c) PFUC change vs. training steps (d) logit margin vs. training steps

Figure 3: **Diagnosing co-adaptation under fixed random mixing.** (a) FID oscillates and eventually deteriorates. (b) PFUC settles into a persistent per-channel pattern. (c) The change between consecutive PFUC profiles, measured by total variation (TV), decays toward a floor. (d) The logit margin stays high and nearly stationary even as FID degrades. The twin run refreshed every 5k steps is faded for reference. Sec. B.3 describes the measurements in detail.

Anisotropy-based co-adaptation ends in degradation. Stack the frozen features as f and write the affine mixing as $\hat{f} = Mf + b$ (Sec. B.1). As a random M is anisotropic (Sec. B.1), the gain $\lambda_M(\eta) = \|M\eta\|$ that it applies to a unit feature direction η varies widely from one direction to another. This gain decides both how much the critic's logits change when the features move along a direction and how much of the generator's correction reaches it. Specifically, the directional derivatives of the critic and the generator along η are bounded as $|\langle \eta, \nabla_f D_i \rangle| \le \lambda_M(\eta)\Lambda_\ell$ and $|\langle \eta, \nabla_f \mathcal{L}_G \rangle| \le \lambda_M(\eta)\|\nabla_{\hat{f}} \mathcal{L}_G\|$, where Λ_ℓ is the Lipschitz bound of Lemma A.1 for the level-ℓ head that produces D_i. So a small-gain direction moves the logits little and is slowly corrected in the generator, however the critic behaves; we call it *weakly measured*. The minimal measured-gap surrogate in Prop. B.1(i) shows the feature-space residual between generated and real data decays along a right-singular direction at a rate equal to the square of that direction's gain. In light of the proposition, high-gain directions are corrected early, whereas weakly measured directions persist, concentrating the surviving residual in them. We thus hypothesize that an unchanging gain landscape sustains realization-specific co-adaptation.

Periodic mixing refresh suppresses co-adaptation-induced degradation. Given a refresh interval τ, at the start of the k-th period we re-draw the mixing $M^{(k)}$ from its initialization law. The generator, the critic heads, and the optimizer states remain unchanged, and no learnable parameter is reset or added. Each re-draw replaces the draw-specific gain landscape, so a residual direction that is weakly measured under $M^{(k)}$ may receive a higher gain under $M^{(k+1)}$. Under the right-isotropy assumption, independent re-draws prevent any fixed direction from retaining a privileged slow correction rate (Prop. B.1(ii)). After a refresh, the critic heads adapted to the preceding mixing are therefore temporarily mismatched with the new measurement geometry. Accordingly, PFUC reorganizes at each re-draw (Fig. 4b,c), while the logit margin resets toward zero and then re-forms as the heads adapt (Fig. 4d). By shortening how long the generator–critic pair can co-adapt to one fixed view, periodic refresh interrupts realization-specific co-adaptation and can re-expose weak residual directions to adversarial correction. Consistent with this mechanism, the refreshed run re-enters an improving trajectory after re-draws and maintains a more stable and lower FID over long-horizon training (Fig. 4a).

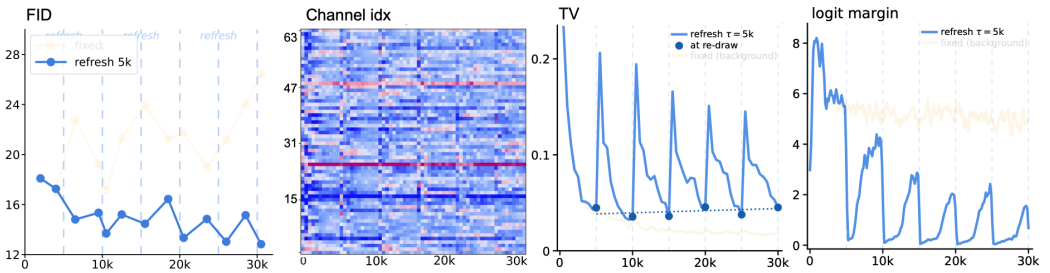
3.3 Jacobian R1 Regularization for Dense Critics

The objective in Eq. (1) defines per-logit nonlinear loss across positions and levels. So its regularizer should control the sensitivities of these individual decisions, not only their aggregate. We abbreviate the vector-valued logit $D_\phi(x, c)$ as $D(x)$ as both ϕ and c are fixed for gradient regularization. Let $J(x) = \partial D(x)/\partial x \in \mathbb{R}^{n \times d_x}$ where d_x is input dimension, with the i-th row being $g_i^\top = (\nabla_x D_i)^\top$.

Pooled R1 misses logit sensitivities. To facilitate analysis, we write the pooled R1 as

$$R_1^{\text{pool}} = \frac{\gamma}{2}\,\mathbb{E}_{x\sim p_r}\frac{1}{n^2}\|J(x)^\top \mathbf{1}\|_2^2 = \frac{\gamma}{2}\,\mathbb{E}_{x\sim p_r}\Big\|\frac{1}{n}\sum_{i=1}^n g_i(x)\Big\|_2^2, \tag{4}$$

which penalizes the squared norm of the averaged gradient. The per-image energy $q_{\text{pool}}(x) = \frac{1}{n^2}\|J(x)^\top \mathbf{1}\|_2^2 = \frac{1}{n^2}\text{tr}(J^\top \mathbf{1}\mathbf{1}^\top J)$ shows that pooled R1 only regularizes the direction spanned by



(a) FID vs. training steps (b) PFUC vs. training steps (c) PFUC change vs. training steps (d) logit margin vs. training steps

Figure 4: **Periodic mixing refresh breaks the persistent critic state.** The refreshed run (blue, every 5k steps) is foregrounded, with fixed mixing faded for reference. (a) FID improves after re-draws and avoids the late deterioration of fixed mixing. (b) PFUC reorganizes under each new view. (c) The PFUC change spikes after each re-draw and then decays; filled dots mark the value at each re-draw. (d) The logit margin resets toward zero at each re-draw and re-forms as the heads adapt.

the all-ones vector in the n-dimensional output space, while leaving the remaining $n-1$ directions unpenalized. Particularly, for $J^\top \mathbf{1} = 0$ the averaged logit is insensitive to every input perturbation to first order, while the individual logits need not be. In this case, pooled R1 does not impose any regularization at all, even though the dense loss of Eq. (1) is driven by those individual logits.

JR1 exploits full logit sensitivities. Motivated by input–output Jacobian sensitivity (Novak et al., 2018) and logit-level gradient penalties (Varga et al., 2018), we penalize the full input–logit Jacobian:

$$R_1^{\mathrm{J}} = \tfrac{\gamma_{\mathrm{J}}}{2}\,\mathbb{E}_{x\sim p_r}\tfrac{1}{n}\|J(x)\|_F^2 = \tfrac{\gamma_{\mathrm{J}}}{2}\,\mathbb{E}_{x\sim p_r}\tfrac{1}{n}\textstyle\sum_{i=1}^{n}\|g_i(x)\|_2^2. \tag{5}$$

So JR1 penalizes the mean of the individual squared norms rather than the squared norm of the mean. In light of the per-image energy $q_{\mathrm{Jac}}(x) = \frac{1}{n}\|J(x)\|_F^2$, all n directions of output space are regularized, including the $n-1$ that pooled R1 neglects. Particularly, when $J^\top \mathbf{1} = 0$, pooled R1 imposes no regularization while JR1 retains every individual sensitivity. *Prop. C.1 provides detailed theoretical comparison between pooled R1 and JR1.*

Efficient estimation with Hutchinson probes. Exact JR1 requires backpropagating each of the n logits separately, i.e., one vector–Jacobian product (VJP), which is computationally expensive. We instead use the classical Hutchinson trace estimator (Hutchinson, 1989), following its use for Jacobian regularization (Varga et al., 2018; Hoffman et al., 2019). Let $v \in \{-1,+1\}^n$ have independent, equiprobable signs, so that $\mathbb{E}[vv^\top] = I_n$. As $\mathbb{E}_v\|J^\top v\|_2^2 = \mathrm{tr}(J^\top \mathbb{E}[vv^\top]J) = \|J\|_F^2$, we have

$$R_1^{\mathrm{J}} = \frac{\gamma_{\mathrm{J}}}{2n}\,\mathbb{E}_{x\sim p_r}\,\mathbb{E}_v\left\|J(x)^\top v\right\|_2^2. \tag{6}$$

In practice we approximate $\mathbb{E}_v$ by a Monte-Carlo average over P independent probes, and use $P{=}1$ per real image, drawn afresh on the lazy schedule of JR1 critic updates. *Prop. C.2 shows that the estimate is unbiased and provides its variance.*

3.4 The complete recipe.

So far, the three components accomplish modernization of projected GAN, addressing three degrees of freedom of a representation-space critic, what judges (Sec. 3.1), through what measurement (Sec. 3.2), and how sharply (Sec. 3.3). Together they make the minimal objective of Eq. (1) sufficient at modern text-to-image scale.

4 Experiments

Setup. We train one-step generators for SDXL (Podell et al., 2024), a diffusion model with a UNet architecture, and SD3.5-M, a flow-matching model with an MMDiT architecture. Across the paper, DINOv3 ViT-B is chosen as the backbone for feature extraction unless otherwise specified. We use a dataset of 500K images from LAION-Aesthetic 5.5+ (Schuhmann et al., 2022) built by DMD2. Following (Yin et al., 2024a; Sauer et al., 2024b; Zhou et al., 2025b), we evaluate on a random set of 10K COCO-2014 validation prompts. For both models we report FID, CLIP

Method	Training		FID ↓	CLIP ↑	HPSv2 ↑	Pick ↑	Aes ↑
	Tech.	Stage					
SDXL base (cfg 6)	–	–	19.29	0.331	28.33	22.11	6.19
Hyper-SDXL	Traj.+Reward+ ScoreD+GAN	6	33.66	0.328	**29.97**	**22.71**	6.59
DMD2	ScoreD+GAN	2	19.01	0.336	28.36	22.09	6.09
DMDX	Adv. ScoreD	2	–	**0.352**	27.70	22.27	–
SiDa	ScoreD+GAN	1	15.23	0.326	27.80	21.72	6.16
SDXL-Turbo	GAN+ScoreD	1	24.56	0.337	28.53	22.40	6.08
SDXL-Lightning	Traj.+GAN	5	23.93	0.316	28.13	21.92	6.19
NitroSD-Realism	ScoreD+GAN	3	20.80	0.335	28.84	22.37	6.31
ROAD (ours)	**GAN**	**1**	**11.03**	0.338	28.58	22.13	**6.67**

Table 2: **Evaluation of one-step SDXL.** All results are reproduced under our protocol with official checkpoints, except DMDX, whose values are from its paper. Unlike prior methods that require multiple objectives and multi-stage training (Sec. D), ROAD uses a single stage with a pure GAN loss.

Method	Tech.	NFE	FID ↓	CLIP ↑	HPSv2 ↑	IR ↑	Pick ↑	Aes ↑	GE ↑
SD3.5-M (cfg 4.5)	–	40	18.01	0.341	29.19	1.00	22.38	6.12	0.69
SD3.5-M-Turbo	–	4	23.22	0.330	26.80	0.47	21.59	5.86	0.55
SenseFlow	ScoreD+GAN+Traj.	4	28.07	0.328	**29.54**	0.93	22.59	6.28	0.62
SwD	ScoreD+GAN+MMD	4	26.56	0.338	29.48	**1.15**	**22.91**	**6.39**	**0.71**
SiD-DiT	ScoreD	4	22.02	0.338	29.36	1.01	22.59	6.24	0.64
ROAD (ours)	**GAN**	1	**13.73**	**0.342**	28.85	0.94	22.00	6.15	0.66

Table 3: **Evaluation of few-step SD3.5-M.** All results are reproduced under our protocol with official 4-step checkpoints; there are no 1-step methods yet, to the best of our knowledge.

score, and aesthetic score. We additionally report human-preference metrics: PickScore (Kirstain et al., 2023) and HPSv2 (Wu et al., 2023) for SDXL, as in (Lu et al., 2025; Ren et al., 2024), and PickScore, HPSv2, ImageReward (Xu et al., 2023), and GenEval (Ghosh et al., 2023) for SD3.5-M, following (Starodubcev et al., 2026; Chen et al., 2025). Complete descriptions of the dataset and evaluation metrics, training hyperparameters, and additional experiments are provided in Secs. E.2, E.3, and E.4, respectively.

4.1 Main Results

Does a single GAN loss reach state-of-the-art one-step quality? **On SDXL**, we compare with strong baselines in Tab. 2. Typically, most of these methods consist of multiple training losses and require trajectory (traj.) distillation, score distillation (ScoreD), GAN, and reward learning, along with multiple stages such as ODE regression training followed by ScoreD+GAN. Compared to them, ROAD is much simpler, with a pure GAN loss and a single training stage. Nevertheless, it achieves 11.03 FID, ahead of all the baselines, and is also competitive on CLIP score, aesthetic score, and preference scores, suggesting that ROAD does not trade preference or alignment for fidelity. The comparison **on SD3.5-M**, a flow-based MMDiT generator, is given in Tab. 3. To the best of our knowledge, no one-step generators exist for it yet, so we compare with few-step models. Again, one-step ROAD achieves strong results on FID, much better than the competing methods as well as the teacher. Its CLIP score is among the best, indicating promising text-image alignment, while its preference scores are slightly inferior to those of the compared methods. We note that all these SOTA methods use 4 steps against our single step and are therefore 4 times slower than ours, while most adopt three training losses against our single GAN loss.

4.2 Ablation Study

Across this section, the one-step SDXL generator is used for ablation, in which the refresh interval τ is 5k and the regularization is JR1 with $\gamma_J = 7.5 \times 10^{-3}$ unless otherwise specified.

Transformer head. Tab. 1 (Sec. 3.1) isolates the critic head design and the text conditioning under fixed mixing and pooled R1. Increasing the capacity of the convolutional head does not improve performance. Our Lipschitz-controlled ViT head instead yields a substantial gain in both fidelity and alignment, and it also outperforms the ViTGAN alternative. For conditioning, token-level cross-attention provides a clearly stronger alignment gain than Miyato-style pooled-text projection.

Component decomposition. Tab. 4(a) presents a breakdown of ROAD's three components. Lipschitz control first makes the ViT head trainable, indicating its importance for stabilizing GAN. Periodic refresh then improves FID from 14.54 to 12.08 and CLIP from 0.331 to 0.335, demonstrating its benefit of refresh mixing over no refresh (i.e. fixed mixing). Finally, three components together reduces FID to 11.03 while improving CLIP to 0.338. Further, trajectory-level comparison (Fig. 6) shows JR1's benefit persists across training rather than appearing only at the selected checkpoint.

Table 4: **Analysis of key components of ROAD.**

(a) Component decomposition

Critic	Lip. Ctrl.	Refresh	Regularizer	FID ↓	CLIP ↑
ViT	✗	✗	Pooled R1	collap	–
ViT	✓	✗	Pooled R1	14.54	0.331
ViT	✓	5k	Pooled R1	12.08	0.335
ViT	✓	✗	JR1	13.62	0.332
ViT	✓	5k	JR1	**11.03**	**0.338**

(b) Frozen backbone

Frozen backbone	FID ↓	CLIP ↑
PE-Spatial ViT-B	13.59	0.332
SigLIP2 ViT-B	13.37	0.334
DINOv3 ViT-S	15.14	0.328
DINOv3 ViT-B	12.08	0.335
DINOv3 ViT-L	16.33	0.325

(c) Regularizer

Regularizer	FID ↓	CLIP ↑
Pooled R1	12.08	0.335
Feature R1	13.65	0.327
Approx. R1	14.49	0.330
JR1 (ours)	**11.03**	**0.338**

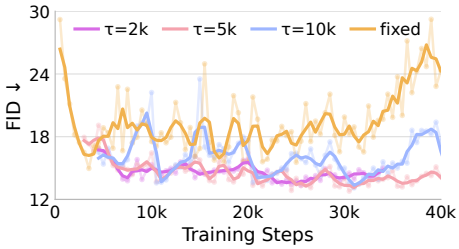


Figure 5: **Effect of refresh interval without gradient regularization.**

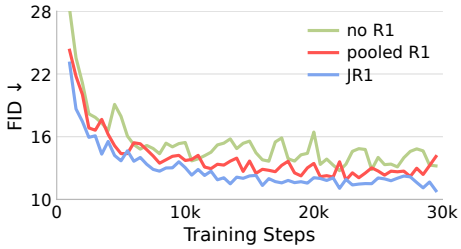


Figure 6: **Pooled R1 against Jacobian R1 (JR1).**

Frozen backbone. Tab. 4(b) contrasts feature extractors from several different foundation models. Among DINOv3 ViTs of different capacities, ViT-B is the strongest, while both the smaller ViT-S and the larger ViT-L underperform it. This suggests that a stronger representation does not necessarily translate into better GAN performance. PE-Spatial ViT-B and SigLIP2 ViT-B are close behind DINOv3 ViT-B, so ROAD is not tied to one feature extractor.

Regularizer. Tab. 4(c) compares JR1 with pooled R1, feature R1 that moves R1 penalty from image space to the frozen feature taps (Sauer et al., 2024b; Chong et al., 2024), and Approx. R1 (APT) that approximates the image-space penalty without a double backward pass (Lin et al., 2025). JR1 is the strongest of the four. It improves on pooled R1 from 12.08 to 11.03 FID, whereas both the feature-space and the approximate penalty fall behind pooled R1. Sec. E.4 further shows that JR1 consistently outperforms pooled R1 across a broad range of γ_J.

Trajectory of Refresh interval. Fig. 5 analyzes the effect of periodic refresh at $\tau \in \{2k, 5k, 10k\}$, without gradient regularization to isolate the refresh effect. Fixed mixing oscillates strongly and deteriorates late in training, whereas all refreshed variants remain markedly more stable and achieve better FID. More frequent refresh yields smoother trajectories, consistent with periodic re-drawing interrupting co-adaptation to a single random view. The best FIDs nevertheless remain close across intervals, indicating low sensitivity to τ. We use $\tau = 5k$ by default.

Trajectory of Jacobian R1. Fig. 6 contrasts no gradient regularization, pooled R1, and JR1 with refresh interval $\tau = 5k$. Pooled R1 yields a smoother FID trajectory than the unregularized run, and JR1 maintains a consistently lower FID with smaller fluctuations across training. This suggests that gradient regularization is complementary to our refresh mechanism; besides, the optimization benefits more from JR1 which regularizes the full logit sensitivities.

5 Conclusion and Limitations

We modernize Projected GAN for one-step diffusion distillation with a Lipschitz-controlled Transformer head, a periodic refresh of the random mixing, and a Jacobian R1 penalty. The resulting method, ROAD, converges fast and attains the best FID on both SDXL and SD3.5-M with a single adversarial objective and one training stage, whereas competing methods combine several objectives and stages. To our knowledge, ROAD is among the first to obtain a competitive one-step SD3.5-M generator. Its preference scores, however, still trail those of few-step competitors (Tab. 3). Text reaches the critic only through cross-attention, so an explicit text–image alignment term is a natural next step, though it would mean replacing DINOv3 with a contrastively trained encoder such as SigLIP2 (Tab. 4(b)). Scaling the frozen encoder does not help by itself either, since within DINOv3 ViT-B beats ViT-S and ViT-L (Tab. 4(b)). We attribute this to the two-player game's balance, which our refresh mechanism improves but does not settle, and addressing it directly is future work.

AI USE STATEMENT

In this work, we used generative AI tools to clean and reformat datasets, and provide feedback on experiments. We did not use generative AI tools to generate synthetic datasets; help develop theoretical models or conceptual frameworks; formulate mathematical claims; provide critical ingredients for proving mathematical claims; assist in the writing of proofs; propose or refine hypotheses; implement methods; or interpret results. Assistance with translation and support for qualitative and thematic data analysis were not applicable to this work.

Additionally, we used generative AI tools to modify scientific figures, create or edit software code, summarize or analyze existing literature, edit the research paper to improve readability, and format references.

We carefully reviewed all AI-assisted work. AI-assisted code modifications were manually inspected, debugged, and tested before being incorporated into the final implementation. We also verified all edited portions of the manuscript to ensure that our intended meaning was preserved. References formatted with AI assistance were carefully checked for formatting accuracy against the corresponding bibliographic information. We take responsibility for the final content of this work, including text, claims, code, and other artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Training configurations for both base models are listed in Sec. E.3 as extracted from the launch scripts, including teacher, batch, learning rates, head and mixing configuration, and refresh interval; unavailable launch-script fields are marked in the table. The shared evaluation protocol for SDXL and SD3.5-M, including prompt set, resolution, FID reference, and CLIP encoder, is given in Sec. E.2, together with the checkpoint-selection rule (best FID on a 500-step evaluation grid, CLIP from the same checkpoint). The JR1 estimator, including probe distribution, count and lazy scaling, is specified in Sec. C. Code and configurations will be released upon publication.

REFERENCES

Alexandre Araujo, Benjamin Negrevergne, Yann Chevaleyre, and Jamal Atif. On Lipschitz regularization of convolutional layers using Toeplitz matrix theory. In *AAAI*, volume 35, pp. 6661–6669, 2021.

Jordan T Ash and Ryan P Adams. On warm-starting neural network training. In *NeurIPS*, 2020.

Hmrishav Bandyopadhyay et al. SD3.5-Flash: Distribution-guided distillation of generative flows. *arXiv:2509.21318*, 2025.

Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Rasheed, Junke Wang, Marco Monteiro, Hu Xu, Shiyu Dong, Nikhila Ravi, Daniel Li, Piotr Dollár, and Christoph Feichtenhofer. Perception Encoder: The best visual embeddings are not at the output of the network. In *NeurIPS*, 2025.

Timothy I Cannings and Richard J Samworth. Random-projection ensemble classification. *Journal of the Royal Statistical Society Series B*, 2017.

Dar-Yen Chen, Hmrishav Bandyopadhyay, Kai Zou, and Yi-Zhe Song. NitroFusion: High-fidelity single-step diffusion through dynamic adversarial training. In *CVPR*, 2025.

Min Jin Chong, Krishna Kumar Singh, Yijun Li, Jingwan Lu, and David Forsyth. P2D: Plug and play discriminator for accelerating GAN frameworks. In *WACV*, 2024.

Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Diverse branch block: Building a convolution as an inception-like unit. In *CVPR*, pp. 10886–10895, 2021a.

Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. RepVGG: Making VGG-style convnets great again. In *CVPR*, pp. 13733–13742, 2021b.

Shibhansh Dohare, J Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A Rupam Mahmood, and Richard S Sutton. Loss of plasticity in deep continual learning. *Nature*, 632:768–774, 2024.

Xingtong Ge et al. SenseFlow: Scaling distribution matching for flow-based text-to-image distillation. In *ICLR*, 2026.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *NeurIPS*, 2023.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *ICCV*, pp. 1026–1034, 2015.

Judy Hoffman, Daniel A Roberts, and Sho Yaida. Robust learning with Jacobian regularization. *arXiv:1908.02729*, 2019.

Yiwen Huang, Aaron Gokaslan, Volodymyr Kuleshov, and James Tompkin. The GAN is dead; long live the GAN! A modern GAN baseline. In *NeurIPS*, 2024.

Drew A Hudson and C Lawrence Zitnick. Generative adversarial transformers. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4487–4499, 2021.

Michael F Hutchinson. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Communications in Statistics—Simulation and Computation*, 18(3):1059–1076, 1989.

Daniel Jakubovitz and Raja Giryes. Improving DNN robustness to adversarial attacks using Jacobian regularization. In *ECCV*, pp. 514–529, 2018.

Yifan Jiang, Shiyu Chang, and Zhangyang Wang. TransGAN: Two pure transformers can make one strong GAN, and that can scale up. In *NeurIPS*, 2021.

Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up GANs for text-to-image synthesis. In *CVPR*, 2023.

Minguk Kang, Richard Zhang, Connelly Barnes, Sylvain Paris, Suha Kwak, Jaesik Park, Eli Shechtman, Jun-Yan Zhu, and Taesung Park. Distilling diffusion models into conditional GANs. In *ECCV*, 2024.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *NeurIPS*, 2020a.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *CVPR*, 2020b.

Hyunjik Kim, George Papamakarios, and Andriy Mnih. The Lipschitz constant of self-attention. In *ICML*, 2021.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. In *NeurIPS*, 2023.

Kwonjoon Lee, Huiwen Chang, Lu Jiang, Han Zhang, Zhuowen Tu, and Ce Liu. ViTGAN: Training GANs with vision transformers. In *ICLR*, 2022.

Shanchuan Lin, Anran Wang, and Xiao Yang. SDXL-Lightning: Progressive adversarial diffusion distillation. *arXiv:2402.13929*, 2024.

Shanchuan Lin, Xin Yang, Yuxi Li, Xing Xia, et al. Diffusion adversarial post-training for one-step video generation. In *ICML*, 2025.

Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, et al. Swin transformer V2: Scaling up capacity and resolution. In *CVPR*, 2022.

Yanzuo Lu, Yuxi Ren, Shanchuan Lin, Xin Xia, 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 *ICCV*, 2025.

Weijian Luo, Tianyang Hu, Shifeng Zhang, Jiacheng Sun, Zhenguo Li, and Zhihua Zhang. Diff-Instruct: A universal approach for transferring knowledge from pre-trained diffusion models. In *NeurIPS*, 2023.

Vladimir A Marchenko and Leonid A Pastur. Distribution of eigenvalues for some sets of random matrices. *Mathematics of the USSR-Sbornik*, 1(4):457–483, 1967.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for GANs do actually converge? In *ICML*, 2018.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018.

Behnam Neyshabur, Srinadh Bhojanapalli, and Ayan Chakrabarti. Stabilizing GAN training with multiple random projections. *arXiv:1705.07831*, 2017.

Thuan Hoang Nguyen and Anh Tran. SwiftBrush: One-step text-to-image diffusion model with variational score distillation. In *CVPR*, 2024.

Evgenii Nikishin, Max Schwarzer, Pierluca D'Oro, Pierre-Luc Bacon, and Aaron Courville. The primacy bias in deep reinforcement learning. In *ICML*, 2022.

Roman Novak, Yasaman Bahri, Daniel A. Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Sensitivity and generalization in neural networks: an empirical study. In *ICLR*, 2018.

Jeeseung Park and Younggeun Kim. Styleformer: Transformer based generative adversarial networks with style vector. In *CVPR*, 2022.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024.

Xianbiao Qi, Jianan Wang, Yihao Chen, Yukai Shi, and Lei Zhang. LipsFormer: Introducing Lipschitz continuity to vision transformers. In *ICLR*, 2023.

Yuxi Ren, Xin Xia, Yanzuo Lu, Jiacheng Zhang, Jie Wu, Pan Xie, Xing Wang, and Xuefeng Xiao. Hyper-SD: Trajectory segmented consistency model for efficient image synthesis. In *NeurIPS*, 2024.

Kevin Roth, Aurelien Lucchi, Sebastian Nowozin, and Thomas Hofmann. Stabilizing training of generative adversarial networks through regularization. In *NeurIPS*, 2017.

Axel Sauer, Kashyap Chitta, Jonas Müller, and Andreas Geiger. Projected GANs converge faster. In *NeurIPS*, 2021.

Axel Sauer, Katja Schwarz, and Andreas Geiger. StyleGAN-XL: Scaling StyleGAN to large diverse datasets. In *SIGGRAPH*, 2022.

Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. StyleGAN-T: Unlocking the power of GANs for fast large-scale text-to-image synthesis. In *ICML*, 2023.

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*, 2024a.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *ECCV*, 2024b.

Christoph Schuhmann et al. LAION-5B: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022.

Hanie Sedghi, Vineet Gupta, and Philip M. Long. The singular values of convolutional layers. In *ICLR*, 2019.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *TMLR*, 2026.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR Workshop*, 2014.

Ghada Sokar, Rishabh Agarwal, Pablo Samuel Castro, and Utku Evci. The dormant neuron phenomenon in deep reinforcement learning. In *ICML*, 2023.

Nikita Starodubcev, Ilya Drobyshevskiy, Denis Kuznedelev, Artem Babenko, and Dmitry Baranchuk. Scale-wise distillation of diffusion models. In *ICLR*, 2026.

Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *WACV*, 2022.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv:2502.14786*, 2025.

Dániel Varga, Adrián Csiszárik, and Zsolt Zombori. Gradient regularization improves accuracy of discriminative models. *Schedae Informaticae*, 27:31–45, 2018.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *NeurIPS*, 2023.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Frédo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. In *NeurIPS*, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *CVPR*, 2024b.

Bowen Zhang, Shuyang Gu, Bo Zhang, Jianmin Bao, Dong Chen, Fang Wen, Yong Wang, and Baining Guo. StyleSwin: Transformer-based GAN for high-resolution image generation. In *CVPR*, 2022.

Long Zhao, Zhixing Zhang, Ting Chen, Dimitris N Metaxas, and Han Zhang. Improved transformer for high-resolution GANs. In *NeurIPS*, 2021.

Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient GAN training. In *NeurIPS*, 2020.

Bowen Zheng and Tianming Yang. Revisiting diffusion models: From generative pre-training to one-step generation. In *ICML*, 2025.

Mingyuan Zhou, Huangjie Zheng, Zhendong Wang, Mingzhang Yin, and Hai Huang. Score identity distillation: Exponentially fast distillation of pretrained diffusion models for one-step generation. In *ICML*, 2024.

Mingyuan Zhou, Yi Gu, Huangjie Zheng, Liangchen Song, Guande He, Yizhe Zhang, Wenze Hu, and Yinfei Yang. Score distillation of flow matching models. *arXiv:2509.25127*, 2025a.

Mingyuan Zhou, Zhendong Wang, Huangjie Zheng, and Hai Huang. Long and short guidance in score identity distillation for one-step text-to-image generation. In *ICLR*, 2025b.

Mingyuan Zhou, Huangjie Zheng, Yi Gu, Zhendong Wang, and Hai Huang. Adversarial score identity distillation: Rapidly surpassing the teacher in one step. In *ICLR*, 2025c.

Zhiming Zhou, Jiadong Liang, Yuxuan Song, Lantao Yu, Hongwei Wang, Weinan Zhang, Yong Yu, and Zhihua Zhang. Lipschitz generative adversarial nets. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7584–7593, 2019.

A LIPSCHITZ ANALYSIS OF THE TRANSFORMER HEAD

This appendix supports Sec. 3.1. We bound the sensitivity of one Transformer head, viewed as the map from the mixed tokens of its feature level to the dense logits of that level at a fixed prompt.

Setting. Fix a feature level ℓ and a prompt c. The head reads the mixed tokens $X = \hat{f}_\ell \in \mathbb{R}^{N_\ell \times d}$ of Eq. (2), where d is the token width, and returns the N_ℓ logits $D^\ell(X, c) \in \mathbb{R}^{N_\ell}$ of level ℓ. The head has B_ℓ pre-norm blocks (Sec. E.1). Each block contains a cosine self-attention branch followed by an FFN branch. The first block also contains a cosine cross-attention branch between the two. Its queries are the patch tokens, and its keys and values are computed from the N_c text tokens of c. Every branch applies RMSNorm to its input, passes the result through its sublayer, multiplies the output channel-wise by a learnable scale in $\mathbb{R}^d$, and adds the product to the input. We write $\alpha_{\mathrm{sa}}^{(b)}$ and $\alpha_{\mathrm{ffn}}^{(b)}$ for the residual scales of block b and α_{ca} for the cross-attention gate, and we set $a_{\mathrm{sa}}^{(b)} = \|\alpha_{\mathrm{sa}}^{(b)}\|_\infty$, $a_{\mathrm{ffn}}^{(b)} = \|\alpha_{\mathrm{ffn}}^{(b)}\|_\infty$, and $a_{\mathrm{ca}} = \|\alpha_{\mathrm{ca}}\|_\infty$. A final RMSNorm and the linear logit head of Sec. E.1 produce the logits.

We measure token matrices in the Frobenius norm, logit vectors in the Euclidean norm, and Jacobians in the operator norm induced by these two norms. We write $\mathrm{Lip}(f; \mathcal{S})$ for the Lipschitz constant of f on a set $\mathcal{S}$. The RMSNorm layers use $\varepsilon > 0$ and learnable weights w with $\|w\|_\infty \le \bar{w}$, and we set $c_{\mathrm{n}} = \bar{w}/\sqrt{\varepsilon}$ and $\nu = \sqrt{d}\,\bar{w}$. The FFN activation is GELU, and $c_{\mathrm{ffn}} = \sup_{t\in\mathbb{R}} |\mathrm{GELU}'(t)| \le 1.13$. Attention head h has temperature s_h, and its clamped parametrization (Sec. E.1) guarantees $0 < s_h \le s_{\max}$. In cross-attention head h, let $V_{c,h}$ be the value matrix computed from the text tokens of c, and let $\kappa_c = \max_h \|V_{c,h}\|_2$.

Lemma A.1 (Conditional Lipschitz bound for a Transformer head)**.** *Assume that every spectrally normalized linear map W of the head satisfies $\|W\|_2 \le 1$. Let $\mathcal{S} \subset \mathbb{R}^{N_\ell \times d}$ be a convex set such that, for every $X \in \mathcal{S}$, every query and key vector has norm at least $\rho > 0$ immediately before ℓ_2-normalization. Set*

$$c_{\mathrm{sa}} = \sqrt{N_\ell}\left(1 + \frac{s_{\max}\sqrt{N_\ell}\,\nu}{\rho}\right), \qquad c_{\mathrm{ca}} = \frac{s_{\max}\sqrt{N_c}\,\kappa_c}{2\rho}.$$

Then the Jacobian of $D^\ell(\cdot, c)$ has norm at most Λ_ℓ at every $X \in \mathcal{S}$, and $\mathrm{Lip}\big(D^\ell(\cdot, c); \mathcal{S}\big) \le \Lambda_\ell$, where

$$\Lambda_\ell = c_{\mathrm{n}}\big(1 + a_{\mathrm{ca}} c_{\mathrm{n}} c_{\mathrm{ca}}\big) \prod_{b=1}^{B_\ell} \big(1 + a_{\mathrm{sa}}^{(b)} c_{\mathrm{n}} c_{\mathrm{sa}}\big)\big(1 + a_{\mathrm{ffn}}^{(b)} c_{\mathrm{n}} c_{\mathrm{ffn}}\big). \tag{7}$$

The cross-attention factor is absent in a head without cross-attention. At initialization, the gate satisfies $a_{\mathrm{ca}} = 0$, and every residual scale equals $1/B_\ell$ (Sec. E.1). Since $(1 + t/B_\ell)^{B_\ell} \le e^t$ for $t \ge 0$, the bound at initialization satisfies $\Lambda_\ell \le c_{\mathrm{n}} \exp\big(c_{\mathrm{n}}(c_{\mathrm{sa}} + c_{\mathrm{ffn}})\big)$ for every depth B_ℓ. The residual scaling therefore keeps the initial bound below a constant that does not depend on the depth of the head.

Proof. All bounds below concern Jacobians at a point of $\mathcal{S}$, and $\delta(\cdot)$ denotes the first-order change caused by a perturbation δX. By the chain rule, the Jacobian norm of the head is at most the product of the Jacobian norms of its layers. Because $\mathcal{S}$ is convex, integrating this bound along segments in $\mathcal{S}$ gives the Lipschitz bound.

Step 1: RMSNorm and FFN. For one token $x \in \mathbb{R}^d$, let $r(x) = \sqrt{\|x\|_2^2/d + \varepsilon}$, so that $\mathrm{RMSNorm}(x) = w \odot x/r(x)$. Its Jacobian is

$$\frac{\mathrm{diag}(w)}{r(x)}\Big(I_d - \frac{x x^\top}{d\, r(x)^2}\Big).$$

The matrix in parentheses has eigenvalue 1 on $x^\perp$ and eigenvalue $\varepsilon/r(x)^2 \le 1$ along x. Since $r(x) \ge \sqrt{\varepsilon}$, the Jacobian has norm at most c_{n}. Since $\|x\|_2 < \sqrt{d}\, r(x)$, every output token has norm at most ν. RMSNorm acts on each token separately. On token matrices, its Jacobian also has norm at most c_{n}, and $\|\mathrm{RMSNorm}(X)\|_F \le \sqrt{N_\ell}\,\nu$. The FFN composes two linear maps of norm at most one with GELU, so its Jacobian has norm at most c_{ffn}.

Step 2: Cosine self-attention. Two primitives are shared with cross-attention. The map $u \mapsto u/\|u\|_2$ has Jacobian $(I - \widehat{u}\,\widehat{u}^\top)/\|u\|_2$ with $\widehat{u} = u/\|u\|_2$, whose norm is at most $1/\rho$ when $\|u\|_2 \geq \rho$. At a probability vector p, softmax has Jacobian $\mathrm{diag}(p) - pp^\top$, whose largest eigenvalue is at most $1/2$. Row-wise softmax therefore has Jacobian norm at most $1/2$.

Let $Z = \mathrm{RMSNorm}(X)$ be the input of the sublayer. For head $h = 1, \ldots, H$, let

$$Q_h = ZW_{Q,h}, \qquad K_h = ZW_{K,h}, \qquad V_h = ZW_{V,h}, \qquad P_h = \mathrm{softmax}(A_h),$$

where $A_h = s_h \widehat{Q}_h \widehat{K}_h^\top$, the matrices $\widehat{Q}_h$ and $\widehat{K}_h$ normalize each row, with rows $\widehat{q}_{h,i}$ and $\widehat{k}_{h,j}$, and softmax acts on each row. The sublayer output is $[P_1V_1, \ldots, P_HV_H]W_O$. Rotary position embeddings rotate each query and key by an orthogonal matrix, so they change none of the norms below. Each per-head projection is a column block of its full projection, so that, for example, $\sum_h \|\delta V_h\|_F^2 = \|\delta Z\, W_V\|_F^2 \leq \|\delta Z\|_F^2$. The perturbation of each head output splits into a value path and a weight path, $\delta(P_hV_h) = P_h\,\delta V_h + \delta P_h\,V_h$.

For the value path, each row of P_h is a probability vector, so $\|P_h\|_2 \leq \|P_h\|_F \leq \sqrt{N_\ell}$. The value path therefore contributes at most $\sqrt{N_\ell}\,\|\delta Z\|_F$. For the weight path, the normalized rows have unit norm, so $|\delta A_{h,ij}| \leq s_{\max}\big(\|\delta\widehat{q}_{h,i}\|_2 + \|\delta\widehat{k}_{h,j}\|_2\big)$. Squaring with $(a+b)^2 \leq 2a^2 + 2b^2$, summing over i, j and h, and applying the $1/\rho$ bound to each row gives

$$\sum_h \|\delta A_h\|_F^2 \leq \frac{4 s_{\max}^2 N_\ell}{\rho^2}\|\delta Z\|_F^2, \qquad \sum_h \|\delta P_h\|_F^2 \leq \frac{s_{\max}^2 N_\ell}{\rho^2}\|\delta Z\|_F^2,$$

where the second inequality uses the softmax bound. Since $\|V_h\|_2 \leq \|Z\|_F \leq \sqrt{N_\ell}\,\nu$, the weight path contributes at most $(s_{\max} N_\ell \nu/\rho)\|\delta Z\|_F$. The output projection W_O has norm at most one. Adding the two paths shows that the sublayer has Jacobian norm at most c_{sa}.

Step 3: Cosine cross-attention. The keys and values of cross-attention come from the fixed text tokens, so only the queries vary with X. The argument of Step 2 then gives $|\delta A_{h,ij}| \leq s_{\max}\|\delta\widehat{q}_{h,i}\|_2$ for each of the N_c keys j. Summing over keys, queries, and heads, and applying the $1/\rho$ and softmax bounds, gives $\sum_h \|\delta P_h\|_F^2 \leq \big(s_{\max}^2 N_c/(4\rho^2)\big)\|\delta Z\|_F^2$. The value matrices contribute the factor κ_c, and W_O has norm at most one, so the sublayer has Jacobian norm at most c_{ca}. Selecting the patch-token queries and writing the output back into the full token matrix are maps of norm one, so they leave this bound unchanged.

Step 4: Composition. The Jacobian of a branch is the identity plus the product of three factors: the channel-wise scale, the sublayer Jacobian, and the RMSNorm Jacobian. The first factor has norm $\|\alpha\|_\infty$, and the last has norm at most c_{n}. By Steps 1–3, the self-attention and FFN branches of block b therefore have Jacobian norms at most $1 + a_{\mathrm{sa}}^{(b)} c_{\mathrm{n}} c_{\mathrm{sa}}$ and $1 + a_{\mathrm{ffn}}^{(b)} c_{\mathrm{n}} c_{\mathrm{ffn}}$, and the cross-attention branch has Jacobian norm at most $1 + a_{\mathrm{ca}} c_{\mathrm{n}} c_{\mathrm{ca}}$. The final RMSNorm contributes c_{n}. The linear logit head maps each token to one logit with a vector of norm at most one, so it contributes at most one. The product of these factors is Λ_ℓ in Eq. (7). □

Scope of the bound. Lemma A.1 is conditional. The floor ρ is an assumption about the forward pass, and the architecture does not enforce it. The bound diverges as a query or key approaches zero before normalization. The RMSNorm weights and the residual scales are trainable, so Λ_ℓ changes during training through $\bar{w}$, a_{ca}, $a_{\mathrm{sa}}^{(b)}$, and $a_{\mathrm{ffn}}^{(b)}$. The lemma treats $\|W\|_2 \leq 1$ as exact. Spectral normalization with finitely many power iterations satisfies this condition only approximately, and any realized norm above one must be included in the bound of the corresponding sublayer. The bound concerns the head alone at a fixed prompt. Extending it to images requires, in addition, Jacobian bounds for the frozen encoder and the current random mixing. The constants are worst-case. For example, the RMSNorm Jacobian attains c_{n} only near the zero token, and the factors $\sqrt{N_\ell}$ come from Frobenius-norm accounting over all tokens. The lemma shows how spectral normalization, clamped cosine attention, RMSNorm, and residual scaling each enter a finite sensitivity budget. Tab. 1 evaluates the three controls of Sec. 3.1 jointly.

B PERSISTENT RANDOM MIXING: ANALYSIS AND DIAGNOSTICS

This appendix supports Sec. 3.2. We describe the implemented operator and its gain spectrum, make precise the sense in which a direction is weakly measured under a draw, and isolate the long-horizon consequence in a minimal measured-gap model. We report the empirical diagnostics in Sec. B.3–B.4 and state the limits of the account in Sec. B.5.

B.1 THE IMPLEMENTED RANDOM MIXING AND ANISOTROPY

Random mixing in matrix form. Eq. (2) composes CCM and CSM along a top-down path, but it leaves the two operators unspecified. We specify them here, and, because the mixing path carries no nonlinearity, we follow each one through to the matrix that Sec. 3.2 reasons about. The construction proceeds in three steps: the matrix of one convolutional branch, then the matrix of one mixing, then the matrix through which the head of a given level reads.

Setting. For both M_ℓ^{cc} and M_ℓ^{cs}, the patch tokens are reshaped to their two-dimensional grid. On a ViT backbone all L taps share that grid, so every level carries the same number N_{p} of patch positions, each of token width d. The summary token is carried by its own random linear map, and it enters neither the matrices below nor the analysis that follows, so f_ℓ and $\hat{f}_\ell$ denote the patch part of the level-ℓ features throughout this appendix. We vectorize a level channel-major, so the features of one level form a vector of dimension $N_{\mathrm{p}}d$.

Branch matrices. Each operator passes the grid through three independent, size-preserving $d \to d$ convolutions, whose 1×1, 3×3, and 5×5 kernels are drawn by fan-in-scaled Kaiming-uniform initialization (He et al., 2015). Write $\mathcal{M}(W)$ for the matrix of one such convolution with kernel W. A 1×1 kernel $W^{(1)} \in \mathbb{R}^{d\times d}$ recombines channels and leaves positions untouched, so $\mathcal{M}(W^{(1)}) = W^{(1)} \otimes I_{N_{\mathrm{p}}}$. A $k\times k$ kernel $W^{(k)} \in \mathbb{R}^{d\times d\times k\times k}$ gives the standard matrix of a multi-channel two-dimensional convolution (Araujo et al., 2021; Sedghi et al., 2019). That matrix is a $d \times d$ array of blocks, and its (c', c) block is the matrix $\mathcal{T}\big(W^{(k)}_{c'c}\big)$ of order N_{p} generated by the $k\times k$ slice $W^{(k)}_{c'c}$ on the patch grid,

$$\mathcal{M}\big(W^{(k)}\big) = \Big[\,\mathcal{T}\big(W^{(k)}_{c'c}\big)\,\Big]_{c',c=1}^{d} \in \mathbb{R}^{N_{\mathrm{p}}d\times N_{\mathrm{p}}d}.$$

The convolutions zero-pad, so each block is banded doubly block-Toeplitz (Araujo et al., 2021). Circular padding would instead make each block doubly block-circulant and would block-diagonalize the operator by the two-dimensional DFT, which is the standard idealization of a zero-padded layer (Sedghi et al., 2019). We invoke it only where the gain spectrum is needed, because decomposing the implemented operator directly is intractable at this width. Setting $k = 1$ recovers $\mathcal{T}\big(W^{(1)}_{c'c}\big) = W^{(1)}_{c'c} I_{N_{\mathrm{p}}}$, for which the two paddings agree exactly.

Mixing matrices. Each mixing averages its three branches, so

$$M_\ell^{\mathrm{cc}} = \tfrac{1}{3} \sum_{k\in\{1,3,5\}} \mathcal{M}\big(W^{(k)}\big),$$

and M_ℓ^{cs} takes the same form from its own independent draw. All of these parameters are frozen after sampling, and they are reinitialized together at every refresh. Both matrices are square of order $N_{\mathrm{p}}d$, so each mixing preserves the backbone's channel width. The three branches share their stride and their sliding-window alignment, so this average is itself a single 5×5 convolution. Zero-padding $W^{(1)}$ and $W^{(3)}$ to a 5×5 support and averaging the three kernels gives one kernel $\bar{W}$ with $\frac{1}{3}\sum_k \mathcal{M}\big(W^{(k)}\big) = \mathcal{M}(\bar{W})$ (Ding et al., 2021a;b). The branches therefore do not enlarge the operator class, and what they change is the law from which the 5×5 kernel is drawn. Fan-in scaling gives branch k an entry variance proportional to $1/(dk^2)$, so the taps of $\bar{W}$ are center-heavy: the central tap receives all three branches, the remaining 3×3 taps receive two, and the outer 5×5 ring receives one. Mixing therefore stays predominantly a channel mixer, and it couples positions through a decaying spatial tail.

The mixing matrix. Stacking the levels, $f = [f_1^\top, \dots, f_L^\top]^\top$ and $\hat{f} = [\hat{f}_1^\top, \dots, \hat{f}_L^\top]^\top$, the recursion of Eq. (2) unrolls into the affine map $\hat{f} = Mf + b$ used throughout the analysis. Its matrix is block

upper triangular in the level index,

$$M = \big[M_{\ell m}\big]_{\ell,m=1}^{L} \in \mathbb{R}^{d_f \times d_f}, \quad M_{\ell m} = \begin{cases} \big(M_\ell^{\mathrm{cs}} M_{\ell+1}^{\mathrm{cs}} \cdots M_m^{\mathrm{cs}}\big) M_m^{\mathrm{cc}}, & m \geq \ell, \\ 0, & m < \ell, \end{cases} \quad d_f = L N_{\mathrm{p}} d.$$

Every block is square of order $N_{\mathrm{p}}d$, so each level keeps the backbone's channel width. Two readings of this matrix matter later. Row ℓ vanishes to the left of the diagonal, which says that the head of level ℓ reads the backbone features of levels ℓ through L and of those alone. The diagonal block $M_{\ell\ell} = M_\ell^{\mathrm{cs}} M_\ell^{\mathrm{cc}}$ is the CCM–CSM composition at that level, and it dominates column ℓ, because every off-diagonal block reaching that column carries at least one further CSM factor. Since the diagonal blocks are almost surely invertible, so is M, and no feature direction is left unmeasured; what varies is how strongly each one is measured.

Anisotropy. For a unit direction η, we define its gain as $\lambda_M(\eta) = \|M\eta\|$. In the circular-padding idealization of the token-grid convolutions above (Sedghi et al., 2019), these gains follow the singular values of square random matrices (Marchenko & Pastur, 1967). With the implemented initialization, a single CCM or CSM has a median gain of 0.27, and $\sim 5\%$ of directions lie below one tenth of that median. The diagonal block $M_{\ell\ell}$, which composes CCM with CSM, broadens the spectrum further, to a median of 0.063 with $\sim 12\%$ of directions below one tenth of the median; these blocks set the scale of the spectrum of M, whose off-diagonal blocks are smaller by a further CSM factor each. Zero padding truncates the kernels at the grid border, so the implemented operator has medians about 4% and 9% below these two values, and its tail fractions are unchanged. The exact weak directions depend on the draw. The overall scale of the head gradient may change during training, but a fixed draw preserves the relative gain profile throughout the refresh interval.

B.2 Gain-weighted correction under fixed and refreshed draws

We isolate the effect of the mixing draw with a deliberately minimal surrogate. Let $\delta \in \mathbb{R}^{d_f}$ be a feature-space discrepancy between generated and real data, and let the generator reduce the measured gap $\frac{1}{2}\|M\delta\|^2$ by the gradient flow $\dot{\delta} = -M^\top M\delta$. Write $M = \sum_i \lambda_i a_i \psi_i^\top$ for its singular value decomposition. Because M is almost surely invertible (Sec. B.1), the ψ_i form an orthonormal basis of $\mathbb{R}^{d_f}$ and every λ_i is positive, so the expansions below are over a full basis.

Proposition B.1 (Correction under a fixed draw and refreshed draws)**.** (i) **Fixed draw.** *For a fixed M,*

$$\delta(t) = \sum_i e^{-\lambda_i^2 t} \langle \psi_i, \delta(0) \rangle \psi_i.$$

Hence, the component along ψ_i decays at rate λ_i^2.

(ii) **Independent refreshes.** *Suppose that independent draws $M^{(1)}, \ldots, M^{(K)}$ act for consecutive intervals of length τ. Let $A^{(k)} = \exp\big(-\tau (M^{(k)})^\top M^{(k)}\big)$, and assume that their law is right-isotropic in the sense that $\mathbb{E}[(A^{(k)})^\top A^{(k)}] = c(\tau) I$. Then*

$$\mathbb{E}\|\delta(K\tau)\|^2 = c(\tau)^K \|\delta(0)\|^2.$$

Thus, in expectation, no fixed direction retains a privileged slow rate across refresh periods.

Proof. Part (i). Because $M^\top M = \sum_i \lambda_i^2 \psi_i \psi_i^\top$, the coefficient of $\delta(t)$ along ψ_i satisfies $\dot{\delta}_i(t) = -\lambda_i^2 \delta_i(t)$. Solving each scalar equation gives the stated expansion.

Part (ii). By construction, $\delta(K\tau) = A^{(K)} \cdots A^{(1)} \delta(0)$. Condition on $y = A^{(K-1)} \cdots A^{(1)} \delta(0)$. Independence and the right-isotropy assumption give $\mathbb{E}[\|A^{(K)} y\|^2 \mid y] = y^\top \mathbb{E}[(A^{(K)})^\top A^{(K)}] y = c(\tau)\|y\|^2$. Iterating this identity over the K periods proves the result. □

Part (i) formalizes the fixed-draw selection described in Sec. 3.2. High-gain components decay first, so the surviving residual becomes enriched in weakly measured directions. The measured gap can therefore settle while a feature-space residual remains. Weak measurement here is a matter of degree rather than of kind: no direction is annihilated, because M is almost surely invertible, but a direction with a small λ_i is corrected at the far smaller rate λ_i^2 and can therefore survive an entire fixed-draw run.

Part (ii) isolates the effect of refresh. Each independent draw replaces the gain landscape, so in expectation no fixed direction is systematically attenuated over the whole run. The implemented convolutional law has exact permutation and sign symmetries over channel coordinates, and every level is read chiefly through the diagonal block that dominates its column, so the expected gain is comparable across levels. Right isotropy over the full feature space is the corresponding idealization, and we use it only to make this temporal distinction precise.

The surrogate predicts differential correction and stall. The nonlinear claim that realization-specific co-adaptation leaves new discrepancies uncorrected in weak directions and causes FID oscillation or degradation therefore remains a hypothesis. The refresh intervention in Sec. 3.2 validates this hypothesis.

B.3 Diagnostics

Definitions. For an image x, let $f_\ell(x) \in \mathbb{R}^{N_\ell \times d}$ denote the level-ℓ backbone features after register-token handling and before mixing, and let $\bar{D}(x) = \frac{1}{n}\sum_i D_i(x)$ be the mean critic logit. With the encoder detached, the gradient $\nabla_{f_\ell}\bar{D}(x)$ is the critic sensitivity in the shared backbone coordinates. We define the *Pre-projection Feature Utilization Coverage* (PFUC) profile by

$$u_c(x) = \sigma_c \sqrt{\tfrac{1}{N_\ell}\textstyle\sum_{j=1}^{N_\ell}\big(\partial \bar{D}(x)/\partial f_{\ell,j,c}(x)\big)^2}, \qquad U_c = \mathbb{E}_x[u_c(x)], \qquad \pi_c = U_c \Big/ \textstyle\sum_{c'} U_{c'},$$

where j indexes tokens, σ_c is the feature scale of channel c, computed once on a reference set and shared across runs, and the expectation is over a fixed diagnostic set of real and generated images. We quantify the change between two PFUC profiles by their total-variation distance,

$$\mathrm{TV}(\pi, \pi') = \tfrac{1}{2}\textstyle\sum_c |\pi_c - \pi'_c|,$$

which is the fraction of normalized sensitivity mass reassigned across channels. PFUC characterizes the breadth and stability of the critic's input-gradient sensitivity (Simonyan et al., 2014).

The *logit margin* is $m_t = r_t - q_t$, where r_t and q_t are the batch-mean values of $\bar{D}$ on the real and generated halves of the step-t batch, respectively. We apply ± 150-step smoothing only for display. Unlike accuracy, m_t retains the magnitude of the real–generated separation, and the separate traces of r_t and q_t reveal how that separation changes.

Measured values (Figs. 3 and 4). The two diagnostic runs use SDXL with an effective batch size of 32 and share the seeds, data, diagnostic set, and channel feature scales. They differ only in whether M is fixed or re-drawn every 5k steps. The PFUC heat maps display the first 64 of 768 channels at the deepest tap on a shared logarithmic color scale, with one profile every 500 steps; all TV values use the complete 768-channel profiles. In the refresh run, an additional profile is recorded immediately after each re-draw.

B.4 Longitudinal probes on a fixed-mixing run

The fixed-run measurements in Sec. B.3 establish a persistent critic state, but they do not determine whether its separation is specific to the training realization $M^{(0)}$. We therefore apply two post-hoc diagnostics to frozen checkpoints of the same run.

Fresh-draw transfer. At every 5k-step checkpoint from 5k to 30k, we hold the checkpoint, critic head, and matched real and generated batches fixed, and replace only $M^{(0)}$ by eight independently drawn mixings. The margin under $M^{(0)}$ remains 5.1–5.3 through 25k (4.5 at 30k), whereas the margins under fresh draws are approximately zero at every checkpoint. Because only the mixing changes, this collapse shows that the fixed head's real–generated separation does not transfer to a fresh measurement basis. This result directly supports its coupling to $M^{(0)}$.

Linear probe. We also fit a separate linear classifier to the frozen mixed features at each level to distinguish real from generated samples. This classifier is used only as a diagnostic and does not update the generator, backbone, mixing, or critic. Beginning at approximately 2.5k steps, real and generated samples are harder for a linear classifier to distinguish under $M^{(0)}$ than under fresh draws, particularly for the shallow-level features.

Over the same fixed-mixing run, FID oscillates and later worsens, whereas the margin under the fixed training mixing $M^{(0)}$ remains large and nearly stationary. The fresh-draw and linear-probe diagnostics show that this separation is specific to the pairing $D_\phi \circ M^{(0)}$.

B.5 Scope of the account

The account makes two claims at different strengths. First, Prop. B.1 shows in a linear surrogate why weak-direction residuals persist under a fixed draw and why independent draws remove a permanently privileged direction in expectation. The chain-rule bounds of Sec. 3.2 tie the visibility and correction of a direction to its gain. Second, the realization-specific co-adaptation of Sec. 3.2, in which prolonged optimization under the fixed gain landscape hardens both players around it until quality degrades, is a mechanistic reading of the ongoing nonlinear game. It is supported jointly by PFUC, the margin, the fresh-draw probes of Sec. B.4, and the fixed-versus-refreshed intervention, but none of these alone establishes it.

C Jacobian R1: Analysis and Estimation

This appendix supports Sec. 3.3. We compare pooled R1 and JR1 in the output space of the critic in Sec. C.1, analyze the Rademacher estimator of JR1 in Sec. C.2, and specify its minibatch implementation in Sec. C.3.

All quantities are evaluated at a fixed prompt c, critic parameters ϕ, and mixing M. We use the notation of Sec. 3.3, where $J(x) \in \mathbb{R}^{n\times d_x}$ is the input–logit Jacobian with rows $g_i(x)^\top$, and d_x is the dimension of the image x. Pooled R1 and JR1 in Eqs. (4) and (5) are the expectations of the per-image energies $q_{\mathrm{pool}}(x)$ and $q_{\mathrm{Jac}}(x)$ over real images, scaled by $\gamma/2$ and $\gamma_{\mathrm{J}}/2$. The Jacobian is taken through the entire critic, including the frozen encoder F and the mixing M. Freezing their parameters excludes them from optimization, but it does not remove them from the path of the input gradient. Because both energies depend on x only through $J(x)$, we also evaluate them at an arbitrary matrix $J \in \mathbb{R}^{n\times d_x}$ and write $q_{\mathrm{pool}}(J)$ and $q_{\mathrm{Jac}}(J)$. Both energies are built from averages over the n logits. We call this $1/n$ normalization the averaging convention, and the estimator and the minibatch penalty in Sec. C.2–C.3 retain it.

C.1 Jacobian R1 versus pooled R1

Sec. 3.3 makes three claims about pooled R1. First, pooled R1 penalizes only the direction spanned by $\mathbf{1}$ in the n-dimensional output space, and it leaves the $n-1$ orthogonal directions unpenalized. Second, a nonzero Jacobian can therefore incur zero pooled penalty. Third, the dense loss of Eq. (1) still depends on the sensitivities that pooling neglects. Parts (i)–(iii) of Prop. C.1 establish these claims in turn, and each part also states the corresponding property of JR1.

Fix an image x, and abbreviate $J = J(x)$ and $g_i = g_i(x)$. We use the orthogonal projectors onto $\mathrm{span}\{\mathbf{1}\}$ and onto its $(n-1)$-dimensional orthogonal complement $\mathbf{1}^\perp$, together with the mean gradient:

$$\Pi_\parallel = \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top, \qquad \Pi_\perp = I_n - \Pi_\parallel, \qquad \bar{g} = \tfrac{1}{n}\textstyle\sum_{i=1}^n g_i = \tfrac{1}{n}J^\top\mathbf{1}.$$

The r-th column $j_r = \partial D(x)/\partial x_r$ of J is the response of the logit vector to the r-th input coordinate. We call

$$q_\perp(x) = \tfrac{1}{n}\|\Pi_\perp J\|_F^2 = \tfrac{1}{n}\textstyle\sum_{r=1}^{d_x}\|\Pi_\perp j_r\|_2^2$$

the *neglected energy*. It measures how strongly input perturbations move the logit vector within the $n-1$ directions orthogonal to $\mathbf{1}$.

Proposition C.1 (Pooled R1 and JR1 in output space). (i) **Decomposition.** *The pooled energy satisfies* $q_{\mathrm{pool}}(x) = \frac{1}{n}\|\Pi_\parallel J\|_F^2$, *and*

$$q_{\mathrm{Jac}}(x) = q_{\mathrm{pool}}(x) + q_\perp(x), \qquad q_\perp(x) = \tfrac{1}{n}\textstyle\sum_{i=1}^n\|g_i - \bar{g}\|_2^2. \tag{8}$$

(ii) **Zero set.** *The pooled energy* $q_{\mathrm{pool}}(J)$ *vanishes exactly on the* $(n-1)d_x$*-dimensional subspace* $\mathcal{Z} = \{J \in \mathbb{R}^{n\times d_x} : J^\top\mathbf{1} = 0\}$. *In contrast,* $q_{\mathrm{Jac}}(J)$ *vanishes only at* $J = 0$.

(iii) **Dense-loss gradient.** *Let* $\ell_{\text{real}}(x) = \frac{1}{n}\sum_{i=1}^{n}\zeta(-D_i(x))$ *be the per-image real-image term of* $\mathcal{L}_D$ *in Eq. (1). Then*

$$\|\nabla_x \ell_{\text{real}}(x)\|_2^2 \le q_{\text{Jac}}(x). \tag{9}$$

In contrast, q_{pool} *admits no such bound. If the logits* $D_i(x)$ *are not all equal, then* $\|\nabla_x \ell_{\text{real}}(x)\|_2$ *is unbounded as* $J(x)$ *ranges over* $\mathcal{Z}$ *with* $D(x)$ *held fixed.*

Proof. Part (i). Since $\Pi_{\|}J = \mathbf{1}\bar{g}^\top$, every row of $\Pi_{\|}J$ equals $\bar{g}^\top$, and the i-th row of $\Pi_{\perp}J = J - \Pi_{\|}J$ equals $(g_i - \bar{g})^\top$. Hence $\frac{1}{n}\|\Pi_{\|}J\|_F^2 = \|\bar{g}\|_2^2 = q_{\text{pool}}(x)$, and $q_{\perp}(x) = \frac{1}{n}\sum_i \|g_i - \bar{g}\|_2^2$. The deviations $g_i - \bar{g}$ sum to zero. Averaging $\|g_i\|_2^2 = \|\bar{g} + (g_i - \bar{g})\|_2^2$ over i therefore cancels the cross terms and gives $q_{\text{Jac}}(x) = \|\bar{g}\|_2^2 + q_{\perp}(x)$.

Part (ii). By definition, $q_{\text{pool}}(J) = \frac{1}{n^2}\|J^\top \mathbf{1}\|_2^2$ vanishes if and only if $J^\top \mathbf{1} = 0$. This condition places each of the d_x columns of J in the $(n-1)$-dimensional subspace $\mathbf{1}^\perp$, so $\dim \mathcal{Z} = (n-1)d_x$. The energy $q_{\text{Jac}}(J) = \frac{1}{n}\|J\|_F^2$ is a scaled squared norm, so it vanishes only at $J = 0$.

Part (iii). Let $\eta \in \mathbb{R}^n$ collect the logit derivatives $\eta_i = \partial \ell_{\text{real}}/\partial D_i = -\frac{1}{n}\zeta'(-D_i(x))$. The chain rule gives $\nabla_x \ell_{\text{real}}(x) = J^\top \eta$. The derivative ζ' is the logistic function, which takes values in $(0, 1)$, so $\|\eta\|_2^2 < 1/n$. The spectral norm of J is at most its Frobenius norm. Therefore $\|J^\top \eta\|_2^2 \le \|J\|_F^2 \|\eta\|_2^2 \le \frac{1}{n}\|J\|_F^2 = q_{\text{Jac}}(x)$, which is Eq. (9). For the second statement, note that η depends only on $D(x)$, which is held fixed. The logistic function is strictly increasing. If the logits are not all equal, the entries of η are therefore not all equal, and $\eta_\perp = \Pi_\perp \eta \neq 0$. Fix a unit vector $a \in \mathbb{R}^{d_x}$, and let $J_t = t\,\eta_\perp a^\top$ for $t > 0$. Because $\mathbf{1}^\top \eta_\perp = 0$, we have $J_t^\top \mathbf{1} = 0$, and hence $J_t \in \mathcal{Z}$. Because $\eta_\perp^\top \eta = \|\eta_\perp\|_2^2$, we also have $J_t^\top \eta = t\|\eta_\perp\|_2^2\, a$, whose norm grows without bound as $t \to \infty$. □

C.2 Rademacher estimator: unbiasedness and variance

Let $v_1, \ldots, v_P \in \{-1, +1\}^n$ be independent probes, and let each probe have independent, equiprobable signs, as in Sec. 3.3. The per-image estimator replaces $\mathbb{E}_v$ in Eq. (6) by an average over the P probes:

$$\widehat{q}_P(x) = \frac{1}{nP}\sum_{p=1}^{P}\|J(x)^\top v_p\|_2^2.$$

Its factor $1/n$ follows the averaging convention, so $\widehat{q}_P(x)$ estimates $q_{\text{Jac}}(x)$. This is the trace estimator of Hutchinson (1989) for $\text{tr}\big(J(x)J(x)^\top\big) = \|J(x)\|_F^2$, which earlier work also used for Jacobian regularization (Varga et al., 2018; Hoffman et al., 2019). Prop. C.2 restates its standard properties under the averaging convention.

Proposition C.2 (Unbiasedness and variance)**.** *Fix an image* x*, and write* $g_i = g_i(x)$.

(i) **Unbiasedness.** $\mathbb{E}_v[\widehat{q}_P(x)] = q_{\text{Jac}}(x)$. *If* $J(x)$ *is differentiable in* ϕ*, then also* $\mathbb{E}_v[\nabla_\phi \widehat{q}_P(x)] = \nabla_\phi q_{\text{Jac}}(x)$.

(ii) **Variance.** *The variance over the probes is*

$$\text{Var}_v[\widehat{q}_P(x)] = \frac{2}{n^2 P}\sum_{i \neq j}\big(g_i^\top g_j\big)^2.$$

It vanishes if and only if the per-logit gradients are pairwise orthogonal.

(iii) **Relative variance.** *If* $J(x) \neq 0$*, then* $\text{Var}_v[\widehat{q}_P(x)] \le \frac{2}{P}\big(1 - \frac{1}{n}\big)\, q_{\text{Jac}}(x)^2$. *Equality holds when every* g_i *equals* $\pm g$ *for a common vector* $g \neq 0$.

Proof. For a single probe v, the identity $v_i^2 = 1$ gives

$$\|J(x)^\top v\|_2^2 = \sum_{i,j} v_i v_j\, g_i^\top g_j = \sum_i \|g_i\|_2^2 + 2\sum_{i<j} v_i v_j\, g_i^\top g_j.$$

Each product $v_i v_j$ with $i < j$ has zero mean and unit variance, and distinct products are uncorrelated.

Part (i). The cross terms have zero mean, so $\mathbb{E}_v\|J(x)^\top v\|_2^2 = \sum_i \|g_i\|_2^2 = n\, q_{\text{Jac}}(x)$. Averaging over the P probes and dividing by n gives the first identity. The probe law is uniform on the finite

set $\{-1,+1\}^n$ and does not depend on ϕ. The expectation $\mathbb{E}_v$ is therefore a finite average, which commutes with ∇_ϕ.

Part (ii). The cross terms are uncorrelated and have unit variance. A single probe therefore gives $\mathrm{Var}_v\|J(x)^\top v\|_2^2 = 4\sum_{i<j}(g_i^\top g_j)^2 = 2\sum_{i\neq j}(g_i^\top g_j)^2$. The P probes are independent, so averaging over them divides this variance by P. The factor $1/n$ divides it by n^2.

Part (iii). The Cauchy–Schwarz inequality gives $(g_i^\top g_j)^2 \le \|g_i\|_2^2\|g_j\|_2^2$ for every pair, and it also gives $\sum_i\|g_i\|_2^4 \ge \frac{1}{n}\big(\sum_i\|g_i\|_2^2\big)^2$. Hence

$$\sum_{i\neq j}(g_i^\top g_j)^2 \le \Big(\sum_i\|g_i\|_2^2\Big)^2 - \sum_i\|g_i\|_2^4 \le \Big(1-\frac{1}{n}\Big)\Big(\sum_i\|g_i\|_2^2\Big)^2 = \Big(1-\frac{1}{n}\Big)n^2 q_{\mathrm{Jac}}(x)^2.$$

Substituting this bound into part (ii) gives the claim. If every g_i equals $\pm g$, both inequalities hold with equality. □

Choice of Rademacher probes. Gaussian probes $v \sim \mathcal{N}(0, I_n)$ also satisfy $\mathbb{E}[vv^\top] = I_n$, so they also give an unbiased estimator. For Gaussian probes, however, the diagonal terms $v_i^2\|g_i\|_2^2$ fluctuate. At equal probe count, they add $\frac{2}{n^2P}\sum_i\|g_i\|_2^4$ to the variance of part (ii). Rademacher signs remove this term because $v_i^2 = 1$ holds exactly.

C.3 Minibatch implementation and normalization

Consider a minibatch $\mathcal{B}$ of real image–prompt pairs (x, c). For each pair, we draw P probes $v_{x,1}, \ldots, v_{x,P}$ with the law of Sec. C.2. The probes are independent across pairs, and they are drawn afresh at every evaluation of the penalty. The minibatch penalty is

$$\widehat{R}_1^{\mathrm{J}} = \frac{\gamma_{\mathrm{J}}}{2|\mathcal{B}|}\sum_{(x,c)\in\mathcal{B}}\widehat{q}_P(x) = \frac{\gamma_{\mathrm{J}}}{2n|\mathcal{B}|P}\sum_{(x,c)\in\mathcal{B}}\sum_{p=1}^{P}\big\|\nabla_x\big(v_{x,p}^\top D(x)\big)\big\|_2^2, \tag{10}$$

where $D(x)$ is evaluated at the prompt c paired with x. The second form uses the identity $J(x)^\top v = \nabla_x\big(v^\top D(x)\big)$. By Prop. C.2(i), the expectation of $\widehat{R}_1^{\mathrm{J}}$ over the probes and over minibatches drawn from p_r is R_1^{J} of Eq. (5). The same holds for its gradient with respect to ϕ. The probes are independent across images, so the probe noise in Eq. (10) is averaged over the $|\mathcal{B}|$ images as well as over the P probes.

Cost and order of operations. Each term of Eq. (10) is one vector–Jacobian product (VJP). A single backward pass of the scalar $v_{x,p}^\top D(x)$ through the critic returns $J(x)^\top v_{x,p}$. The estimator therefore costs P VJPs per image, whereas the exact penalty costs n VJPs per image (Sec. 3.3). Pooled R1 in Eq. (3) backpropagates the scalar $\frac{1}{n}\mathbf{1}^\top D(x)$ in the same way. With $P = 1$, the two penalties therefore have the same cost, and they differ only in the weights of the logit sum. Each probe gradient must be squared before the average over probes is taken. Averaging the P probe gradients first would give an estimate with expectation $q_{\mathrm{Jac}}(x)/P$.

Lazy scaling and reported coefficients. We apply the lazy regularization of Sec. 2 (Karras et al., 2020b). On every k_{lazy}-th critic update, the critic objective $\mathcal{L}_D + R_D$ uses $R_D = k_{\mathrm{lazy}}\widehat{R}_1^{\mathrm{J}}$. On all other critic updates, $R_D = 0$. Both configurations in Tab. 5 use $P = 1$ and $k_{\mathrm{lazy}} = 16$. The table reports γ_{J} as the coefficient of the averaged penalty in Eq. (5), before the factor k_{lazy} is applied. The reported values, $\gamma_{\mathrm{J}} = 7.5\times10^{-3}$ for SDXL and $\gamma_{\mathrm{J}} = 7.5\times10^{-4}$ for SD3.5-M, therefore require no conversion.

D Related Work

Few-step distillation of text-to-image diffusion models. ***Multi-objective or multi-stage training.*** The existing methods mostly use multiple training objectives and often consist of multiple training stages. Hyper-SD employs a six-phase hybrid training pipeline: four stages of trajectory-segmented consistency distillation, followed by human-feedback optimization and DMD-based one-step score distillation (Ren et al., 2024). For the one-step SDXL model, DMD2 uses a two-stage pipeline consisting of brief regression pretraining followed by joint distribution-matching and adversarial training stabilized by a two-timescale update rule (Yin et al., 2024a, Sec. F.4); DMDX is also a two-stage pipeline, consisting of adversarial distillation pretraining on teacher-generated ODE pairs with hybrid latent- and pixel-space discriminators, along with adversarial distribution-matching fine-tuning (Lu et al., 2025). SiDA jointly trains a one-step generator with score-identity and adversarial losses, reusing the fake score network's encoder as a parameter-free discriminator (Zhou et al., 2025c). ADD (SDXL-Turbo) uses single-stage joint training with teacher-based score distillation and a DINOv2-feature adversarial loss (Sauer et al., 2024b). SDXL-Lightning employs a five-stage progressive distillation pipeline: one MSE-based trajectory-distillation stage followed by four adversarial stages (Lin et al., 2024). NitroFusion uses dynamic adversarial refinement with a pool of conditional and unconditional global–local discriminators, *relying on a DMD2 one-step student and four-step teacher (Chen et al., 2025), so separate score-based pretraining is a prerequisite*. SenseFlow combines score-based distribution matching, trajectory-based intra-segment guidance, and timestep-aware VFM adversarial supervision (Ge et al., 2026). SwD produces a progressive few-step generator that increases latent resolution across denoising steps, combining patch-level MMD with score-based distribution matching and adversarial loss (Starodubcev et al., 2026). ***Adversarial objective as a single loss.*** Both APT (Lin et al., 2025) and LADD (Sauer et al., 2024a) perform latent-space adversarial training and derive *multilevel discriminator features from pretrained diffusion models*. LADD freezes the diffusion teacher, attaches discriminator heads to its intermediate features, and uses teacher-generated samples as real targets. APT initializes a separate discriminator from the pretrained diffusion model, adds multilevel output heads, and fine-tunes it against real data. D2O (Zheng & Yang, 2025) converts a pretrained diffusion U-Net into a one-step generator through pure adversarial fine-tuning; it is evaluated *only on low-resolution unconditional and class-conditional benchmarks and does not address text-to-image synthesis*. Unlike APT and LADD, both of which use diffusion-based critics—trainable in APT and frozen in LADD—ROAD uses a frozen foundation model (e.g., DINOv3) with lightweight, prompt-conditioned Transformer heads. Compared with D2O, we propose Transformer heads, refreshed mixing, and JR1, thereby scaling pure adversarial training to large-scale text-to-image generation.

Projected GAN and adjacent critic architectures. ***Convolutional heads over frozen encoders.*** Projected GAN trains shallow convolutional heads over a frozen pretrained encoder, whose features are scattered by fixed random cross-channel and cross-scale mixings (Sauer et al., 2021); StyleGAN-XL scales it to ImageNet (Sauer et al., 2022), inheriting Projected GAN's heads, and StyleGAN-T to text-to-image with 1-D convolutional heads over frozen ViT tokens (Sauer et al., 2023). Every head in this line is convolutional, the only reported head comparison being 1-D versus 2-D convolution (Sauer et al., 2023), and positions are scored independently, so the critic is vector-valued, a property ROAD keeps and regularizes. As far as we know, no work in this line attaches a Transformer head to frozen features. ***Transformer discriminators from scratch.*** ViTGAN scores a ViT on raw pixels from the class token, stabilized by L_2 attention with tied query and key projections, and applies spectral normalization only to initialize the linear weights (Lee et al., 2022). It reports that the vanilla ViT discriminator trains markedly worse than the stabilized one. TransGAN's convolution-free discriminator ingests patches of several sizes (Jiang et al., 2021). Other work pairs Transformer generators with convolutional discriminators or embeds attention inside a CNN (Zhang et al., 2022; Zhao et al., 2021; Park & Kim, 2022; Hudson & Zitnick, 2021). Our Lipschitz controls build on analyses of attention (Kim et al., 2021; Qi et al., 2023), scaled cosine attention (Liu et al., 2022), and spectral normalization (Miyato et al., 2018); their combination into the projected critic's ViT head with an explicit bound (Lemma A.1) and its conditioning on individual prompt tokens are new.

Random feature mixing and time-varying discriminators. ***Fixed random mixing.*** Projected GAN samples the mixing once at initialization and holds it fixed for the whole run (Sauer et al., 2021), and every method in that line inherits the choice; the consequence of one draw persisting through long training, a critic that specializes to it, is, to our knowledge, unexamined. Previously, multiple-random-projection GANs train simultaneous discriminators on a fixed ensemble of input

projections (Neyshabur et al., 2017), as do random-projection ensembles for classification (Cannings & Samworth, 2017); in both, the set of views is fixed in advance. ***Renewing trainable parameters.*** The closest precedents do alter the discriminator over time. NitroFusion keeps a pool of trainable heads, samples a subset per iteration, and re-initializes a small fraction of it, and SD3.5-Flash follows with stochastic head renewal (Chen et al., 2025; Bandyopadhyay et al., 2025); SDXL-Lightning resets its discriminator between progressive-distillation stages rather than within one (Lin et al., 2024). *Outside GANs,* re-initializing trainable parameters is the standard remedy for lost plasticity, through last-layer resets, recycling of dormant neurons, shrink-and-perturb, and continual backpropagation (Nikishin et al., 2022; Sokar et al., 2023; Ash & Adams, 2020; Dohare et al., 2024). Our ROAD instead re-draws the frozen measurement basis, the random mixings, at deterministic intervals while every trainable parameter persists, so it targets not lost plasticity in the head but the head's specialization to a single persistent realization.

Discriminator gradient regularization. ***Zero-centered penalties in GANs.*** Zero-centered discriminator-gradient penalties stabilize GAN training and admit local convergence analyses under suitable assumptions (Roth et al., 2017; Mescheder et al., 2018), and R3GAN combines R1 and R2 (Huang et al., 2024). D2O applies pooled R1 to the averaged logit (Zheng & Yang, 2025), ADD and P2D regularize in feature space (Sauer et al., 2024b; Chong et al., 2024), and APT uses a finite-difference approximation (Lin et al., 2025). Each constrains a scalar critic output, or an aggregate of a vector-valued one, which is the assumption our ROAD removes. ***Jacobian regularization in classification.*** Novak et al. (2018) study the empirical relationship between input–output Jacobian sensitivity and generalization, whereas Jakubovitz & Giryes (2018) explicitly regularize the logit Jacobian to improve adversarial robustness, and Varga et al. (2018); Hoffman et al. (2019) develop efficient randomized Jacobian regularization. We apply a squared Jacobian-norm penalty to real images in a dense adversarial game, motivated specifically by the limitation of pooled R1, which aggregates the vector-valued logit (Sec. 3.3). Through Rademacher Hutchinson estimation (Hutchinson, 1989) we recover the Jacobian energy in expectation with one input VJP per probe.

E Implementation Details and Additional Experiments

E.1 Critic design

This subsection specifies the critic heads of Sec. 3.1. The critic has one head per feature level, so it has $L = 4$ heads (Sec. 2). Head ℓ reads the mixed tokens $\hat{f}_\ell$ of Eq. (2), which the random mixing computes from the features of the frozen encoder F (Sec. B.1).

Level and depth schedules. Let L_F be the number of blocks of the frozen encoder, indexed from 0 to $L_F - 1$. The features f_ℓ of Eq. (2) are the tokens output by block t_ℓ of F, and the head of level ℓ has B_ℓ Transformer blocks, where

$$t_\ell = \left\lfloor \frac{\ell L_F}{L} \right\rfloor - 1, \qquad B_\ell = \min\big(L - \ell + 1,\, 4\big), \qquad \ell = 1, \ldots, L.$$

The first rule divides F into L contiguous spans of equal length and taps the last block of each span, so the deepest level always taps the final block, $t_L = L_F - 1$. The second rule gives the deepest level a single block, because its tokens are already highly semantic and need only light adaptation. The depth then increases by one per level toward the shallowest level, which receives the most reorganization. The cap of four blocks keeps the heads shallow and limits the number of factors in the Lipschitz bound of Eq. (7). For the configuration used throughout the paper, DINOv3 ViT-B with $L_F = 12$, the taps are $(2, 5, 8, 11)$, and the depths are $(B_1, \ldots, B_4) = (4, 3, 2, 1)$, which gives ten trainable blocks in total. Applied to a 24-block ViT-L, the same rule gives the taps $(5, 11, 17, 23)$. The depths depend only on L, so they are unchanged.

Transformer block. A block follows the pre-norm Transformer design. It contains a self-attention branch followed by an FFN branch, and the first block of every head contains a cross-attention branch between the two. The block departs from the standard design in four ways. First, RMSNorm replaces LayerNorm. Second, queries and keys have width $d/2$, where d is the token width. Third, the FFN uses GELU with an expansion ratio of 2 instead of 4. Fourth, self- and cross-attention use cosine attention with a clamped temperature. All learned linear maps of the head are spectrally normalized (Sec. 3.1), and token positions are encoded with rotary position embeddings (Tab. 5). Sec. A analyzes the resulting head.

Cross-attention. The cross-attention branch has the same cosine form. Its queries are the patch tokens of the level, and its keys and values are computed from the prompt. The summary token is excluded from the queries, so conditioning acts on the patch tokens only. The token sequences of all text encoders are concatenated along the channel axis into a single sequence and normalized without affine parameters. Two spectrally normalized linear maps then project this sequence to the widths of the visual queries and values. Each level has its own pair of text projections, so different levels can read the same prompt differently.

Attention temperature. Each attention head h has a learnable temperature s_h. It is stored in the log domain, initialized at $\log s_{\text{init}}$, and clamped at $\log s_{\max}$ before exponentiation, with $s_{\text{init}} = 10$ and $s_{\max} = 100$. This parametrization guarantees $0 < s_h \leq s_{\max}$. Queries and keys are ℓ_2-normalized, the normalized queries are multiplied by s_h, and the result is passed to scaled dot-product attention. Normalization, scaling, and attention are computed in single precision and cast back to the training precision. This keeps the clamp exact under mixed-precision training.

Residual scales and gate. Each self-attention and FFN branch of block b has its own learnable per-channel scale, $\alpha_{\text{sa}}^{(b)} \in \mathbb{R}^d$ or $\alpha_{\text{ffn}}^{(b)} \in \mathbb{R}^d$. For example, the self-attention branch maps its input X to $X + \alpha_{\text{sa}}^{(b)} \odot \text{SA}\big(\text{RMSNorm}(X)\big)$, where SA is the cosine self-attention sublayer and the product acts on the channels of every token. Both scales are initialized to the constant $1/B_\ell$. A head of B_ℓ blocks therefore starts as a small perturbation of the identity whose size is set by its depth, and Sec. A shows that this initialization keeps the bound of Eq. (7) below a constant that does not depend on the depth. The cross-attention branch has the same form with a gate α_{ca} in place of the residual scale. The gate is initialized to zero, so conditioning contributes nothing at initialization. The output projection of the cross-attention branch is initialized normally (Sec. E.3), so the gate receives a nonzero gradient from the first update.

Logit head. Every head ends in a final RMSNorm and a linear logit head, which applies one spectrally normalized linear map to each token independently. Each patch token and the summary token therefore yields one logit. Level ℓ thus emits N_ℓ logits, and concatenating the L levels gives the $n = \sum_\ell N_\ell$ logits of Eq. (1). We also implemented a convolutional logit head. It reshapes the patch tokens to their spatial grid and applies consecutive spectrally normalized 4×4 grouped convolutions, which reduce the grid to a 5×5 logit map. In this head, the summary token can contribute one further logit through its own linear map. In our experiments, the two logit heads performed comparably. We therefore report only the linear head, which is also the head analyzed in Sec. A.

E.2 Dataset and evaluation protocols

Real-image training set. We use the fixed dataset of 500K real image–caption pairs constructed by DMD2 (Yin et al., 2024a) from LAION-Aesthetics 5.5+ (Schuhmann et al., 2022) for both SDXL and SD3.5-M. The dataset discards unsafe images and images whose width or height is below 1024 pixels. Model-specific decoding, resizing, and feature extraction are applied only after sampling from this common pool.

COCO evaluation sets. For both models, we follow the DMD2 SDXL protocol without re-sampling its prompts or reference images. We use the fixed draw of 10K distinct images from the COCO-2014 validation set and one associated caption per image, as stored in the `coco10k/all_prompts.pkl` file and paired `coco10k/subset` directory. We generate one 1024×1024 image per prompt. The prompt order and sampling seeds are fixed and shared by all methods.

FID and CLIP score. We compute FID with the DMD2 implementation of `clean-fid` in its clean mode. Before Inception feature extraction, the longer spatial dimension is center-cropped to form a square and the image is resized to 512×512 with Lanczos interpolation. The resulting FID compares 10K generated images with their corresponding 10K real images. We compute CLIP score with the DMD2 evaluator for both models. It center-crops each generated image, resizes it to 256×256, and applies the OpenCLIP ViT-g/14 model pretrained on LAION-2B (`laion2b_s12b_b42k`). The text encoder receives the original caption prefixed by "A photo depicts". We report the mean cosine similarity over all generated image–caption pairs.

Auxiliary metrics. All auxiliary COCO metrics use the same generated images and the unmodified captions used by the corresponding FID evaluation. We report the mean raw score of LAION-Aesthetics Predictor V2, the official HPSv2 score (Wu et al., 2023), and the score produced by `yuvalkirstain/PickScore_v1` (Kirstain et al., 2023). HPSv2 and PickScore are shown on the $\times 100$ scale used in the tables. ImageReward is the mean raw score from `ImageReward-v1.0` (Xu et al., 2023). GenEval (Ghosh et al., 2023), reported only for SD3.5-M, is evaluated separately because it defines its own prompt suite. We use the released 553 prompts, generate four images per prompt, run the official Mask2Former/OpenCLIP evaluator with its default thresholds, and report the unweighted mean accuracy over its six compositional tasks.

Checkpoint selection. We evaluate ROAD checkpoints every 500 training steps and select the checkpoint with the lowest FID. Every other metric in a row is computed from that same checkpoint; we never select a separate checkpoint for alignment, aesthetics, human preference, or GenEval. Public baselines are evaluated with the same prompt files, reference images, preprocessing, metric implementations, and reporting scales as the corresponding ROAD model.

E.3 Training hyperparameters

Tab. 5 lists the training configurations of SDXL and SD3.5-M. On SDXL the critic scores the summary (cls) token alongside the patch tokens, the register tokens of DINOv3 are dropped, the cross-attention gate is zero-initialized while its output projection is normally initialized, and the critic and student are updated at a 1:1 ratio. Each of the four projected feature levels receives its own head, with depths decreasing from shallow to deep, $B_\ell{=}(4, 3, 2, 1)$: shallower backbone representations receive more trainable reorganization steps, while already highly semantic deep representations need only light adaptation; every head ends in a token-wise linear logit map (block structure in Sec. A). SD3.5-M uses the same head design as SDXL.

Table 5: **Training configurations of SDXL and SD3.5-M.**

Setting	SDXL	SD3.5-M
Teacher model	SDXL-Base-1.0	SD3.5-Medium
Teacher arch.	UNet	MMDiT
Conditioning timestep / σ	$t{=}999$	$\sigma{=}1.0$
GPUs $\times$ batch	$8{\times}8 = 64$	$8{\times}4 = 32$
G learning rate	$5{\times}10^{-7}$	$5{\times}10^{-7}$
D learning rate	$5{\times}10^{-6}$	$5{\times}10^{-6}$
Optimizer	AdamW, $\beta{=}(0, 0.99)$, no weight decay	AdamW, $\beta{=}(0, 0.99)$, no weight decay
Gradient clipping (G / D)	10 / 10	1.0 / 10.0, global L_2 norm
Position / linear norm	RoPE / SN	RoPE / SN
Refresh interval τ	5k	5k
Gradient penalty	$\gamma_{\mathrm{J}}{=}7.5{\times}10^{-3}$ with one Rademacher probe; lazy interval $k_{\mathrm{lazy}}{=}16$.	$\gamma_{\mathrm{J}}{=}7.5{\times}10^{-4}$ with one Rademacher probe; lazy interval $k_{\mathrm{lazy}}{=}16$.

E.4 EXTRA EXPERIMENTS

Random mixing configuration	FID ↓	CLIP ↑
Projected GAN	14.80	0.328
1×1+3×3	14.48	0.328
3×3+5×5	13.35	0.333
1×1+3×3+5×5	**11.03**	0.338

Table 6: Multi-branch mixing design. SDXL ablations that differ only in the composition of the random mixings M.

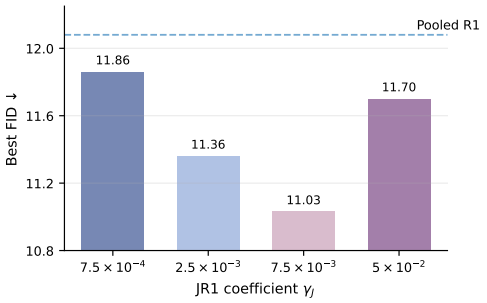


Figure 7: JR1 coefficient sensitivity on SDXL. Bars report the best FID for each γ_J; the dashed line denotes the pooled-R1 baseline.

Analysis of multi-branch random mixing. The completed rows isolate the contribution of receptive field. Both pairs average two supports and differ only in which ones, yet 3×3+5×5 reaches 13.35 against 14.48 for 1×1+3×3, which suggests extending the mixing over the token grid buys more than retaining the original channel-only map. The triple then improves on both pairs by at least 2.32 FID, a margin larger than the 1.13 separating the pairs themselves, so the three kernels are complementary rather than redundant: each scatters evidence over a different extent, and averaging them turns one random map into a mixture of maps with different supports.

Discriminator augmentation under fixed mixing. Late-stage degradation can resemble discriminator overfitting, for which data augmentation is a standard remedy. So we examine DiffAugment (Zhao et al., 2020) and adaptive discriminator augmentation (ADA) (Karras et al., 2020a) to test whether these remedies alleviate the observed degradation.

We follow the SDXL protocol of Sec. E.3 and compare fixed mixing with no augmentation, DiffAugment, or ADA, alongside the 5k-refresh reference. All configurations share the Transformer critic with pooled R1, training data, optimization settings, and a 40k-step budget. Performance is evaluated with the shared COCO-10K protocol (Sec. E.2).



Figure 8: **Augmentation under fixed mixing.** Representative discriminator anti-overfitting interventions do not materially alleviate late-stage FID degradation.

Mixing / augmentation	Best FID ↓
Fixed	14.54
Fixed + DiffAugment	14.46
Fixed + ADA	14.38
Refresh (5k)	12.08

Table 7: **Best FID on COCO-10K.** Augmentation controls under fixed mixing, with 5k refresh for reference. Unaugmented results are from Tab. 4(a).

The FID trajectories (Fig. 8) show that neither augmentation materially alleviates late-stage degradation under fixed mixing; Tab. 7 reports the best FID. The persistence of this failure mode despite these standard remedies motivates further investigation (Sec. 3.2).

Sensitivity to the JR1 coefficient. We scan the JR1 coefficient (γ_J) on SDXL and compare the best FID of each setting with the pooled-R1 baseline. Every tested JR1 coefficient improves over pooled R1. The best result is obtained at $\gamma_J = 7.5 \times 10^{-3}$, while the overall variation across the scanned range remains modest, suggesting that the method is not sensitive to this hyperparameter.

F Qualitative Results

F.1 Prompts for Qualitative Figures

We use the following prompts for Fig. 1, ordered by the top-left corner of each image, from top to bottom and then left to right:

- A steampunk woman wearing brass goggles stands by a sunlit window in an ornate Victorian room, beside a large vintage mechanical instrument.
- bedroom melting into the galaxy, penthouse apartment bedroom with big windows, unsplash, photo of the day
- mystical fantasy cat of many colors
- Ship in the form of a fish tank with fish and corals
- crowd of people on top of a skyscraper at night standing near a spotlight cityscape around them, from high angle looking down on rooftop, bladerunner
- Photo English country palace of aristocrats on a background of nature painted in watercolor on paper
- a unicorn standing in a meadow, morning sun in the background.
- concept art of a handsome and regal looking evil sorcerer with a circlet of thorns on his head, magic and lightning
- A glass teapot filled with blooming jasmine tea on a wooden table by a rainy window.
- A teddy bear reading a newspaper on a park bench.

The prompts for the SDXL comparison in Fig. 11, from top to bottom, are:

- isometric beach scenery
- Russian-inspired building with towers and spires, smokestacks spewing smoke in the background against a red sky.
- A cute mouse with a dress in a kitchen, fantasy, cinematic, door
- An old man playing chess against a robot.
- steampunk chameleon automaton bot, close up, realistic, highly detail
- An oil painting of a young woman dressed in a flowing, white dress sitting on a chair with a red drape. An easel rests on her knees and she is evidently drawing. She is gazing directly at the observer.

The prompts for the SD3.5-M comparison in Fig. 12, from top to bottom, are:

- an anthropomorphic owl courier, monochromatic with subtle red accents
- A futuristic city with flying cars, tall glass towers covered in plants, and a large park in the center, bright daylight.
- A bookshelf filled with colorful books, a potted plant and a small globe.
- A photo of an cosplay girl dressed as a fantasy warrior, with a full suit of armor and a sword at her side, standing on a hilltop with a dramatic sunset behind her
- bedroom melting into the galaxy, penthouse apartment bedroom with big windows, unsplash, photo of the day
- A teddy bear reading a newspaper on a park bench.

F.2 Qualitative Samples from ROAD

Figures 9 and 10 show qualitative generations from ROAD distilled on SDXL and SD3.5-M, respectively.



Figure 9: **ROAD samples on SDXL.** Qualitative generations from the distilled model.



Figure 10: **ROAD samples on SD3.5-M.** Qualitative generations from the distilled model.

F.3 Qualitative Comparison with Other Distillation Methods

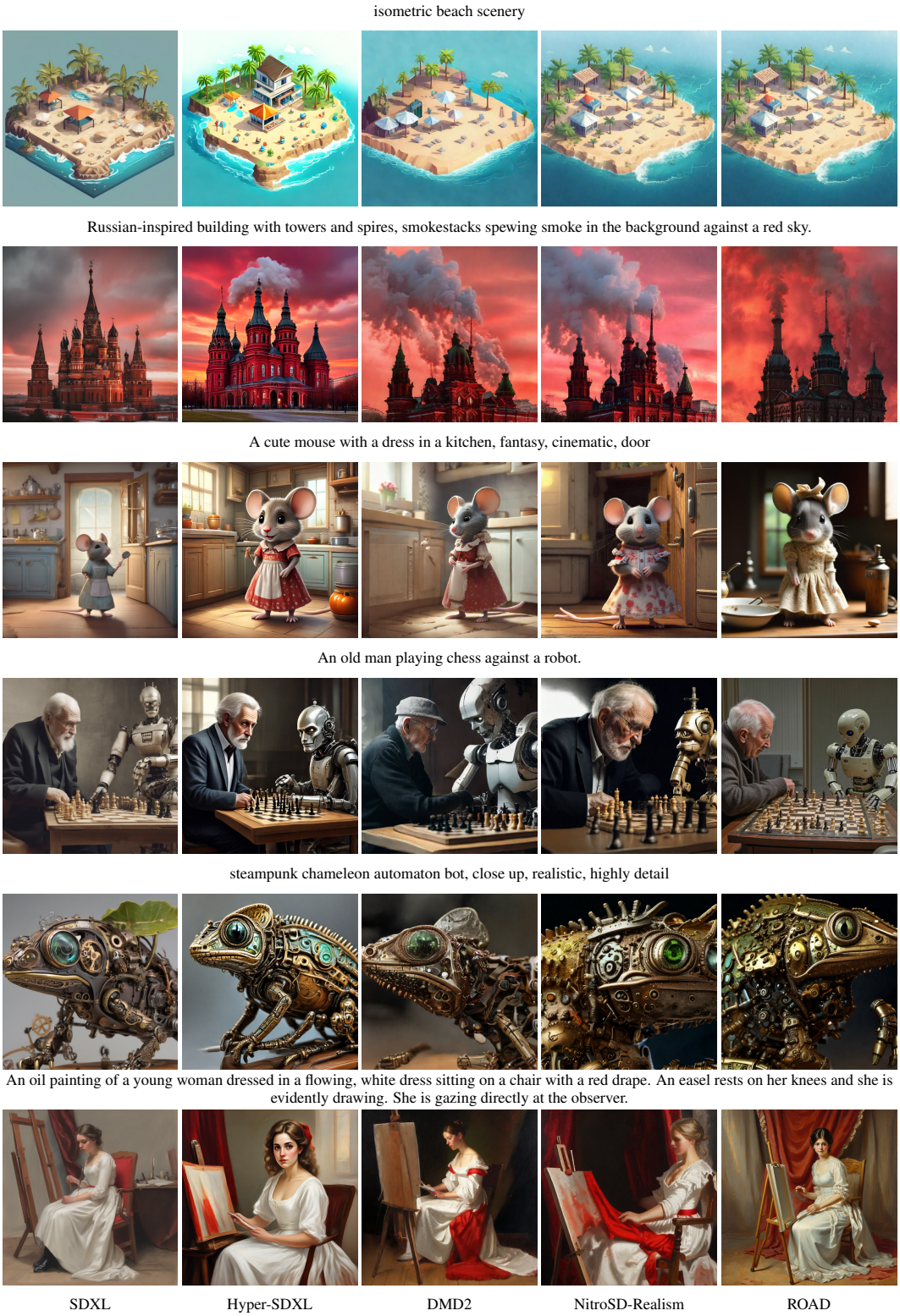


Figure 11: **Qualitative SDXL comparison under shared prompts.** The SDXL teacher uses 40 inference steps; all four distilled methods use one step.

an anthropomorphic owl courier, monochromatic with subtle red accents



A futuristic city with flying cars, tall glass towers covered in plants, and a large park in the center, bright daylight.



A bookshelf filled with colorful books, a potted plant and a small globe.



A photo of an cosplay girl dressed as a fantasy warrior, with a full suit of armor and a sword at her side, standing on a hilltop with a dramatic sunset behind her



bedroom melting into the galaxy, penthouse apartment bedroom with big windows, unsplash, photo of the day



A teddy bear reading a newspaper on a park bench.



SD3.5-M	SiD-DiT	SwD	SenseFlow	ROAD
40 steps / 80 NFE	4 steps / 4 NFE	4 steps / 4 NFE	4 steps / 4 NFE	1 step / 1 NFE

Figure 12: **Qualitative SD3.5-M comparison under shared prompts.** Inference steps and network evaluations (NFE) are given below the columns.