PIXEL SPACE DOESN'T SAVE YOU: TWO MECHANISMS OF CONDITIONING HOMOGENIZATION IN DIFFUSION TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

A key motivation for pixel-space Diffusion Transformers is that deleting the VAE removes a lossy representation bottleneck. We show this fails for the *conditioning* pathway: the bottleneck was never in the VAE. Across 12 checkpoints and 9 architecture families, pixel- and latent-space models interleave along the homogenization axis, with conditioning vectors degenerating toward near-identical representations across instances. We trace this to two mechanisms—a frequency-schedule mismatch in pooled conditioning, and text-stream writeability in token conditioning—and derive from the first a training-free, checkpoint-level diagnostic, the norm ratio $\|t_{\text{emb}}\|/\|y_{\text{emb}}\|$ ($r = 0.83$, $p < 10^{-3}$), validated out-of-sample on four held-out architectures under pre-registered predictions. The same ratio predicts *class-code fragility* across those checkpoints ($r = -0.888$), showing functional consequences in models we did not train. We validate the second mechanism by perturbing *only* the prompt-specific residual, which changes the named attribute in 92–100% of trials on SD3.5 and FLUX. We then propose *Conditioning Norm Balancing (CNB)*, a training-time regularizer that restores balanced conditioning geometry, and evaluate it under a shared-noise paired protocol (10,000 pairs per cell, McNemar's exact test, FID and accuracy from the *same* samples). *CNB* holds the ratio at parity and improves *both* metrics at each model's own optimal guidance, on *both* sides of the pixel/latent divide: $+2.70$ pp top-1 on SiT-B/2 (p=1.8×10^{-10}) and $+4.80$ pp on PixelDiT-B, which has no VAE (p=2.6×10^{-20}), with FID improving in both cases. At lower guidance, the two metrics give *opposite* verdicts on identical weights. A single FID number cannot certify a conditioning pathway.

1 INTRODUCTION

Diffusion transformers are mid-shift. A recent generation of models (Yu et al., 2026; Li & He, 2026; Wang et al., 2026a; Chen et al., 2025) removes the variational autoencoder and denoises directly in pixel space, on the premise that latent compression is a lossy bottleneck a sufficiently large transformer no longer needs. The accompanying framing is intuitive and unexamined: *fewer compression stages, fewer representation bottlenecks*.

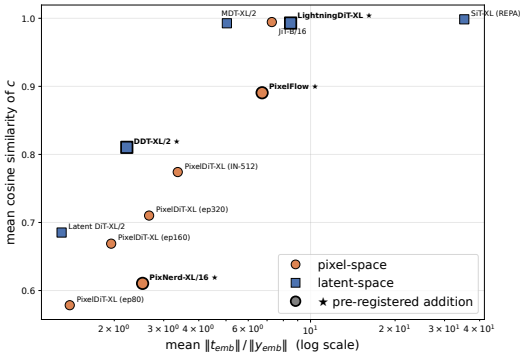

Figure 1: **Norm ratio predicts homogenization** of pixel and latent models. 12 checkpoints, 9 architecture families. *Circles*: pixel space; *Squares*: latent space; *Stars*: pre-registered held-out models.

This paper tests that premise on the one component nobody re-examined when the VAE was removed–the conditioning path–and finds it false. Conditioning is computed and combined *before* the network touches an image or a latent, so the pixel/latent decision is orthogonal to it (Fig. 1). JiT-B/16 (Li & He, 2026), a pixel-space model, homogenizes as completely as any latent model we test (mean cross-class cosine similarity 0.994, against SiT-XL's 0.998 (Ma et al., 2024b)). PixelDiT-T2I (Yu et al., 2026), also pixel-space, is the *most* token-homogenized of four text-

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.

conditioned models, above SD3.5 (Esser et al., 2024), FLUX.1 (Labs et al., 2025) and Qwen-Image (Wu et al., 2025), all of which compress through a VAE first. Along the axis that actually predicts homogenization, the two domains interleave rather than separate. Our contributions:

1. We uncover a *conditioning-norm imbalance in existing diffusion transformers*, showing that *pixel space does not escape conditioning homogenization*. A two-norm ratio predicts the degree of homogenization across 12 checkpoints and 9 model families ($r = 0.83$), including out-of-sample predictions (Sec. 4, Fig. 1).
2. We identify that *conditioning homogenization is governed by stream writeability, not depth*: it occurs only when the architecture writes to the conditioning stream, making cross-attention immune by construction. Perturbing *only* the prompt-specific residual then changes the named attribute in 92–100% of trials on SD3.5 and FLUX, showing the homogenized structure is semantically load-bearing (Sec. 6, 7).
3. We show that *homogenization has a functional cost that FID misses*. The norm ratio $\|t_{\text{emb}}\|/\|y_{\text{emb}}\|$ predicts class-code fragility ($r = -0.888$), while FID and class accuracy can strongly dissociate (Sec. 5).
4. We propose *Conditioning Norm Balancing (CNB) to correct this imbalance during training*. *CNB* restores balanced conditioning geometry and improves both FID and class accuracy across latent- and pixel-space DiTs, while identifying its specific failure regime (Sec. 5.3).

Why this matters now. The bottleneck survives VAE removal, arises from two inherited architectural choices, and *persists* throughout training while the norm ratio ($\|t_{\text{emb}}\|/\|y_{\text{emb}}\|$) and class-code robustness keep degrading—yet *it is diagnosable from a checkpoint*, before spending GPU-hours for training or sampling. We hope this inspires future works to treat conditioning geometry as a first-class design target, and to look for the same failure in conditioning pathways we did not examine.

2 Related Work

Pixel-space diffusion transformers. PixelDiT (Yu et al., 2026), JiT (Li & He, 2026), PixNerd (Wang et al., 2026a) and PixelFlow (Chen et al., 2025) remove the VAE and operate directly on pixels. The stated motivation is the removal of a lossy compression stage. Our results concern a component orthogonal to that choice.

Conditioning redundancy. Prior work (Pham et al., 2026) observes that conditioning signals in latent DiTs are highly redundant and exploits this for pruning. The natural reading is that redundancy is downstream of latent compression; we show it is not, and identify what actually predicts it. We differentiate our framing from concurrent within-instance channel-criticality analyses in Sec. N.

Representation quality in diffusion models. A productive line of work improves the *image*-stream representation, most prominently by aligning diffusion features with self-supervised encoders REPA (Yu et al., 2025a). Our findings are orthogonal and, we argue, complementary: image-stream representation work does not touch the conditioning stream, and the two can come apart sharply. Notably, the representation-alignment model SiT-XL (trained with REPA) has the *most fragile class conditioning* of any checkpoint we measure ($\epsilon_{50} = 0.30$, against PixNerd-XL's 5.6; Sec. 5).

Evaluating conditional generators. FID (Heusel et al., 2017) is the field default. Classifier-based evaluation is not new—Classification Accuracy Score (Ravuri & Vinyals, 2019) trains a classifier on synthetic data and evaluates on real, and "target class accuracy" appears in the guidance literature (Ho & Salimans, 2022) as a readout of guidance strength. We claim neither the metric nor the idea of classifier-based evaluation. What we contribute is its use as a *paired diagnostic tied to a mechanism*, and the demonstration that it *dissociates* from FID on identical weights.

3 Setup and Definitions

Pooled conditioning (Mechanism-A). Class-conditional DiT-style models form a single conditioning vector:

$$c = t_{\text{emb}} + y_{\text{emb}}, \tag{1}$$

optionally followed by a nonlinearity, which then modulates the network through adaLN. We measure *homogenization* as the mean pairwise cosine similarity of c across all $K = 1000$ classes at a fixed timestep, $\cos(c) = \mathbb{E}_{i \neq j}\left[\langle \hat{c}_i, \hat{c}_j \rangle\right]$, and the *norm ratio* as $\|t_{\text{emb}}\|/\|y_{\text{emb}}\|$.

Token-sequence conditioning (Mechanism-B). Text-conditioned MMDiT (Esser et al., 2024) carry a sequence `encoder_hidden_states`. We measure within-prompt similarity of these token representations as a function of block depth.

Class-code fragility. If the K class codes occupy a narrow cone, a small perturbation anywhere in the AdaLN pathway maps one code onto another. We perturb $c'_k = c_k + \epsilon \, \|c_k\| \, n/\sqrt{D}$, $n \sim \mathcal{N}(0, I)$, and report ϵ_{50}: the relative noise level at which nearest-neighbour class identity is recovered only half the time. Low ϵ_{50} means fragile class codes. *This is a property of trained conditioning geometry*, not of sample quality, and costs seconds per checkpoint with no sampling. Sec. C gives the exact computation, what ϵ_{50} is and is not sensitive to, and its relation to existing robustness measurements.

Paired evaluation protocol. Throughout Sec. 5, we compare two training arms that differ only in one loss term. Both arms use the *same* initial noise x_T and class labels for every sample, enabling within-pair comparison; significance is assessed with McNemar's exact test. We evaluate all 1000 ImageNet classes with 10 samples each, yielding 10,000 paired samples per cell. Each run *generates once and scores twice*: the same image tensors are used for both classifier evaluation and FID, eliminating differences in samples, guidance, or checkpoints as explanations for metric divergence.

We compute FID using the standard ADM/OpenAI (Dhariwal & Nichol, 2021) implementation and its standard ImageNet-256 reference statistics. We report FID-10k because each evaluation cell contains 10,000 generated samples, and we do not compare these values directly with published FID-50k results in our controlled settings.

4 Mechanism-A: Pooled Conditioning Is Set by a Frequency-Schedule Mismatch

The mismatch. Two pixel-space DiTs with near-identical architectures land at opposite ends of the spectrum (Tab. 2): JiT-B/16 sits at $\|t_{\text{emb}}\|/\|y_{\text{emb}}\| = 7.28$ with $\cos(c) = 0.994$, while PixelDiT-XL sits at 1.39 and 0.578. The difference is a single configuration choice. JiT inherits `max_period`=10000 from the discrete-DDPM (Ho et al., 2020; Huang et al., 2026) lineage while sampling continuous $t \in [0, 1]$; PixelDiT scales `max_period`=10 to its own sampling range.

The consequence is measurable on the embedding itself, before any training. For the standard construction with $\text{freqs}_i = \texttt{max_period}^{-i/(d/2)}$, a channel carries information only if $t \cdot \text{freqs}_i$ traverses an appreciable fraction of a period over the sampled range of t. Under the mismatch it does not: **69.5% of the 256 channels are effectively constant** across all of $t \in [0, 1]$ (peak-to-peak variation below 1% of the full swing), leaving 78 informative dimensions. PixelDiT's rescaled schedule leaves 218, and the DDPM schedule used with the t range it was designed for leaves 247 (Tab. 1). **The schedule is not defective; the pairing of schedule and sampling range is.**

Table 1: *The mismatch destroys most of the timestep signal before training begins.* Constant-channel criterion and interpretation are detailed in Sec. 4.

Configuration	`max_period`	t range	constant channels	informative dims
Mismatch (JiT, SiT lineage)	10000	$[0, 1]$	**69.5%**	**78 / 256**
Rescaled (PixelDiT)	10	$[0, 1]$	14.8%	218 / 256
Designed Schedule (DDPM)	10000	$[0, 999]$	3.5%	247 / 256

Two clarifications matter here. First, `max_period` (a frequency-scaling constant) and T (the number of diffusion steps) are unrelated quantities that happen to differ by $10\times$ in the original DDPM lineage; conflating them is what produces the mismatch in the first place. Second, channel collapse alone does *not* explain the label-swamping norm ratio—it removes signal, but the amplification is a learned property of the timestep MLP.

Holding JiT's trained `t_embedder` fixed and varying only the input frequency encoding leaves $\cos(c)$ between 0.9936 and 0.9956 across every variant we tried, including the corrected one (Tab. 9 of Sec. B.2). **The defect is baked into the weights and cannot be undone at inference**, which is what motivates a training-time fix (Sec. 5.3).

The diagnostic. Across 12 checkpoints of State-of-the-art diffusion transformer frameworks (Tab. 2) the norm ratio explains homogenization with $r = 0.83$ (Spearman $\rho = 0.88$, $p = 8.8 \times 10^{-4}$, bootstrap 95% CI $[0.73, 0.95]$); see Fig. 1 and Sec. B. The lowest-ratio model in the study is a *latent* one (DiT-XL/2, ratio 1.30), sitting below every pixel-space PixelDiT variant.

Table 2: *Mechanism-A across 12 independently trained checkpoints from 9 architecture families*, sorted by norm ratio. Pixel- and latent-space models *interleave*, with pixel-space models spanning both extremes. ⋆ marks four held-out architectures with pre-registered predictions. ϵ_{50} is the relative perturbation at which nearest-neighbour class identity survives half the time (*lower is more fragile*). The norm ratio strongly predicts ϵ_{50} (log–log $r = -0.888$; full statistics in Sec. 5.2). The upper block contains public model checkpoints; controlled CNB pairs below are excluded from the correlation.

Model	Domain	$\|t_{\text{emb}}\|/\|y_{\text{emb}}\|$	$\cos(c)$	ϵ_{50}	Regime
Latent DiT-XL/2 (Peebles & Xie, 2023)	latent	1.30	0.685	4.84	label-informative
PixelDiT-XL (ep80) (Yu et al., 2026)	pixel	1.39	0.578	5.57	label-informative
PixelDiT-XL (ep160) (Yu et al., 2026)	pixel	1.95	0.669	5.18	label-informative
DDT-XL/2⋆ (Wang et al., 2026b)	latent	2.22	0.810	4.23	intermediate
PixNerd-XL/16⋆ (Wang et al., 2026a)	pixel	2.52	0.611	**5.64**	label-informative
PixelDiT-XL (ep320) (Yu et al., 2026)	pixel	2.66	0.710	4.74	intermediate
PixelDiT-XL (IN-512) (Yu et al., 2026)	pixel	3.37	0.774	3.97	intermediate
MDT-XL/2 (Gao et al., 2023)	latent	5.04	0.993	0.68	homogenized
PixelFlow⋆ (Chen et al., 2025)	pixel	6.72	0.891	2.45	intermediate
JiT-B/16 (Li & He, 2026)	pixel	7.28	0.994	0.58	homogenized
LightningDiT-XL⋆ (Yao et al., 2025)	latent	8.49	0.993	0.62	homogenized
SiT-XL (REPA) (Yu et al., 2025b)	latent	35.26	0.998	**0.30**	homogenized
Controlled pairs (ours); CNB is the only difference within each pair:					
SiT-B/2 @100k, baseline	latent	13.95	0.922	1.49	homogenized
+ CNB (ours)	latent	**1.00**	**0.530**	**5.74 (3.86×)**	label-informative
SiT-B/2 @400k, baseline	latent	28.18	0.944	1.12	homogenized
+ CNB (ours)	latent	**1.00**	**0.522**	**5.78 (5.17×)**	label-informative
PixelDiT-B @100k, baseline	pixel	4.60	0.954	1.71	homogenized
+ CNB (ours)	pixel	**1.02**	**0.583**	**5.50 (3.22×)**	label-informative
PixelDiT-B @400k, baseline	pixel	4.33	0.961	1.57	homogenized
+ CNB (ours)	pixel	**1.02**	**0.717**	**4.48 (2.85×)**	intermediate

Out-of-sample validation. To guard against post-hoc fitting we registered predictions in writing for four held-out 2025-era architectures (LightningDiT-XL, DDT-XL/2, PixNerd-XL/16, PixelFlow) *before* measuring them; they are reproduced verbatim in Sec. B.1. Three of four were correct. The miss is informative, and we report it: PixelFlow has a correctly matched schedule yet develops a large ratio through learned MLP gain. The consequence is a sharpening of the claim rather than a weakening of it: the schedule mismatch is *a* cause of a high ratio, not the only one, and **the diagnostic must be computed from the checkpoint rather than read off a config**. Adding the four held-out models left the coefficient essentially unchanged ($0.846 \to 0.828$) while cutting the p-value roughly ninefold and tightening the CI.

It persists across training; it does not wash out. A natural objection is that homogenization is an undertraining artifact that more compute would dissolve. It is not. On controlled pairs trained to 400k steps the baseline $\cos(c)$ at $t{=}0$ is *already* 0.999 at 100k and is 1.000 at 400k—saturated from roughly step 1000 onward (Fig. 2). Averaged over five timesteps, the baselines sit at $0.92 \to 0.94$ (SiT) and $0.95 \to 0.96$ (PixelDiT) across 100k–400k: high and flat (the full curve comparison is in Sec. L).

Because the cosine is at the ceiling, it cannot register further degradation, so the sharper evidence is in the underlying quantities, which keep worsening—SiT's norm ratio grows $13.95 \to 28.18$ and its class-code robustness *falls*, $\epsilon_{50} = 1.49 \to 1.12$ (Tab. 2). Additional training makes the conditioning geometry worse, not better.

5 What Homogenization Costs, and Why FID Cannot See It

5.1 The Measurement Gap

This section concerns *Mechanism-A* throughout: the object is the pooled class code c, and the models are class-conditional. Sec. 6 takes up the second mechanism once the account of the first is complete. FID compares generated and real image *distributions*. It never asks whether a sample conditioned on class k depicts class k; a perfectly distributed but class-scrambled ImageNet scores well. Dense-correspondence probes measure *image*-stream feature quality and do not touch the conditioning pathway at all. Neither tests **class controllability**–whether the conditioning signal supplied is the one that comes out–which is precisely what a homogenized c should damage.

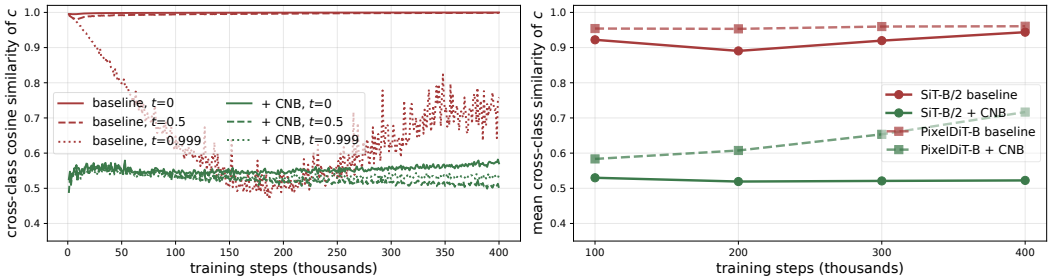

Figure 2: **Baseline homogenization persists across training; CNB holds it off.** Left: SiT training-time logs, ∼400 points per arm over 400k steps. The baseline is saturated at t=0 and t=0.5 from roughly step 1000 and never recovers; the noisier t=0.999 trace accounts for most of the movement in any average. Right: measured directly from the checkpoints of both architectures, so the claim does not rest on training-time instrumentation. CNB holds flat on SiT ($0.53 \rightarrow 0.52$) but partially erodes on PixelDiT ($0.58 \rightarrow 0.72$), for the reason analysed in Sec. J.

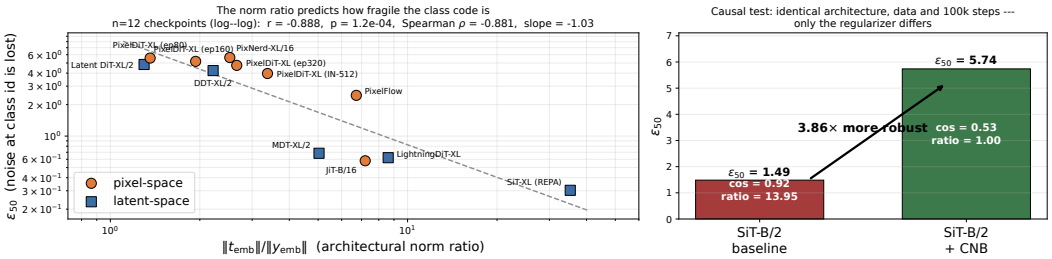

Figure 3: **The conditioning geometry predicts class-code fragility out-of-sample.** Left: the architectural norm ratio against ϵ_{50} over the 12 checkpoints of Tab. 2, pixel- and latent-space interleaved. Right: the controlled pair, where CNB is the only difference. Fit statistics and effect size are printed in the panels.

5.2 The geometry predicts fragility, out-of-sample

Across all 12 Mechanism-A checkpoints (Tab. 2, Fig. 3), the architectural norm ratio predicts class-code fragility with $r = -0.888$ ($p = 1.2\times10^{-4}$, Spearman $\rho = -0.881$, fitted slope ≈ -1.0, i.e. ϵ_{50} roughly inversely proportional to the ratio). The spread is nearly $20\times$: PixNerd-XL/16 at $\epsilon_{50} \approx 5.6$ versus SiT-XL (REPA) at $\epsilon_{50} = 0.30$, whose class codes are destroyed by perturbations a third the magnitude of the code itself. **None of these checkpoints was trained by us or saw our intervention**, making this the strongest single piece of evidence in the paper: a two-norm diagnostic predicts a functional property out-of-sample. Visualized ϵ_{50}-curves is shown in Appendix Fig. 10.

CNB restores robust class codes on both architectures. Tab. 2 shows the intervention moving ϵ_{50} by 2.9–5.2$\times$ in every controlled pair. Two details are worth stating. First, the baseline geometry *degrades* with training on both architectures—SiT's ratio rises $13.95 \rightarrow 28.18$ and its ϵ_{50} falls $1.49 \rightarrow 1.12$; on PixelDiT the ratio is instead flat ($4.60 \rightarrow 4.33$, Fig. 15) and the degradation shows up in $\cos(c)$ ($0.954 \rightarrow 0.961$) and ϵ_{50} ($1.71 \rightarrow 1.57$)—so the defect compounds rather than anneals. Second, CNB's own recovery is not uniform: on SiT it holds ($5.74 \rightarrow 5.78$, a $5.17\times$ improvement at 400k) while on PixelDiT it partially erodes ($5.50 \rightarrow 4.48$), for the reason analysed in Sec. J. Notably, PixelDiT reaches the *lower* ϵ_{50} of the two at 400k yet delivers the *larger* accuracy gain (+4.80 vs. +2.70 pp), so we caution explicitly against reading effect size off the conditioning geometry.

5.3 Conditioning Norm Balancing (CNB): a Training-Time Fix

Sec. 4 identifies $\|t_{\text{emb}}\|/\|y_{\text{emb}}\|$ as the quantity that predicts homogenization, and Sec. 5 shows that quantity predicts a functional property out-of-sample. If the mechanism is causal, penalizing the ratio during training should reduce homogenization and recover class controllability. *We therefore propose a single term to the diffusion objective:*

$$\mathcal{L}_{\text{CNB}} = \mathbb{E}_{t,y}\left[\frac{\left(\|t_{\text{emb}}\| - \|y_{\text{emb}}\|\right)^2}{\|y_{\text{emb}}\|^2 + \varepsilon}\right], \quad \mathcal{L} = \mathcal{L}_{\text{diff}} + \lambda\,\mathcal{L}_{\text{CNB}}. \tag{2}$$

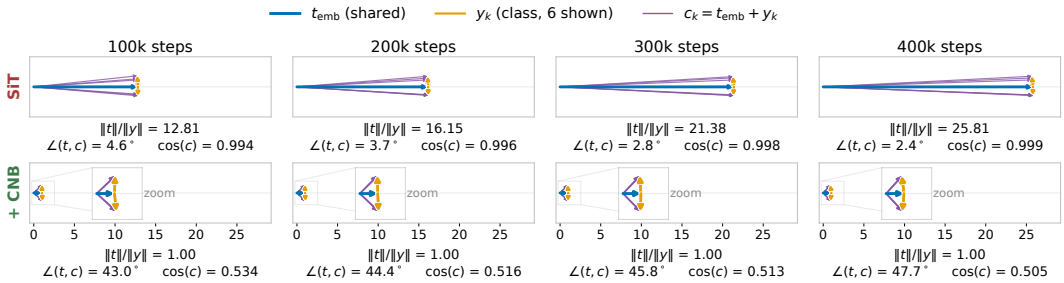

Figure 4: **Conditioning Vectors Evolution.** SiT adds, $c_k = t_{\text{emb}} + y_k$, so each class's three vectors share one plane. Lengths are in units of $\|y_{\text{emb}}\|$ on both rows; the CNB inset is magnified. Arrow lengths and every $\angle(t_{\text{emb}}, c_k)$ are exact, but the side each class is drawn on is a legibility choice, so inter-class angles are given as $\cos(c)$. Every quantity here is at t=0.5; Tab. 2 averages five timesteps, and the ratio falls steeply with t, so its value is higher (28.18 against 25.81 for SiT at 400k).

The denominator makes the penalty *relative*, so one λ transfers across architectures whose embedding norms differ by an order of magnitude; we use $\lambda = 0.5$ everywhere without tuning. CNB adds two norm computations per step, requires no extra parameters, no auxiliary network, and no change to sampling. CNB does what it targets, in the most direct sense. Fig. 4 draws the conditioning vectors themselves: the penalized quantity is the length of t_{emb} relative to the class term, and holding it at parity is what stops the class code from being swamped. The angular consequence is immediate—at 400k and t=0.5 the baseline's entire class identity rotates c_k by 2.4°, against 47.7° under CNB—and the two arms never come close to each other at any checkpoint (per-timestep norms in Fig. 15).

Fig. 5 shows the resulting geometry at 400k steps: the distribution of all $\binom{1000}{2} = 499{,}500$ pairwise cosine similarities between class codes separates completely between the two arms—the SiT baseline is a near-delta spike at 0.999 while CNB sits at a broad mode around 0.505, with no overlap—and the class codes span substantially more directions (effective dimensionality 85.7 → 132.9 on SiT, 97.3 → 215.0 on PixelDiT). The change is also immediate and stable: cosine similarity drops within the first 1–2k steps and holds for the rest of training (Fig. 2). We train matched pairs differing *only* in Eq. 2 on a latent model (SiT-B/2, 130.5M) and a pixel-space model with no VAE (PixelDiT-B, 184.7M), both to 400k steps on ImageNet-256, and evaluate under the paired protocol of Sec. 3. **The same curve shape on both sides of the divide.** Both architectures are null at low guidance, significant from cfg ≈ 3, and show an FID cost that declines monotonically and crosses into a benefit (SiT at cfg ≈ 3.5, PixelDiT at ≈ 5) (Tab. 3 and Fig. 6.)

Table 3: **Optimal guidance sweeps.** Δ *is our regularizer minus baseline*; positive ΔFID means the regularizer is *worse* and vice versa; positive Δtop-1 means better. † marks each baseline's own FID optimum. **Bold** means **Ours** is better.

	SiT (latent) (Ma et al., 2024a)			PixelDiT (pixel) (Yu et al., 2026)		
CFG	Δtop-1	p	ΔFID	Δtop-1	p	ΔFID
1.0	−1.04	1.7×10^{-3}	+6.34	−0.03	0.92	+3.33
2.0	**+0.01**	1.00	+2.44	**+0.15**	0.68	+3.50
3.0	**+2.30**	1.5×10^{-7}	+0.26	**+1.38**	1.1×10^{-3}	+2.10
4.0†	**+2.70**	1.8×10^{-10}	**−0.70**	**+1.66**	4.5×10^{-4}	+1.01
6.0	**+2.57**	3.1×10^{-10}	**−1.52**	**+3.21**	3.3×10^{-10}	**−0.71**
8.0	**+3.32**	1.1×10^{-16}	**−2.04**	**+3.89**	3.2×10^{-14}	**−1.38**
10	**+2.90**	2.5×10^{-12}	**−2.25**	**+4.20**	4.7×10^{-16}	**−1.77**
12	**+2.25**	1.5×10^{-7}	**−2.51**	**+4.74**	5.5×10^{-20}	**−1.96**
16†	+0.00	1.00	**−2.08**	**+4.80**	2.6×10^{-20}	**−1.78**

FID and class accuracy dissociate. At SiT cfg=2 the regularizer is worse by FID (+5.95) and better by accuracy (+1.19 pp, $p = 8.2 \times 10^{-4}$); at cfg=8 it is better by both. PixelDiT reproduces the dissociation independently at cfg=3 and cfg=4. **Verdicts reverse with a sampling hyperparameter on identical weights**, so a single FID number at a single guidance setting cannot certify a conditioning pathway. This claim does not depend on which operating point one prefers. **At each baseline's own optimum, CNB wins on both metrics.** We deliberately pin the comparison to the *baseline's* FID optimum—chosen without reference to our method—rather than letting each arm use its own best, which would amount to selecting the operating point in our favour. Under that conservative choice, CNB still

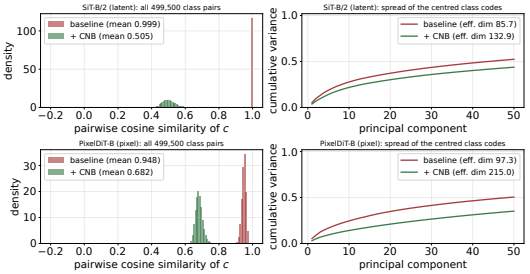

Figure 5: **Effect of CNB on conditioning geometry** (400k checkpoints, t=0.5). **Left:** pairwise cosine similarity of the class codes; PixelDiT shifts 0.948 → 0.682. **Right:** cumulative variance of the centred codes. Construction in Sec. K.

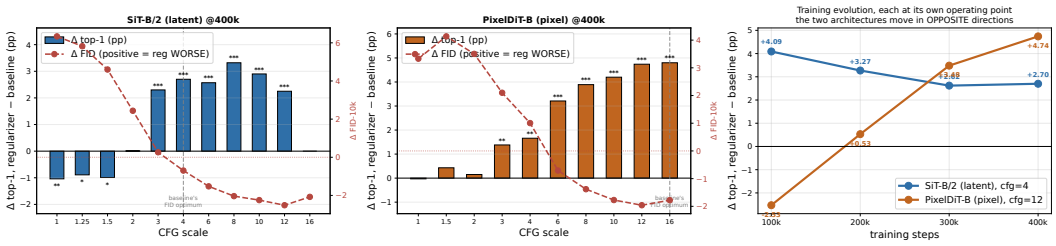

Figure 6: **The intervention on both sides of the pixel/latent divide.** Left/middle: guidance sweeps with each baseline's FID optimum marked. Right: training evolution at each model's operating point—*the two architectures move in opposite directions*, which we report without an explanation.

improves distributional quality *and* class fidelity simultaneously, on both a latent and a pixel-space DiT (Tab. 4). To our knowledge, *this work is the first demonstration that a conditioning-geometry intervention transfers across the pixel/latent divide.*

Table 4: **CNB (FID and Top-1) at the baseline's own optimal guidance**, 400k steps, 10,000 paired samples per cell. Both metrics improve on both architectures. **Bold** means **Ours** is better.

Model	Operating point	Baseline	+ **CNB Regularizer (ours)**
SiT-B/2 (latent)	cfg=4	20.10 FID / 74.68%	**19.41 / 77.38%** (+2.70 pp, p=1.8×10^{-10})
PixelDiT-B (pixel)	cfg≈12–16	28.39 FID / 58.57%	**26.61 / 63.37%** (+4.80 pp, p=2.6×10^{-20})

Qualitative evidence. Fig. 22 shows paired cases where the two arms disagree. The failures are *near-misses between neighbouring classes*—streetcar→"trolleybus", lion→"cougar", Pembroke→"Cardigan"—so each baseline image is a plausible photograph that is simply not the requested class. This is exactly the failure a distribution-level metric cannot register. We show both directions of disagreement, and Sec. M contains the failure cases. More qualitative results are in the Appendix 8.

5.4 Robustness

Sampler budget: the effect is flat across a 20× range. Re-running the entire paired protocol at 10/50/100/200 Euler steps—16 cells, 320,000 paired generations—leaves the effect essentially unchanged (Tab. 6, right; full grid with ΔFID in Tab. 13). At cfg=2 it spans 1.16–1.26 pp and at cfg=4 4.05–4.36 pp, significant at every setting. This also rules out a competing explanation: if CNB merely conferred robustness to *coarse* sampling, the gap would shrink as steps increase. It does not; if anything, it drifts marginally upward. Baseline accuracy itself saturates by ∼50 steps, so the additional budget probes the same operating point rather than a different one.

Training duration: the trend is architecture-specific. Tab. 6 (left) gives Δtop-1 at each model's operating point across four checkpoints of the same two runs. On SiT, the benefit is largest early and settles to a plateau; on PixelDiT, it begins significantly *negative* and climbs monotonically. We have no account of the sign difference and draw no general conclusion about how the intervention scales with training (Fig. 6, right panel).

Table 5: **FID seed variance**, SiT-B/2, three independent noise seeds per cell. Δ is CNB minus baseline; negative ΔFID favours CNB. All six seed-level tests favour CNB.

Setting	FID baseline	FID + CNB	ΔFID	Δtop-1 (pp)
cfg=6	41.35 ± 0.33	$\mathbf{40.20 \pm 0.22}$	$\mathbf{-1.15} \pm 0.39$	$\mathbf{+5.50 \pm 0.14}$
cfg=8	41.58 ± 0.07	$\mathbf{39.43 \pm 0.11}$	$\mathbf{-2.15} \pm 0.17$	$\mathbf{+5.63 \pm 0.43}$

FID error bars. Every accuracy claim above carries a p-value while FID was a single measurement, so we replicated the two settings that carry the headline across three independent noise seeds (10,000 paired samples each). Within a seed both arms share x_T, making ΔFID a paired quantity with a tighter spread than the raw FIDs.

Tab. 5 gives both settings. At cfg=8 the effect is ∼13× its own standard deviation, and **3/3 seeds favour CNB at both settings**. Accuracy is markedly more stable than FID (sd 0.14–0.43 pp), which is why we lead with accuracy and treat FID as corroboration. With $n = 3$ we report mean ± sd and sign-consistency and deliberately do *not* quote a t-test, since three points estimate a standard deviation too poorly for the resulting p to mean what it appears to.

5.5 Practical recommendations

Three concrete takeaways follow from our measurements, none of which requires adopting CNB.
(i) Audit the ratio before training. Loading a checkpoint and comparing two embedding norms

Table 6: **Robustness of the accuracy gain. Left**: Δtop-1 across training, at each model's own operating point. **Right**: Δtop-1 across sampler budgets (SiT). Both panels are read in Sec. 5.4.

Training Steps	SiT (cfg 4)	PixelDiT (cfg 12)
100k	**+4.09**	−2.53
200k	**+3.27**	**+0.53**
300k	**+2.62**	**+3.48**
400k	**+2.70**	**+4.74**

Euler steps	cfg 1	cfg 1.5	cfg 2	cfg 4
10	**+0.04**	**+0.27**	**+1.16**	**+4.05**
50	−0.13	**+0.14**	**+1.19**	**+4.09**
100	**+0.02**	**+0.52**	**+1.26**	**+4.36**
200	**+0.10**	**+0.57**	**+1.17**	**+4.32**

Table 7: **Mechanism-B is decided by stream writeability, not depth.** Depth and homogenization are uncorrelated: the deepest model is the least homogenized, and the shallowest joint-stream model is the most. Cross-attention architectures never write back to `encoder_hidden_states` and are therefore immune by construction—verified from source, requiring no experiment.

Model	Text-stream access	Blocks	Depth profile
Qwen-Image (Wu et al., 2025)	writable (dual)	60	lowest of all tested, despite max depth
SD3.5 (Esser et al., 2024)	writable (dual)	24	early peak, then mild decline
FLUX.1 (Labs et al., 2025)	writable (dual + single)	19 + 38	peaks, then declines at the single-stream boundary
PixelDiT-T2I (Yu et al., 2026)	writable (dual)	14	*highest* plateau of all four (0.845)
SANA1.5 (Xie et al., 2025)	read-only (cross-attn)	—	immune by construction
PixArt-Σ (Chen et al., 2024b)	read-only (cross-attn)	—	immune by construction
Wan2.1-T2V (Wan et al., 2025)	read-only (cross-attn)	—	immune by construction (video)

takes seconds and requires no sampling; a ratio above ∼5 has corresponded to $\cos(c) > 0.99$ in every checkpoint we measured. Because a matched schedule can still develop a large ratio through learned MLP gain (Sec. 4), this must be computed from weights rather than read off a config. **(ii) Check `max_period` against the t-sampling distribution.** Inheriting `max_period`=10000 from the discrete-DDPM lineage while sampling continuous $t \in [0, 1]$ freezes most sinusoidal channels into near-constants. This is a one-line change at initialization and costs nothing. **(iii) Report a conditioning-fidelity number alongside FID.** Top-1 agreement of a pretrained classifier with the requested class costs one forward pass per sample on top of generation you are already doing. Our results show the two metrics can move in opposite directions on identical weights, so reporting only one leaves a failure mode invisible.

6 Mechanism-B: Token Conditioning Is Decided by Stream Writeability

Everything to this point concerns *Mechanism-A*: a single pooled vector $c = t_{\text{emb}} + y_{\text{emb}}$. Text-conditioned MMDiTs instead carry a *sequence* of text tokens through every block, so homogenization means those tokens converging toward one another—a different object with a different cause, to which neither the norm ratio nor CNB applies.

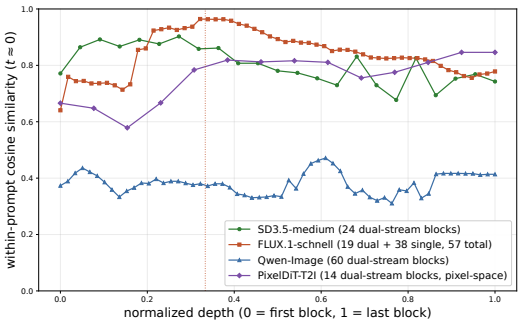

Figure 7: **Within-prompt token similarity vs. normalized depth.** The dotted line marks FLUX's dual→single-stream boundary, where similarity begins to decline. Qwen-Image remains lowest despite being by far the deepest, while the pixel-space model reaches the highest plateau (Tab. 7).

Across seven architectures (Tab. 7, depth profiles in Fig. 7) token-sequence homogenization is real but **not monotonic in depth**. Qwen-Image (Wu et al., 2025), the deepest model tested (60 dual-stream blocks), is the *least* homogenized; PixelDiT-T2I (Yu et al., 2026), with 14, is the most (final within-prompt similarity 0.845). FLUX (Labs et al., 2025) peaks during its dual-stream blocks and *declines* through its single-stream blocks—the dotted marker in Fig. 7—where diverse image tokens are mixed into the text embedding; rerunning with shared image noise rules out a noise-injection confound (Sec. D). What decides whether the mechanism can occur at all is architectural and verifiable without experiments (lower block of Tab. 7): *cross-attention architectures never update* `encoder_hidden_states`. SANA1.5 (Xie et al., 2025), PixArt-Σ (Chen et al., 2024a) and Wan2.1-T2V (Wan et al., 2025) therefore give a zero-homogenization boundary condition by construction. Wan2.1-T2V addition-



Figure 8: **Top**: paired class-conditional samples at 400k sharing x_T and the label; class predictions are shown below. Pairs are selected from 24 searches per class, not by eye. **Bottom**: residual-only interventions on latent- and pixel-space text-to-image models (Tabs. 4 and 11).

ally keeps timestep conditioning on a separate path. Our account of Qwen-Image's resistance is incomplete—per-block gate magnitudes correlate only weakly with similarity ($r = -0.40$)—and we report it as an anomaly, not an explanation (Sec. E).

7 Validating Mechanism-B: The Homogenized Structure Is Semantically Load-Bearing

A high cosine similarity in high dimensions may be a degenerate statistic rather than evidence about representations. The two mechanisms need separate answers: this section validates Mechanism-B, as Sec. 5 has already done for Mechanism-A, and neither transfers to the other.

The prediction. Given the convergence of Sec. 6, each prompt's `encoder_hidden_states` at a mid-network block should split into a *shared* component carrying little prompt-specific information and a small *residual* carrying the rest. This is falsifiable, not a restatement of the cosine: were the similarity degenerate, the residual would be noise, and perturbing it would yield nothing or incoherent output. We take the shared component as the mean over 20 unrelated reference prompts and manipulate *only* the residual, training-free (Sec. D.1).

It holds. 14 targets $\times$ 3 attributes $\times$ 2 models, CLIP-scored: **all nine model$\times$attribute distributions are significant** under both a one-sample t-test and a Wilcoxon signed-rank test ($p < 0.05$, most $p < 0.001$; Cohen's d 0.63–5.15), at 92–100% success on SD3.5 and FLUX (Tab. 11). Fig. 8 shows both mechanisms with clear qualitative comparisons; more panels in Fig. 11 and 12.

And effect size tracks the architecture rule. Sec. 6 identifies stream writeability as what governs token homogenization, and the reliability map follows the same variable: SD3.5, least diluted, shows the largest margins; FLUX is damped $\sim$2.6$\times$, its intervention block followed by 38 blocks that mix text with image tokens; and PixelDiT-T2I, the most token-homogenized, *reorders* the map rather than scaling it (style 85.7% against 64.3%). A generic editor would not track architecture this way.

Scope. This is a validation instrument, not a proposed editing method. We report its failures: two cheap predictors of when it works both fail, and scene-level video attributes fail at two structurally different injection points. A colour-transfer result we previously reported for video does not survive scrutiny—CLIP calls it a success, a direct hue measurement disagrees—and we retract it (Sec. F, G).

8 Conclusion

Removing the VAE does not remove the representation bottleneck: it persists in the *conditioning pathway*. We identify two mechanisms: in pooled conditioning, a frequency-schedule mismatch lets timestep embeddings overwhelm class embeddings, with the resulting norm ratio predicting class-code fragility *out-of-sample* across 12 models ($r = -0.888$); in token conditioning, stream writeability rather than depth determines homogenization, leaving cross-attention immune by construction, and perturbing *only* the prompt-specific residual changes the named attribute in 92–100% of trials. Conditioning Norm Balancing (CNB) corrects the pooled imbalance, holds the ratio at parity, and improves *both* FID and top-1 accuracy at each baseline's FID optimum on latent- and pixel-space DiTs (+2.70 and +4.80 pp). Conditioning geometry should be diagnosed before training and fidelity reported alongside FID; Sec. A gives the limitations and the open extension to token conditioning.

AI USE STATEMENT

In this work, we used generative AI tools for language polishing, grammar refinement, and sentence rephrasing of the manuscript. We have not used generative AI tools for conceptualizing research ideas, deriving theoretical proofs, designing experimental methodologies, analyzing raw data, or drafting initial prose, and secondary tasks involving automated data collection or synthetic data generation are not applicable to this work. Additionally, we used generative AI tools for writing boilerplate plotting scripts and formatting LaTeX tables. We have reviewed all AI-assisted work. Specifically, LLM-suggested code snippets for visualization were independently verified and tested for correctness by two authors, and all polished prose was carefully reviewed to ensure technical accuracy and faithfulness to our original research findings. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work focuses on foundational analysis and diagnostic tools for generative Diffusion Transformers (DiTs). While generative models can potentially be used for generating misleading or synthetic media, our research aims to expose and correct internal representation failures–specifically conditioning homogenization–thereby improving model transparency, controllability, and robustness against unexpected generative artifacts. All experiments were conducted using standard, publicly available datasets and open-source model checkpoints. No human subjects were involved, and no private or personally identifiable information was utilized. We do not foresee any immediate negative ethical impacts or direct harms arising from this theoretical and algorithmic investigation.

REPRODUCIBILITY STATEMENT

All measurements derive from public checkpoints or from controlled pairs we train with a single differing loss term. Sec. H gives the full evaluation protocol, Sec. B the per-checkpoint measurements, and Sec. I the training configurations. Every figure in this paper is produced by a script released with the code.

REFERENCES

Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. In *European Conference on Computer Vision*, pp. 74–91. Springer, 2024a. 8

Junsong Chen, Jincheng Yu, Chongjian Ge, et al. PixArt-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *International Conference on Learning Representations (ICLR)*, 2024b. 8

Shoufa Chen, Chongjian Ge, Shilong Zhang, Peize Sun, and Ping Luo. Pixelflow: Pixel-space generative models with flow. *arXiv preprint arXiv:2504.07963*, 2025. 1, 2, 4, 13, 16

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 3

Patrick Esser, Sumith Kulal, Andreas Blattmann, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning (ICML)*, 2024. 2, 3, 8, 20

Shanghua Gao, Pan Zhou, Ming-Ming Cheng, and Shuicheng Yan. Masked diffusion transformer is a strong image synthesizer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 23164–23173, 2023. 4, 13, 16

Robert Geirhos, Carlos RM Temme, Jonas Rauber, Heiko H Schütt, Matthias Bethge, and Felix A Wichmann. Generalisation in humans and deep neural networks. *Advances in neural information processing systems*, 31, 2018. 18

Andrew Gelman and John Carlin. Beyond power calculations: Assessing type s (sign) and type m (magnitude) errors. *Perspectives on Psychological Science*, 9(6):641–651, 2014. 25

Justin Gilmer, Nicolas Ford, Nicholas Carlini, and Ekin Cubuk. Adversarial examples are a natural consequence of test error in noise. In *ICML*, 2019. 18

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HJz6tiCqYm. 18

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 2

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 2

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3

An Huang, Junggab Son, and Zuobin Xiong. Watch your step: Information injection in diffusion models via shadow timestep embedding. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=agDGdEJ983. 3

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025. 2, 8, 20

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36115–36125, 2026. 1, 2, 4, 13, 16

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pp. 23–40. Springer, 2024a. 6

Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. SiT: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision (ECCV)*, 2024b. 1

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023. 4, 13, 15, 16

Trung X. Pham, Kang Zhang, Ji Woo Hong, and Chang D. Yoo. A hidden semantic bottleneck in conditional embeddings of diffusion transformers. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=FetaeuGsEs. 2

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=tL89RnzIiCd. 18

Suman Ravuri and Oriol Vinyals. Classification accuracy score for conditional generative models. *Advances in neural information processing systems*, 32, 2019. 2, 18

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 18

Evelyn Turri, Davide Bucciarelli, Sara Sarto, Lorenzo Baraldi, and Marcella Cornia. Few channels draw the whole picture: Revealing massive activations in diffusion transformers. *arXiv preprint arXiv:2605.13974*, 2026. 29

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 8

Shuai Wang, Ziteng Gao, Chenhui Zhu, Weilin Huang, and Limin Wang. Pixnerd: Pixel neural field diffusion. In *International Conference on Learning Representations*, volume 2026, pp. 43559–43580, 2026a. 1, 2, 4, 13, 16

Shuai Wang, Zhi Tian, Weilin Huang, and Limin Wang. Ddt: Decoupled diffusion transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 40633–40642, 2026b. 4, 13

Felix A Wichmann and N Jeremy Hill. The psychometric function: I. fitting, sampling, and goodness of fit. *Perception & psychophysics*, 63(8):1293–1313, 2001. 18

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025. 2, 8

Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. SANA: Efficient high-resolution text-to-image synthesis with linear diffusion transformers. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=N8Oj1XhtYZ. 8

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15703–15712, 2025. 4, 13, 16

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *International Conference on Learning Representations (ICLR)*, 2025a. 2

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=DJSZGGZYVi. 4, 13, 16

Yongsheng Yu, Wei Xiong, Weili Nie, Yichen Sheng, Shiqiu Liu, and Jiebo Luo. Pixeldit: Pixel diffusion transformers for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14273–14282, 2026. 1, 2, 4, 6, 8, 13, 16, 20

APPENDIX

A DISCUSSION AND LIMITATIONS

What we claim, and what we do not. We claim that the two metrics dissociate, that the conditioning geometry predicts a functional property out-of-sample, and that an intervention targeting that geometry improves both metrics at each model's own operating point on two architectures. We do *not* claim a competitive quality result: these are B-scale models and our FID numbers are not comparable to published FID-50k figures for XL-scale systems. The intervention is evidence that the mechanism is causal, not a proposed state-of-the-art method. For Mechanism B we claim less: that stream writeability decides whether token homogenization can occur, and that the homogenized structure is semantically load-bearing under direct intervention. **We offer no fix for Mechanism B**—CNB is defined on a pooled vector and does not apply to a token sequence—and the paragraphs below concern Mechanism-A.

Where the intervention fails. At cfg≤ 1.5 on SiT the regularizer is *worse* on both metrics (-1.04 pp, $p = 1.7\times10^{-3}$). This matters practically, because well-trained XL-scale models typically optimize

FID near cfg $\approx$ 1.5, whereas our B-scale models optimize at cfg=4–16; the regime where the intervention hurts may be the relevant one at larger scale. This is untested, and we flag it as the principal threat to practical significance.

Training dependence runs in opposite directions. On SiT the benefit is largest early and settles to a plateau ($+4.09 \to +2.70$ pp over 100k–400k); on PixelDiT it starts *significantly harmful* (-2.53 pp at 100k) and climbs monotonically to $+4.74$ pp. We have no mechanistic account of the sign difference. Consequently, **any claim about how this intervention scales with training must be made per-architecture**, and neither trajectory should be extrapolated.

The ratio is one route, not the only one. On PixelDiT the regularizer holds the norm ratio at ≈ 1.03 as designed, yet $\cos(c)$ still drifts $0.63 \to 0.82$ over training, because the raw label table itself collapses ($\cos(y, y)$: $0.072 \to 0.644$). This is a side effect of the objective rather than a quirk of one model: CNB raises $\cos(y, y)$ on *both* architectures, and the cause is exact—label-table similarity equals the squared fraction of each embedding carried by a component shared across classes, and the cheapest way to satisfy a penalty on $\|y_{\text{emb}}\|$ is to add one shared offset to all 1000 embeddings (Sec. J). The norm ratio is thus one route to homogenization and CNB opens another, which a centred formulation of Eq. 2 should close. Notably this does not prevent CNB from working—at 400k PixelDiT has the worse conditioning geometry of the two arms yet the larger accuracy gain—so effect size is not a function of the cosine and we make no such claim.

Negative and corrected results. We report a refuted pre-registration (we predicted the benefit would be largest at low guidance; it is the opposite, monotonically, across nine settings), a Type-M error caught by scaling up (N=1000 overestimated the cfg=2 effect roughly fourfold), and a methodological caution: our early FID numbers came from a 100-sample training-loop monitor that inflated absolute FID by 2–3$\times$ and hid a real cost. Training-time monitors are not evaluation metrics.

B Full Mechanism-A Measurements

Tab. 8 gives the cross-class cosine similarity of c at each of the five sampled timesteps for all 12 checkpoints. Two patterns are visible that the main-text average compresses away. First, the homogenized models (JiT, SiT-XL/REPA, MDT, LightningDiT) are saturated at *every* timestep, not on average—there is no timestep at which their class codes separate. Second, the label-informative models show a characteristic rise with t: the timestep embedding grows relative to the class embedding as $t \to 1$, so homogenization is worst in the high-noise regime where the conditioning matters most for global structure.

Table 8: Cross-class cosine similarity of c at five timesteps. Flow-matching models use $t \in [0, 1]$; the DDPM lineage uses $t \in [0, 999]$; columns are the normalized ordinal positions of the sampled points.

Model	Domain	$t \approx 0$	0.25	0.5	0.75	0.999
PixelDiT-XL (ep80) (Yu et al., 2026)	pixel	0.481	0.304	0.549	0.723	0.836
PixelDiT-XL (ep160) (Yu et al., 2026)	pixel	0.515	0.441	0.678	0.816	0.894
PixelDiT-XL (ep320) (Yu et al., 2026)	pixel	0.443	0.552	0.760	0.870	0.927
PixelDiT-XL (IN-512) (Yu et al., 2026)	pixel	0.388	0.703	0.873	0.940	0.967
PixNerd-XL/16 (Wang et al., 2026a)	pixel	0.521	0.435	0.578	0.696	0.822
PixelFlow (Chen et al., 2025)	pixel	0.568	0.919	0.982	0.993	0.991
JiT-B/16 (Li & He, 2026)	pixel	0.985	0.998	0.998	0.997	0.994
Latent DiT-XL/2 (Peebles & Xie, 2023)	latent	0.937	0.344	0.569	0.762	0.814
DDT-XL/2 (Wang et al., 2026b)	latent	0.613	0.760	0.835	0.902	0.942
MDT-XL/2 (Gao et al., 2023)	latent	0.997	0.995	0.992	0.989	0.990
LightningDiT-XL (Yao et al., 2025)	latent	0.976	0.997	0.997	0.997	0.997
SiT-XL (REPA) (Yu et al., 2025b)	latent	1.000	1.000	0.999	0.999	0.995

Fig. 9 plots the same measurements as curves for better perception.

B.1 Pre-registered predictions, verbatim

Before measuring the four held-out 2025-era architectures we recorded the following predictions in writing, based solely on each model's timestep schedule and t-sampling convention as read from its configuration:

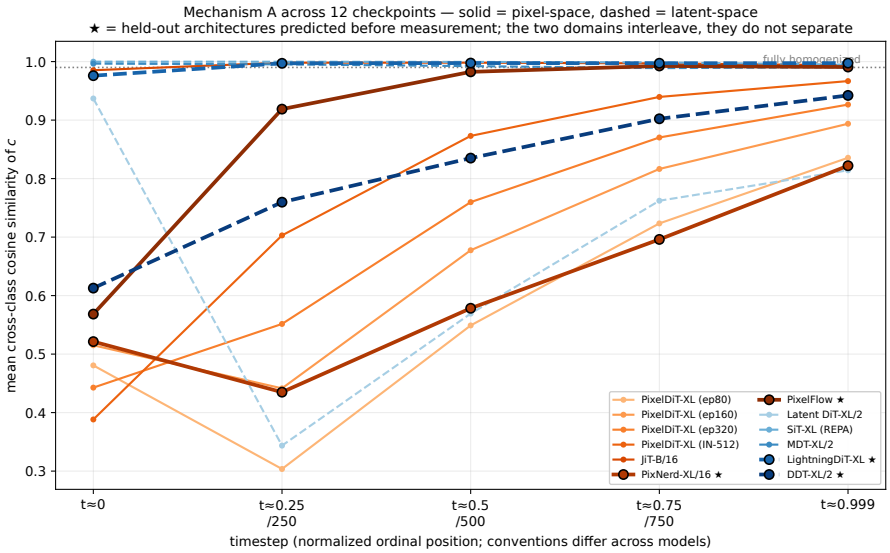

Figure 9: **Table 8 as curves.** Solid lines are pixel-space models, dashed are latent-space; ⋆ marks the four held-out architectures. The two domains interleave at every timestep rather than separating.

> **LightningDiT-XL** — `max_period`=10000 with continuous t: predict *homogenizes*. **DDT-XL/2** — schedule matched to its sampling range: predict *resists*. **PixNerd-XL/16** — matched: predict *resists*. **PixelFlow** — matched: predict *resists*.

Three of four were correct. **PixelFlow was the miss**: its schedule is matched, yet it reaches $\cos(c) = 0.891$ (ratio 6.72), saturating above 0.98 for $t \geq 0.5$. On inspection the ratio is driven by learned gain in the timestep MLP rather than by the frequency encoding. We report this as a correction to the mechanism rather than an outlier: **schedule mismatch is a sufficient but not necessary cause of a high ratio**, and the diagnostic must therefore be computed from trained weights. Excluding PixelFlow would *improve* every statistic we report (r $0.828 \to 0.823$, p $8.8 \times 10^{-4} \to 1.9 \times 10^{-3}$), which is precisely why we keep it.

B.2 The `max_period` ablation: the effect is baked into weights

A natural question is whether homogenization can be undone at inference by recomputing the timestep embedding with a corrected schedule. It cannot. Holding JiT-B/16's trained `t_embedder` weights fixed and varying only the input frequency encoding leaves the cosine similarity essentially unchanged (Tab. 9): the MLP's learned gain dominates whatever frequency spectrum it is fed. Any real fix must occur at training time—which is the motivation for CNB.

Table 9: The `max_period` ablation, in full. $\|y_{\text{emb}}\| = 2.71$ throughout.

Variant	mean $\|t_{\text{emb}}\|/\|y_{\text{emb}}\|$	mean $\cos(c)$
original (`max_period`=10000, raw $t \in [0, 1]$)	7.28	0.9945
`max_period`=1000, raw t	6.70	0.9936
`max_period`=100, raw t	7.05	0.9942
`max_period`=10, raw t (PixelDiT's choice)	10.86	0.9956
`max_period`=10000, t rescaled to $999t$	6.63	0.9939

A second finding from the same ablation deserves its own line: JiT's raw label table *alone*, with no timestep added at all, already sits at 0.822 cross-class cosine similarity, against PixelDiT's 0.0017 (near-orthogonal). Label-table homogenization is therefore a second, independent route that the norm ratio does not capture. Sec. J shows CNB inducing exactly this phenomenon on one architecture.

C THE CLASS-CODE FRAGILITY METRIC ϵ_{50}

ϵ_{50} carries a substantial share of the paper's argument—it is the functional readout in Tab. 2 and the response variable in the out-of-sample prediction of Sec. 5.2—so this appendix gives its definition, its exact computation, its invariances, and its relation to existing measurements.

C.1 WHY A THIRD STATISTIC

Mechanism-A is diagnosed by two quantities: the norm ratio $\|t_{\text{emb}}\|/\|y_{\text{emb}}\|$, a property of the trained weights, and $\cos(c)$, a property of the code geometry. Neither states a consequence. A cosine of 0.999 sounds alarming, but cosine is unitless and has no operational reading: nothing about the number itself says whether the remaining angular budget is enough to separate 1000 classes. ϵ_{50} answers that question in units the conditioning pathway actually experiences—*how much of the conditioning vector can be destroyed before the class it names stops being recoverable*. It also has a practical virtue that matters for our design: it needs only an embedding table, so it can be reported for third-party checkpoints we cannot retrain and cannot always sample from.

C.2 DEFINITION

Let $c_1, \dots, c_K \in \mathbb{R}^D$ be the class-conditioning codes at a fixed timestep, $K = 1000$. Perturb each code with isotropic Gaussian noise scaled to a fixed fraction of its own norm,

$$c'_k \;=\; c_k \;+\; \epsilon\,\|c_k\|\,\frac{n}{\sqrt{D}}, \qquad n \sim \mathcal{N}(0, I_D). \tag{3}$$

Since $\mathbb{E}\|n/\sqrt{D}\|^2 = 1$, the perturbation has expected norm exactly $\epsilon\|c_k\|$. ϵ **is therefore the perturbation size as a fraction of the code's own length**, which makes it dimensionless and independent of how a given architecture chooses to scale its conditioning vectors.

Decode by nearest neighbour against the clean table under the same cosine geometry used to define $\cos(c)$,

$$\hat{k}(c'_k) \;=\; \arg\max_j \,\langle \hat{c}'_k, \hat{c}_j \rangle, \qquad \text{acc}(\epsilon) \;=\; \frac{1}{K}\sum_{k=1}^{K} \mathbb{P}\big[\hat{k}(c'_k) = k\big], \tag{4}$$

where $\hat{\cdot}$ denotes ℓ_2 normalization. $\text{acc}(0) = 1$ and acc decays monotonically toward chance, $1/K$. We define

$$\epsilon_{50} \;=\; \inf\{\epsilon : \text{acc}(\epsilon) \le 1/2\}, \tag{5}$$

estimated by linear interpolation between the two bracketing grid points. Low ϵ_{50} means the class codes are packed tightly enough that a small disturbance anywhere in the adaLN (Peebles & Xie, 2023) pathway maps one class onto another.

C.3 EXACT COMPUTATION

Fig. 10 plots the underlying $\text{acc}(\epsilon)$ curves; the threshold is the point at which each crosses the dotted line. The full procedure:

- **Codes.** c is formed exactly as the architecture forms it: $c = y_{\text{emb}} + t_{\text{emb}}$ for the SiT/DiT/MDT lineage, and $c = \text{SiLU}(y_{\text{emb}} + t_{\text{emb}})$ for PixelDiT, DDT and PixNerd, which apply the nonlinearity before adaLN. Getting this wrong changes the geometry, so it is read from each model's own forward pass rather than assumed.
- **Timesteps.** Five per checkpoint, $t \in \{0, 0.25, 0.5, 0.75, 0.999\}$ for flow-matching models and the corresponding ordinal positions $\{0, 250, 500, 750, 999\}$ for the DDPM lineage. ϵ_{50} is computed independently at each and the five values are averaged; the per-timestep values behind every number in Tab. 2 are given in Tab. 10.
- **Sampling.** 24 independent noise draws per grid point, with a fresh n for every class in every draw ($24{,}000$ perturbed codes per grid point), under a fixed seed.
- **Grid.** $\epsilon \in [0, 1]$ in steps of 0.05, then $\epsilon \in [1.2, 6.0]$ in steps of 0.2. If recovery never falls to $1/2$ within this range the sweep limit is returned, so a reported value of 6.0 at a given timestep should be read as "at least 6.0" rather than as a measurement.

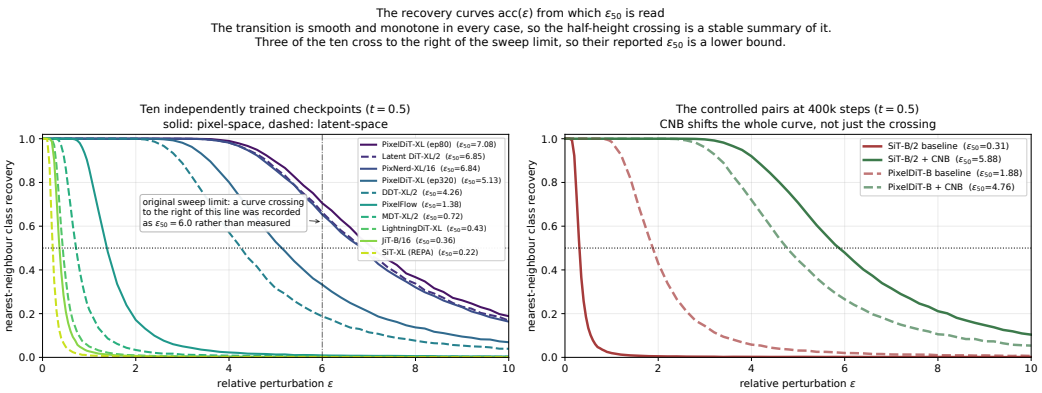

Figure 10: **The recovery curves behind ϵ_{50}.** Left: the ten independently trained checkpoints. Right: the two controlled pairs at 400k. CNB shifts the entire curve, not just its crossing point. Curves are drawn at $t = 0.5$; the tabulated ϵ_{50} averages five per-timestep thresholds, so the crossings here need not equal the tabulated values. The dot-dashed line marks the sweep limit of 6.0, whose consequences are analysed in property (iv) below.

- **Cost.** Seconds per checkpoint on CPU. No sampling, no dataset, no classifier.

Table 10: **Per-timestep ϵ_{50} behind every averaged value in Tab. 2.** Underlined entries sit at the sweep limit and are lower bounds rather than measurements (see property (iv) below); no arm of our own controlled pairs is affected. Two patterns the average hides: the homogenized checkpoints are fragile at *every* timestep rather than on average, and the SiT baseline's mean is carried almost entirely by $t = 0.999$, where the timestep embedding is largest.

Model	$t \approx 0$	0.25	0.5	0.75	0.999	mean
PixNerd-XL/16 (Wang et al., 2026a)	6.00	6.00	6.00	5.79	4.39	5.64
PixelDiT-XL (ep80) (Yu et al., 2026)	6.00	6.00	6.00	5.57	4.29	5.57
PixelDiT-XL (ep160) (Yu et al., 2026)	6.00	6.00	5.96	4.51	3.43	5.18
Latent DiT-XL/2 (Peebles & Xie, 2023)	2.61	6.00	6.00	5.08	4.50	4.84
PixelDiT-XL (ep320) (Yu et al., 2026)	6.00	6.00	5.13	3.76	2.83	4.74
DDT-XL/2	6.00	5.12	4.26	3.26	2.52	4.23
PixelDiT-XL (IN-512) (Yu et al., 2026)	6.00	5.69	3.70	2.55	1.89	3.97
PixelFlow (Chen et al., 2025)	6.00	2.97	1.38	0.90	1.00	2.45
MDT-XL/2 (Gao et al., 2023)	0.48	0.56	0.72	0.87	0.80	0.68
LightningDiT-XL (Yao et al., 2025)	1.31	0.47	0.43	0.46	0.44	0.62
JiT-B/16 (Li & He, 2026)	1.03	0.37	0.36	0.47	0.67	0.58
SiT-XL (REPA) (Yu et al., 2025b)	0.12	0.15	0.22	0.35	0.68	0.30
Controlled pairs (ours):						
SiT-B/2 @100k, baseline	0.28	0.39	0.62	1.36	4.78	1.49
+ CNB	5.66	5.77	5.71	5.80	5.74	5.74
SiT-B/2 @400k, baseline	0.14	0.19	0.31	0.66	4.29	1.12
+ CNB	5.51	5.81	5.88	5.94	5.74	5.78
PixelDiT-B @100k, baseline	0.92	1.48	1.97	2.37	1.80	1.71
+ CNB	5.19	5.39	5.52	5.68	5.69	5.50
PixelDiT-B @400k, baseline	0.90	1.39	1.88	2.19	1.48	1.57
+ CNB	3.59	4.35	4.76	5.06	4.62	4.48

The curves in Fig. 10 are smooth and monotone at the sampling density we use, so the interpolated crossing is stable: it is not sensitive to where between two grid points the threshold falls, and the differences we report between arms (2.9× and more) are far larger than the width of the region in which a curve is ambiguous.

C.4 WHAT ϵ_{50} IS, AND IS NOT, SENSITIVE TO

Three properties govern how the metric may be used.

(i) It is invariant to a global rescaling of c. Replacing every c_k by αc_k scales the perturbation in Eq. 3 by the same α and leaves the cosine decode unchanged, so ϵ_{50} is identical. This is not a technicality. CNB changes the norms of t_{emb} and y_{emb} by construction, and a fragility metric that improved simply because norms moved would be circular. This one cannot.

(ii) It is sensitive to a shared offset, deliberately. Adding a common vector μ to every code increases $\|c_k\|$—and hence the noise—without increasing any inter-class distance, so ϵ_{50} falls. That is the correct response: a shared offset is conditioning capacity spent on nothing. It also means the metric can detect CNB's own side effect, the label-table inflation analysed in Sec. J, and it does: PixelDiT+CNB's ϵ_{50} erodes between 100k and 400k while SiT+CNB's holds (Tab. 2). We regard a metric that penalizes our own intervention where the intervention misbehaves as a point in its favour.

(iii) It is a property of the code table, not of the network. ϵ_{50} asks whether class identity survives *in principle*, under an optimal nearest-neighbour readout. The trained network reads c through learned adaLN projections, which are not optimal readouts. ϵ_{50} is therefore an **upper bound** on conditioning robustness: a low ϵ_{50} implies the information is not there to recover, while a high ϵ_{50} does not guarantee the network exploits it. We use the metric in that direction only. Every functional claim in Sec. 5 rests on classification of generated samples, not on ϵ_{50}. Relatedly, isotropic Gaussian noise is not the perturbation the pathway experiences in practice, where activation noise is structured and input-dependent; ϵ_{50} is a geometric summary with an operational reading, not a measured failure rate.

(iv) The top of its range is bounded by the sweep. Because the sweep ends at $\epsilon = 6.0$, a checkpoint whose codes survive beyond that is recorded at the limit rather than measured. Fig. 10 shows that this binds for the most robust checkpoints: at $t = 0.5$, PixelDiT-XL (ep80), Latent DiT-XL/2 and PixNerd-XL/16 all cross $1/2$ to the right of the limit. The reported ϵ_{50} for those models is therefore a lower bound. The distortion is one-directional—it compresses the top of the range and leaves the fragile checkpoints untouched—so both the $\approx 20\times$ spread and the ratio–ϵ_{50} correlation in Sec. 5.2 understate rather than overstate the effect. Our own controlled pairs are unaffected: no arm of either pair is pinned at the limit at any timestep, so no CNB claim depends on this.

C.5 A CLOSED-FORM ANCHOR

ϵ_{50} is defined operationally, which makes it easy to compute and hard to reason about. A first-order estimate is available. For K codes of common norm at mean pairwise cosine ρ, two codes are separated by $\|c\|\sqrt{2(1-\rho)}$, so the decision boundary lies at half that distance. Isotropic noise of total norm $\epsilon\|c\|$ contributes only $\epsilon\|c\|/\sqrt{D}$ along any one direction, but the decode fails if *any* of the $K-1$ competitors is crossed, and the maximum of $K-1$ Gaussians sits about $\sqrt{2\ln K}$ standard deviations out. Equating the two gives

$$\epsilon_{50} \;\sim\; \sqrt{\frac{D\,(1-\rho)}{2}} \;\Big/\; \sqrt{2\ln K}. \tag{6}$$

This is a scaling argument, not a fit: it assumes equal norms and a single common cosine, where real code tables have a distribution of both, and no claim in the paper rests on it. Its use here is interpretive. It says that ϵ_{50} and $\cos(c)$ are related but not redundant, because the map from ρ to ϵ_{50} is strongly compressive as $\rho \to 1$: the gap between 0.99 and 0.999, nearly invisible on a cosine axis, is a factor of $\sqrt{10}$ in tolerable perturbation. That is precisely the range in which the homogenized checkpoints sit, and it is why we report ϵ_{50} alongside the cosine rather than the cosine alone. It also makes the dependence on width explicit—all else equal, a wider conditioning vector buys robustness as $\sqrt{D}$—which is worth keeping in mind when comparing ϵ_{50} across model scales, as Table 2 does.

C.6 Relation to existing measurements

ϵ_{50} is assembled from standard parts. We are not aware of a prior application to diffusion conditioning embeddings, and claim novelty only for that application and for the finding that a two-norm architectural statistic predicts it out-of-sample; the construction itself sits at the intersection of three existing lines of work.

Minimum-distortion robustness. The adversarial-examples literature characterizes a model by the smallest perturbation that changes its decision (Szegedy et al., 2013). ϵ_{50} asks the same question with two substitutions: the worst case is replaced by the average case over isotropic noise, and the perturbation norm is expressed relative to the input rather than absolutely, which is what makes values comparable across architectures with different conditioning scales.

Accuracy-versus-noise curves. Corruption-robustness benchmarks sweep a severity parameter and report the resulting accuracy curve (Hendrycks & Dietterich, 2019); Gilmer et al. (2019) specifically relate additive Gaussian noise magnitude to error rate, and argue that noise robustness and adversarial robustness are facets of the same underlying quantity. Our $\mathrm{acc}(\epsilon)$ is an object of this kind. Reducing it to a single threshold at half-height, rather than to an area, follows the psychophysical convention in which the 50%-correct point defines the threshold (Wichmann & Hill, 2001), a convention already imported into vision-model evaluation by Geirhos et al. (2018).

Associative retrieval radius. Recovering a stored pattern from a corrupted copy is the defining operation of an associative memory, and the result—classical and modern alike—is that tolerable corruption is governed by how well separated the stored patterns are (Ramsauer et al., 2021). A class-code table read by nearest neighbour *is* such a memory, with $K = 1000$ stored patterns, and ϵ_{50} measures its basin radius. We find this the most useful framing of what homogenization is: a **capacity** failure. The codes crowd, the basins shrink, and the conditioning channel carries less class information than its dimension would allow—which is why the damage shows up in class controllability and not in FID.

Distinction from sample-level measures. Classification Accuracy Score (Ravuri & Vinyals, 2019) evaluates conditioning fidelity by classifying generated samples, and is the right instrument for the question of whether a model *does* respect its conditioning; we use exactly that measure for the functional claims in Sec. 5. ϵ_{50} is complementary rather than competing: it is computed on embeddings alone, costs no sampling, and therefore scales to checkpoints for which generating $10{,}000$ images per configuration is impractical, at the price of measuring a capacity bound rather than a behaviour.

D Mechanism-B Details

D.1 The intervention, in full

Mechanism B says that by mid-depth most prompts' token representations have drifted toward a common, converged direction. If that is true, then at such a block a prompt's representation should decompose into a *shared* component that carries little prompt-specific information and a small *residual* that carries the rest. The intervention tests this by manipulating the residual alone.

Where it acts. We hook a single joint-attention block ℓ and leave everything else untouched: no fine-tuning, no inversion, no extra parameters, no change to the sampler. Concretely we replicate the transformer's own `forward` (patch embed $\rightarrow$ time/text embed $\rightarrow$ context embed $\rightarrow$ blocks $\rightarrow$ norm/proj $\rightarrow$ unpatchify) so that the block loop can be interrupted at ℓ, and we apply the edit to the conditional branch only, leaving the unconditional branch of CFG untouched.

The shared component is computed at the same timestep and latent. Let $E_p^{(\ell)}$ denote the `encoder_hidden_states` of prompt p after block ℓ. The shared component must be the population direction *at that point in the trajectory*, not a static average, so we recompute it at every

denoising step:

$$\bar{E}^{(\ell)}(x_t, t) \;=\; \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} E_r^{(\ell)}(x_t, t), \tag{7}$$

where $\mathcal{R}$ is a fixed set of $|\mathcal{R}| = 20$ unrelated reference prompts spanning animals, scenes, food, objects and abstract imagery, and every $E_r^{(\ell)}$ is evaluated on the *same* evolving image latent x_t as the prompt being edited. Using a per-step, per-latent mean matters: a mean taken from a different timestep is not the direction the current representation has converged toward, and using it degrades the edit.

Manipulation 1: residual scaling. With $R_p^{(\ell)} = E_p^{(\ell)} - \bar{E}^{(\ell)}$,

$$\tilde{E}_p^{(\ell)} \;=\; \bar{E}^{(\ell)} \;+\; \alpha\, R_p^{(\ell)}, \tag{8}$$

and the remaining blocks $\ell+1, \ldots, L$ run unmodified. By construction $\alpha = 1$ reproduces the original sample exactly, which is a useful internal check: any visible change at $\alpha = 1$ would indicate a bug in the replicated forward pass. $\alpha = 0$ replaces the prompt by the population mean, and $\alpha > 1$ exaggerates the prompt-specific component.

Manipulation 2: directional analogy. The second variant tests whether the residual space is linear enough to support transfer between prompts. From a minimal reference pair differing in one attribute (e.g. "a red sports car…" vs. "a blue sports car…") we form

$$D^{(\ell)} \;=\; E_B^{(\ell)} - E_A^{(\ell)}, \qquad \tilde{E}_p^{(\ell)} \;=\; E_p^{(\ell)} + s\, D^{(\ell)}, \tag{9}$$

and apply D to an *unrelated* target prompt p. This is the variant used for all CLIP-scored attribute results (Table 11), with $s \in \{1, 2\}$. Note that D is computed at the same block and timestep as the injection, so it is a direction in the evolved representation rather than in the raw text encoder output.

Trajectory branching, so differences are attributable. All α (or s) values are produced by branching *one* trajectory: the sampler runs once up to block ℓ at each step, then one extra tail pass per setting. Every variant therefore shares its noise, its schedule and its entire history up to the intervention point, so visible differences are attributable to the intervention rather than to sampling variance. This is why the figures can be read as controlled comparisons.

Which block, and why it differs by model. We intervene at SD3.5 block 4 (of 24), FLUX block 18 (of 19 dual-stream, before the 38 single-stream blocks), PixelDiT-T2I block 6 (of 14) and Qwen-Image blocks 26 and 35 (of 60). These are chosen from each model's own depth profile (Fig. 7) as a point past the initial rise but before the representation has fully converged—late enough that a shared component exists to subtract, early enough that downstream blocks can still propagate the change. The FLUX choice is deliberately just before the dual→single-stream boundary, which is what makes its damping interpretable: the modified signal must survive 38 subsequent blocks in which text is concatenated with image tokens under shared weights.

A negative result this design exposes. Because the same code applies the same manipulation at a chosen block in four architectures, effect sizes are comparable across them—and they do not track local homogenization. FLUX.1 has the *highest* within-prompt similarity at its intervention block yet the weakest effect (Tab. 11). The account consistent with the data is that what matters is how much subsequent depth still treats text as a separately evolving stream, not how converged the representation already is. We state this as a hypothesis with a concrete falsification test—intervene at a later FLUX single-stream block, where less dilution remains, and the effect should partially recover—rather than as a finding.

Ruling out a noise-injection confound. FLUX's decline through its single-stream blocks could in principle be an artifact of image tokens carrying independent noise per prompt. We reran the measurement with *identical* image noise across all prompts; the decline persists. It is therefore a structural property of single-stream mixing, in which image tokens—which stay diverse—are concatenated with text tokens under shared weights and mixed back into the text representation.

SD3.5's shape. After correcting an analysis bug in an earlier version of our code, SD3.5 shows a structurally similar early-peak-then-decline despite having *no* single-stream blocks. This suggests that peak-then-partial-reversal may be more general than FLUX's single-stream explanation alone accounts for. We flag this as unresolved rather than fold it into the mechanism.

Source-level verification of cross-attention immunity. For SANA1.5, PixArt-Σ and Wan2.1-T2V we verified directly from the modelling code that `encoder_hidden_states` is computed once and never reassigned inside the block loop; only the image/video latent updates. For Wan2.1-T2V the timestep path is additionally separate, modulating the latent through a scale-shift table while never being combined with the text embedding. These are zero-homogenization boundary conditions established without running a single generation.

E The Qwen-Image Anomaly

Qwen-Image is our least homogenized token-conditioned model despite being by far the deepest, and we do not have a confirmed explanation. We captured per-block multiplicative gate magnitudes on the text stream alongside within-prompt similarity across all 60 blocks. The correlations are weak: combined gate magnitude vs. similarity $r = -0.23$; `txt_gate1` alone $r = 0.03$; `txt_gate2` (the MLP-branch gate) $r = -0.40$. The clearest qualitative signal is local rather than global: over the final nine blocks `txt_gate2` collapses from $\sim$133 to $\sim$3.8 while similarity plateaus, consistent with "the gate shutting off freezes the text stream at whatever similarity it has reached." One local pattern is not a mechanism, and we report this as our most fragile Mechanism-B claim.

F Editing Validation in Full

Tab. 11 gives per-model, per-attribute CLIP-scored success rates for the residual-manipulation experiment of Sec. 7. All nine model×attribute distributions are significant under both a one-sample t-test and a Wilcoxon signed-rank test. Qualitative results are in Fig. 11 (across models, one attribute) and Fig. 12 (across attributes, one model).

Table 11: Residual-only editing, CLIP success rate and mean margin. A trial succeeds when $\cos(\text{edited}, \text{target}) > \cos(\text{unedited}, \text{target})$. PixelDiT-T2I is additionally shown at 2× edit strength.

Model	Attribute	n	Success	Mean margin
SD3.5 (Esser et al., 2024)	time-of-day	14	13/14 (92.9%)	+0.031
	weather	14	14/14 (100%)	+0.084
	style	14	14/14 (100%)	+0.075
FLUX.1 (Labs et al., 2025)	time-of-day	13	12/13 (92.3%)	+0.012
	weather	14	14/14 (100%)	+0.033
	style	14	13/14 (92.9%)	+0.037
PixelDiT-T2I (Yu et al., 2026)	time-of-day	14	9/14 (64.3%)	−0.001
	weather	14	9/14 (64.3%)	+0.007
	style	14	12/14 (85.7%)	+0.050
PixelDiT-T2I (2×) (Yu et al., 2026)	time-of-day	14	11/14 (78.6%)	+0.016
	weather	14	10/14 (71.4%)	+0.031
	style	14	13/14 (92.9%)	+0.058

The reliability map is itself a readout of the mechanisms. This is the per-cell detail behind the effect-size argument of Sec. 7, where the point is that the map tracks stream writeability—the variable Sec. 6 identifies as governing token homogenization—and so reads as a validation of Mechanism B rather than as a demonstration that editing works. SD3.5, whose conditioning path is least diluted, shows the largest effects. FLUX shows a consistent $\sim$2.6× damping across all three attributes, matching the single-stream dilution identified in Sec. 6. PixelDiT-T2I *reorders* the map entirely—style strongest, time-of-day and weather much noisier—rather than uniformly scaling it. The map has a second axis, within a model rather than across models, and Fig. 12 is its qualitative

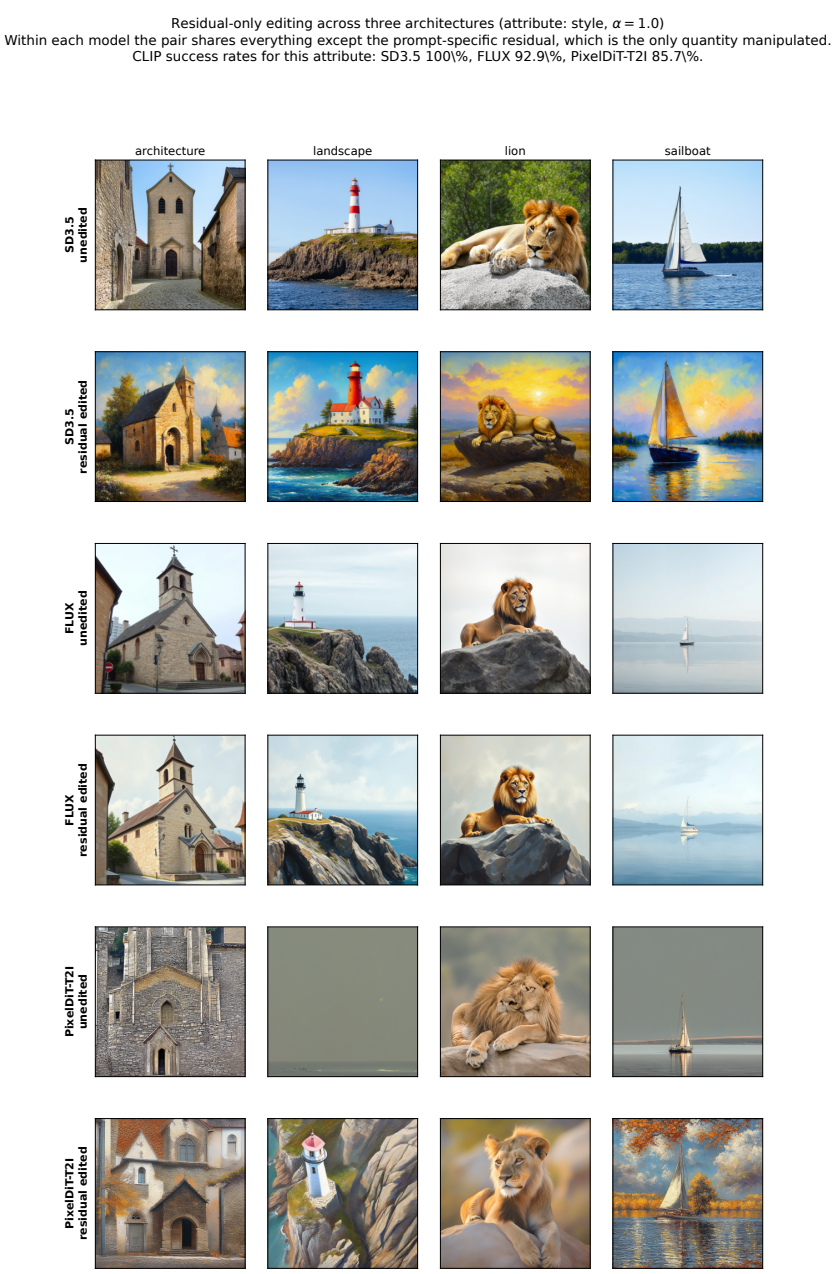

Figure 11: **Residual-only editing across three architectures** (style, $\alpha = 1.0$). Within each model, the pair differs *only* in the prompt-specific residual. The damping visible on FLUX relative to SD3.5 is the same $\sim 2.6\times$ effect the CLIP margins record, and is consistent with the single-stream dilution of Sec. 6. The blank PixelDiT-T2I "landscape" panel is a genuine generation failure of that model at this prompt, left in rather than replaced.

counterpart. On SD3.5 the visible change is largest for weather and style and smallest for time-of-day, in the same order as their margins ($+0.084$, $+0.075$, $+0.031$). That ordering is precisely what the pixel-space model inverts: PixelDiT-T2I's weather margin collapses to $+0.007$ at $1\times$ while style holds at $+0.050$, so what looks like a uniform weakening in the aggregate is, attribute by attribute, a change of rank. A method whose effect were simply attenuated by depth would preserve the within-model ordering and scale it; this does not. Cross-model gaps are mostly significant (SD3.5 $>$ FLUX on all three, $p \leq 0.006$; SD3.5 $>$ PixelDiT-T2I on weather and style, $p \leq 0.0005$), with one honest exception we report rather than smooth over: time-of-day vs. PixelDiT-T2I, $p \approx 0.07$.

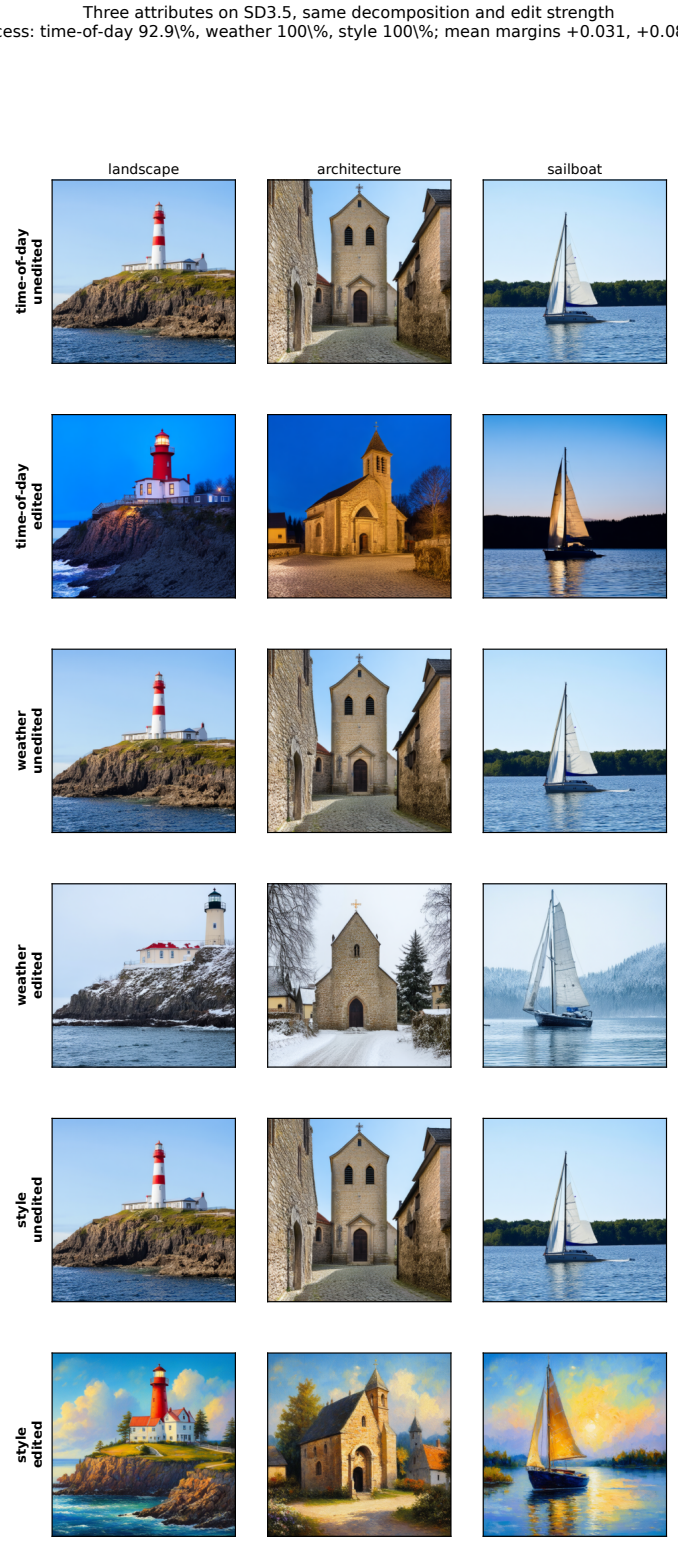

Figure 12: **Three attributes on SD3.5**, same decomposition and edit strength. The visible ordering follows the CLIP margins of Tab. 11.

Two failed predictors of when the intervention works. We tested two cheap, directly computable candidates for predicting effect size across models and both failed. Local homogenization strength at the intervention block predicts the *opposite* of what is observed: FLUX has the highest local similarity at its intervention point and the weakest effect. The account that survives—untested as a causal claim—is that what matters is how much subsequent processing still treats text as a separately evolving stream, which would connect effect size back to the dual/single-stream distinction. We state this as a hypothesis with a concrete falsification test (intervene at a later FLUX single-stream block) rather than as a finding.

G Video-Domain Extension, and a Second Scoring Failure

We applied the same decomposition to Wan2.1-T2V. Scene-level attributes (time-of-day, weather, style) fail at *two* structurally different injection points—pre-projection and post-projection—which rules out "wrong injection site" as the explanation. That is a stable negative result.

Colour appeared to transfer. Under CLIP scoring, 9/16 targets succeed at $\alpha = 1.0$, and sorting by margin puts truck ($+0.030$), bird ($+0.020$) and bicycle ($+0.019$) above a tail that falls away to $+0.009$ and below, with the winners being exactly the targets with a large, camera-facing, naturally variable-coloured surface. This is the result an earlier version of this work reported.

We no longer believe the colour result as stated. The direction is built from "a red sports car…" $\rightarrow$ "a blue sports car…", so edited frames should move *toward* blue. Inspecting the frames, they frequently do not. A crude whole-frame proxy—the mean of the blue minus red channel, where a positive shift means the frame became bluer—gives Tab. 12:

Table 12: Whole-frame $B - R$ shift from the red$\rightarrow$blue edit, averaged over three frames. Positive means the frame moved toward blue, as intended. Two of the three highest-margin "successes" move the *wrong* way.

Target	α=1.0	α=2.0	CLIP verdict
truck	$+5.7$	$+6.6$	success ($+0.030$)
van	$+0.4$	$+2.5$	success
bus	-1.7	$+1.5$	failure
balloon	-0.6	-4.6	success
bicycle	-1.2	-2.7	success ($+0.019$)
motorcycle	-6.6	-4.1	success
horse	-6.3	-6.9	success
bird	-6.5	-13.4	success

Only truck moves meaningfully toward blue. Visual inspection agrees: the truck shifts pink$\rightarrow$red and the motorcycle red$\rightarrow$magenta—a colour change, but not the requested one. The CLIP margin is evidently responding to something other than the target hue.

Caveat on our own check. The $B - R$ proxy is whole-frame and therefore contaminated by background: the truck sequence has a red canyon behind it, and small-in-frame targets contribute little signal. It is suggestive, not conclusive. A proper test requires segmenting the object and comparing its hue distribution, which we have not run.

What we conclude. We do not claim colour transfer in video. We report the CLIP-scored numbers, the contradicting hue measurement, and the fact that we found the discrepancy only by rendering the frames and looking at them. This is the *second* scoring failure we encountered in this domain. The first was starker: an early analysis reported 16/16 successes when three of four attributes had produced noise-collapsed video that CLIP nonetheless scored as closer to the target text, caused by the enormous variation in raw direction magnitude across attributes in Wan's pre-projection embedding space. Two independent failures of the same automated metric in the same domain is the finding worth carrying forward: **CLIP margins transplanted into a new modality can be confidently wrong, and a positive margin is not evidence that the intended edit occurred.** Figure 13 shows the frames so a reader can judge directly.

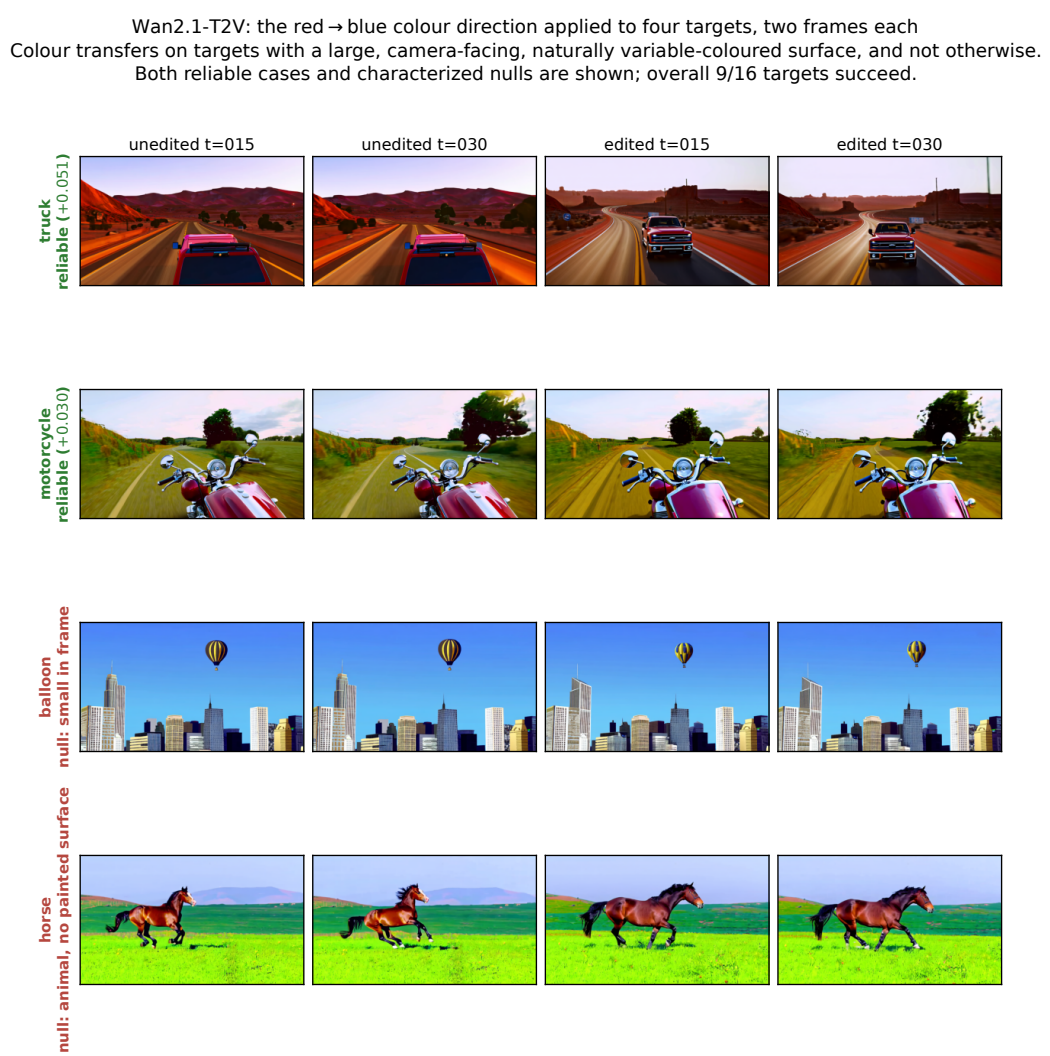

Figure 13: **Wan2.1-T2V frames for the red→blue direction.** The two rows CLIP scores highest are shown alongside two characterized nulls. Neither high-margin case is visibly blue: the truck shifts pink→red and the motorcycle red→magenta. We include this figure because it contradicts the metric, not because it supports it.

H Evaluation Protocol

Paired design. For every cell, both arms receive the same initial noise x_T and the same class label for every sample, so the comparison is within-pair. We use McNemar's exact binomial test on the discordant counts rather than a χ^2 approximation, which is correct at small discordant counts. Sampling covers all 1000 ImageNet classes $\times$ 10 samples.

Generate once, score twice. The identical image tensors feed the ResNet-50 classifier (centre-crop 224, standard ImageNet normalization) and the Inception-v3 feature extractor used for FID. This is what licenses the dissociation claim: a divergence between the metrics cannot be attributed to different samples, guidance settings, or checkpoints.

FID reference. FID is computed against the standard ADM/OpenAI ImageNet-256 reference statistics. We write FID-10k throughout and never compare against published FID-50k numbers.

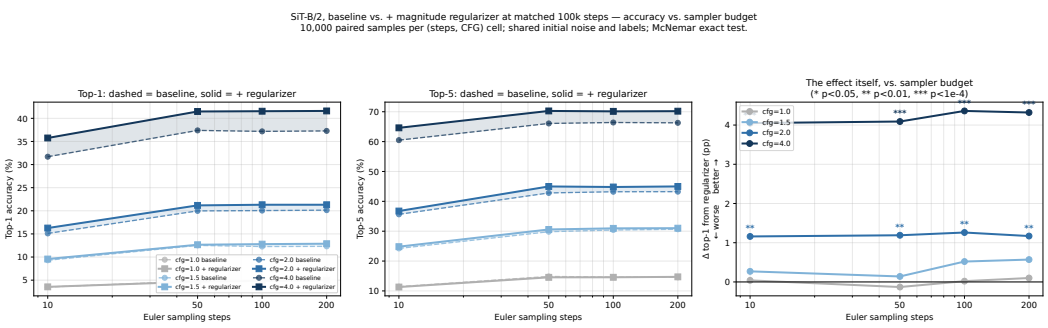

Figure 14: **Sampler-budget sweep, SiT-B/2.** Top-1 and top-5 against Euler steps (baseline dashed, CNB solid, gap shaded) and the effect itself. The flat lines in the right panel are the finding: a 20× change in sampler budget does not move the gap. Baseline accuracy saturates by ∼50 steps.

Sampler-budget sweep. Tab. 13 gives the full grid and Fig. 14 plots it. Both the accuracy effect and the FID delta are essentially flat above 50 steps; the mild extra FID cost at 10 steps saturates quickly, so the bulk of the cost is a property of the model rather than of discretization.

Table 13: SiT-B/2 (100k), the full sampler-budget grid behind Tab. 6 (right). Δtop-1 in pp (left) and ΔFID (right).

	Δ top-1 (pp)				Δ FID			
Steps	cfg 1	1.5	2	4	cfg 1	1.5	2	4
10	+0.04	+0.27	+1.16	+4.05	+9.21	+8.17	+6.78	+1.19
50	−0.13	+0.14	+1.19	+4.09	+8.16	+7.44	+5.95	+1.20
100	+0.02	+0.52	+1.26	+4.36	+8.04	+7.51	+6.04	+1.46
200	+0.10	+0.57	+1.17	+4.32	+8.06	+7.51	+6.11	+1.52

FID seed variance. Because every accuracy claim carries a p-value while FID was a single measurement, we replicated cfg=6 and cfg=8 across three independent noise seeds (10,000 paired samples each). Within a seed both arms share x_T, so ΔFID is itself a paired quantity with a tighter spread than the raw FIDs. We obtain $\Delta\text{FID} = -1.15 \pm 0.39$ (cfg=6, range $[-1.39, -0.70]$) and -2.15 ± 0.17 (cfg=8, range $[-2.27, -1.96]$), with **3/3 seeds favouring CNB at both settings**; at cfg=8 the effect is ∼13× its own standard deviation. Accuracy is markedly more stable than FID (sd 0.14–0.43pp), which is why we lead with accuracy and treat FID as corroboration. With $n = 3$ we report mean ± sd, the observed range and sign-consistency, and deliberately do *not* quote a t-test: three points estimate a standard deviation too poorly for the resulting p to mean what it appears to.

A Type-M error we caught by scaling up. An early pilot at $N = 1000$ paired samples reported +4.7 pp (+32% relative) at cfg=2; the $N = 10{,}000$ measurement gives +1.19 pp (+6%)—roughly a fourfold overestimate. This is the expected behaviour of a marginally powered test, which reaches significance mainly on draws where noise happens to favour the effect (Gelman & Carlin, 2014). The cfg=4 cell, where the true effect is large, was estimated far more accurately (+5.5 vs. +4.09 pp). No $N = 1000$ figure appears anywhere in this paper.

Training-time monitors are not evaluation metrics. An earlier version of this work quoted FID from a 100-sample proxy computed inside the training loop. Measured on the same checkpoint, that proxy read 262.5 against 113.0 at $N = 10{,}000$—a 2.3× inflation—and its noise entirely concealed a real FID cost on one architecture. We report this because the practice is common and the failure is silent.

I TRAINING CONFIGURATIONS

Tab. 14 lists both configurations. The matched pairs differ *only* in λ of Eq. 2 (0 for baseline, 0.5 for CNB). No other hyperparameter, data ordering, or schedule differs.

Table 14: Training configurations for the two controlled pairs.

	SiT-B/2 (latent)	PixelDiT-B (pixel)
Parameters	130.5M	184.7M
Input	$4 \times 32 \times 32$ VAE latent	$3 \times 256 \times 256$ RGB
Dataset	ImageNet-256	ImageNet-256
Batch size	32 (global)	32 (global)
Optimizer	AdamW, lr 10^{-4}, wd 0	AdamW, lr 10^{-4}, wd 0
Precision	bf16	bf16
Path / prediction	linear, v-prediction	linear flow matching
CFG dropout	0.1	0.1
EMA	–	0.9999
Steps	400k	400k
Checkpoints	100k/200k/300k/400k	100k/200k/300k/400k
λ (CNB)	0.5	0.5

J CNB INFLATES THE LABEL TABLE: A SIDE EFFECT OF THE OBJECTIVE

CNB holds the norm ratio at $\approx$1.0 on both architectures exactly as designed. But on PixelDiT $\cos(c)$ still drifts $0.63 \to 0.82$ between 100k and 400k steps, where SiT's CNB arm holds at $\approx$0.52. The cause is a second homogenization route that the ratio penalty not only fails to control but actively encourages (Tab. 15).

Table 15: Raw label-table statistics, no timestep added. $\cos(y, y)$ is the mean pairwise cosine similarity of the 1000 class embeddings; $\|\bar{y}\|/\|y_{\text{emb}}\|$ is the fraction of each embedding's norm carried by the component *shared* by all classes. Note that $\cos(y, y) \approx (\|\bar{y}\|/\|y_{\text{emb}}\|)^2$ in every row.

Model	Arm	$\cos(y, y)$		$\|\bar{y}\|/\|y_{\text{emb}}\|$	
		100k	400k	100k	400k
SiT-B/2	baseline	0.008	0.066	0.103	0.265
	+ CNB	0.015	0.101	0.127	0.319
PixelDiT-B	baseline	0.007	0.121	0.087	0.349
	+ CNB	0.072	**0.644**	0.270	**0.803**

The mechanism. The near-exact identity $\cos(y, y) \approx (\|\bar{y}\|/\|y_{\text{emb}}\|)^2$ says that label-table similarity *is* the squared fraction of each embedding carried by a component common to all classes. CNB penalizes $(\|t_{\text{emb}}\| - \|y_{\text{emb}}\|)^2/\|y_{\text{emb}}\|^2$, which the model can satisfy either by shrinking $\|t_{\text{emb}}\|$ or by growing $\|y_{\text{emb}}\|$. Growing $\|y_{\text{emb}}\|$ is cheap in a particular way: **adding a single shared offset raises all 1000 norms at once**, whereas growing them individually requires learning 1000 separate directions. A shared offset is by construction identical across classes, so it inflates $\cos(y, y)$. Eq. 2 therefore contains an implicit incentive to homogenize the label table—a genuine flaw in the objective, not an artifact of one architecture.

It affects both architectures; only the magnitude differs. CNB raises $\cos(y, y)$ in all four comparisons. On SiT the effect is small in absolute terms ($0.066 \to 0.101$ at 400k, still near-orthogonal); on PixelDiT it runs far, leaving only 0.596 of each embedding's norm class-specific while 80% is a single shared vector. We do not have an account of why PixelDiT is so much more susceptible. The baseline tables also drift upward with training ($0.103 \to 0.265$ shared fraction on SiT), so training alone pushes in this direction and CNB amplifies it.

This does not prevent CNB from working, and effect size does not read off the geometry. At 400k, PixelDiT's CNB arm has the *worse* conditioning geometry of the two ($\cos(c) = 0.72$ and $\epsilon_{50} = 4.48$, against SiT's 0.52 and 5.78) yet delivers the *larger* accuracy gain (+4.80 vs. +2.70 pp). We therefore make no claim that the size of the benefit is a function of the cosine or of ϵ_{50}, and we would caution against reading one off the other.

A concrete fix, untested. The diagnosis suggests a specific remedy rather than a vague direction: compute the penalty on *centred* embeddings, $\|y_k - \bar{y}\|$ in place of $\|y_k\|$, so that inflating the shared component no longer reduces the loss. Equivalently, one can add an explicit penalty on $\|\bar{y}\|/\|y_{\text{emb}}\|$. Either removes the degenerate solution while leaving the intended balancing intact. We did not run this variant and report it as the obvious next experiment.

K Visualizing What CNB Changes

Fig. 5 in the main text renders the conditioning geometry directly. Its left column is the full distribution of all $\binom{1000}{2} = 499{,}500$ pairwise cosine similarities between class codes; the right column is how many directions those codes span once each arm is centred on its own mean, summarised by the participation ratio $(\sum_i \lambda_i)^2 / \sum_i \lambda_i^2$ over the eigenvalues. This section records the reasoning behind that choice of visualization and one negative result.

The norm ratio across training. Fig. 4 in the main text draws the conditioning vectors at a single timestep, t=0.5. Fig. 15 gives the quantity CNB actually penalizes over all four checkpoints and all five sampled timesteps, and separates the two norms that form it. Two points the main-text figure cannot make on its own: the ratio's spread across timesteps is wide for the baseline and negligible under CNB, so the imbalance is not an artifact of the timestep we chose to draw; and the right panel shows that parity is reached by suppressing $\|t_{\text{emb}}\|$ rather than by inflating $\|y_{\text{emb}}\|$, which is the behaviour the relative denominator in Eq. 2 was designed to produce.

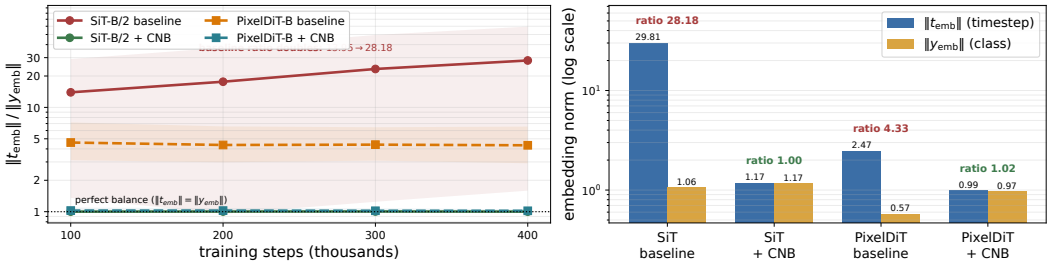

Figure 15: **The penalized quantity across training, per timestep. Left**: $\|t_{\text{emb}}\|/\|y_{\text{emb}}\|$ across saved checkpoints (log scale; the shaded band spans the five sampled timesteps). Both baselines sit far from parity at every checkpoint; both CNB arms stay near it (1.00 and 1.02). **Right**: the imbalance comes from an inflating timestep embedding, not a shrinking class embedding ($\|y_{\text{emb}}\| \approx 1$). CNB restores parity mainly by suppressing $\|t_{\text{emb}}\|$ (29.81 $\rightarrow$ 1.17 on SiT) rather than by inflating $\|y_{\text{emb}}\|$ (1.06 $\rightarrow$ 1.17).

Why this is not a t-SNE. A natural instinct is to visualize the class codes with t-SNE. That would be the wrong tool here, and misleadingly so: t-SNE normalizes global scale, so it expands a collapsed cone and a well-separated set to fill the canvas equally. The two arms would look similar and the collapse— the entire phenomenon—would be invisible. The two views above are chosen because neither can hide it: the cosine histogram makes no projection at all, and the variance spectrum preserves relative scale.

The effect does not appear as feature-space clustering. We also checked whether CNB's accuracy gain shows up as better clustering of *generated samples* in Inception feature space, using the same features FID is computed from. It does not, at a magnitude we can resolve. On ten visually distinct classes both arms reach 100% kNN accuracy—a ceiling that cannot register any difference. On ten confusable classes (dog breeds, chosen because our error analysis shows the failures are near-misses

between neighbours), kNN moves in CNB's favour but within noise: $93.7 \rightarrow 94.8\%$ on SiT and $87.7 \rightarrow 89.8\%$ on PixelDiT, against a binomial standard error of $\approx$1.0 and 1.3 pp at n=600, i.e. 1.1σ and 1.6σ. Silhouette scores are unchanged (0.155 vs. 0.153; 0.099 vs. 0.100). We report this because it delimits the claim: a $\sim$3 pp class-accuracy effect is real under a paired test with 10,000 samples but is *not* large enough to reorganise feature-space neighbourhood structure, and we would caution against reading a t-SNE of generated samples as evidence either way.

L Additional Quantitative Results

The 100k comparison. (Fig. 16). At 100k steps, before either model converges, the picture at each baseline's own optimum is: SiT baseline 41.04 FID / 43.25% top-1 at cfg=6, against CNB's 40.34 / 48.69%. The optima sit at cfg=6–8, characteristic of undertrained models; training to 400k moves both to cfg=4 and halves FID, which is the comparison reported in the main text. Fig. 17 shows the benefit against training duration at cfg=4.

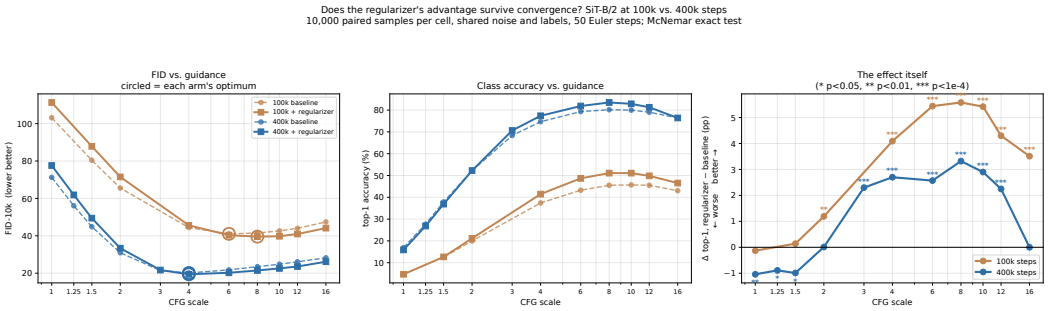

Figure 16: **SiT-B/2 at 100k vs. 400k across the guidance sweep.** Each arm's FID optimum is circled. Training halves FID and moves both optima from cfg=6–8 down to cfg=4, i.e. toward the conventional range.

Homogenization persists across training. Tab. 16 gives cross-class cosine similarity at five timesteps for both controlled pairs. The baselines are saturated at low timesteps throughout—SiT already reads 0.999 at t=0 at 100k—so there is little room left to rise. The mean over timesteps is non-monotonic on SiT ($0.922 \rightarrow 0.891 \rightarrow 0.920 \rightarrow 0.944$) because it is dominated by the noisy t=0.999 column; we therefore do *not* present the mean as evidence of a trend. The defensible statement is that baseline homogenization is high and stable while the norm ratio and ϵ_{50} (Tab. 2) continue to degrade.

Table 16: The same measurement as Tab. 8, for our two controlled pairs rather than public checkpoints.

Model	Arm	Steps	t=0	0.25	0.5	0.75	0.999
SiT-B/2	baseline	100k	0.999	0.998	0.994	0.971	0.649
		400k	1.000	0.999	0.999	0.993	0.727
	+ CNB	100k	0.541	0.524	0.534	0.520	0.530
		400k	0.567	0.516	0.505	0.495	0.528
PixelDiT-B	baseline	100k	0.988	0.969	0.943	0.918	0.953
		400k	0.988	0.971	0.948	0.929	0.968
	+ CNB	100k	0.630	0.598	0.579	0.556	0.554
		400k	0.818	0.737	0.682	0.644	0.702

M Additional Qualitative Results

Fig. 18 and Fig. 19 give the failure direction for both architectures: paired cases where the baseline is correct and CNB is not, produced by the same script and the same selection rule as Figure 22. In the searches underlying these figures, disagreements ran 37/26 in CNB's favour on PixelDiT and 32/19 on SiT. Fig. 20 gives the SiT wins, the counterpart of Fig. 22 in the main text.

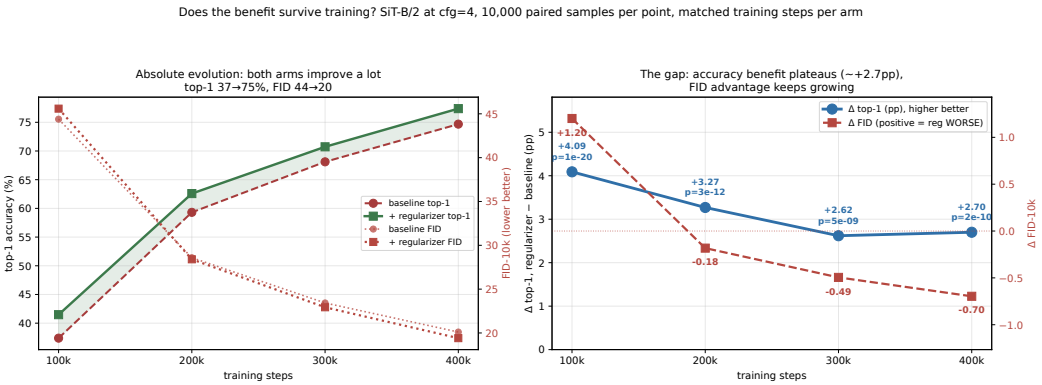

Figure 17: **Does the benefit survive training? SiT-B/2 at cfg=4.** Left: both arms improve substantially over training (top-1 37 $\rightarrow$ 75%, FID 44 $\rightarrow$ 20), which is the context for reading the gap. Right: the gap drops from its undertrained peak and then plateaus at $\sim +2.7$ pp, while ΔFID moves the other way.

Why small grids cannot resolve this effect. Figure 21 shows eight recognisable classes fixed before running, at a single seed, with no filtering. SiT comes out 7/8 for the baseline against 6/8 for CNB—the wrong way—while PixelDiT goes 4/8 to 7/8. At an effect of $+2.7$ pp one expects about 0.2 flips in eight samples, so sampling noise dominates entirely. We include this figure because the temptation to reseed until a small grid flatters the method is exactly the practice that makes qualitative figures untrustworthy. Note also that all four models label the requested tabby cat as "tiger cat", a neighbouring class—a systematic near-miss that FID cannot register.

N Relationship to Concurrent Work

Concurrent work (Turri et al., 2026) analyses *within-instance* channel criticality in diffusion conditioning—which channels of a single conditioning vector matter most for that instance. Our analysis is *across-instance*: how similar the conditioning vectors of *different* classes or prompts are to one another. The two are complementary and can disagree; a model whose conditioning vectors are nearly identical across classes may still have a small number of critical channels within each. Our diagnostic, our mechanisms and our intervention all concern the across-instance geometry.

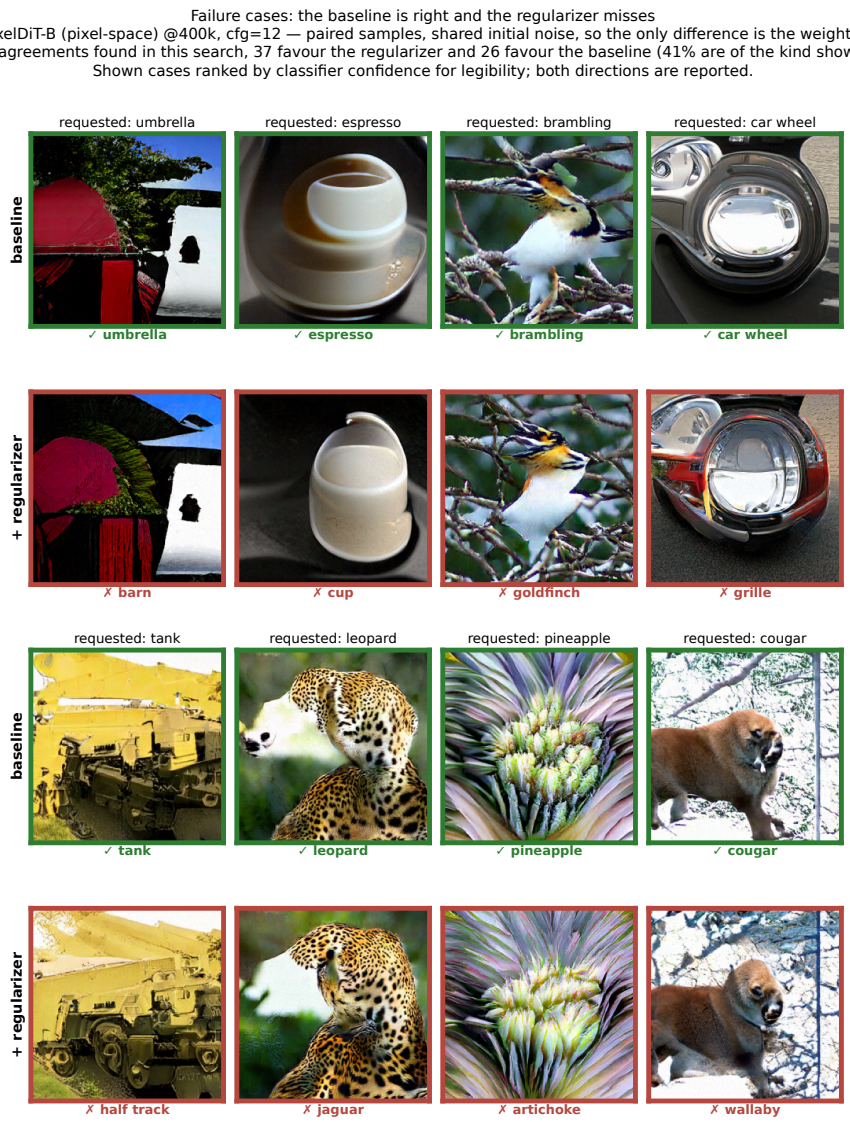

Figure 18: PixelDiT-B failure cases (26 of 63 disagreements).

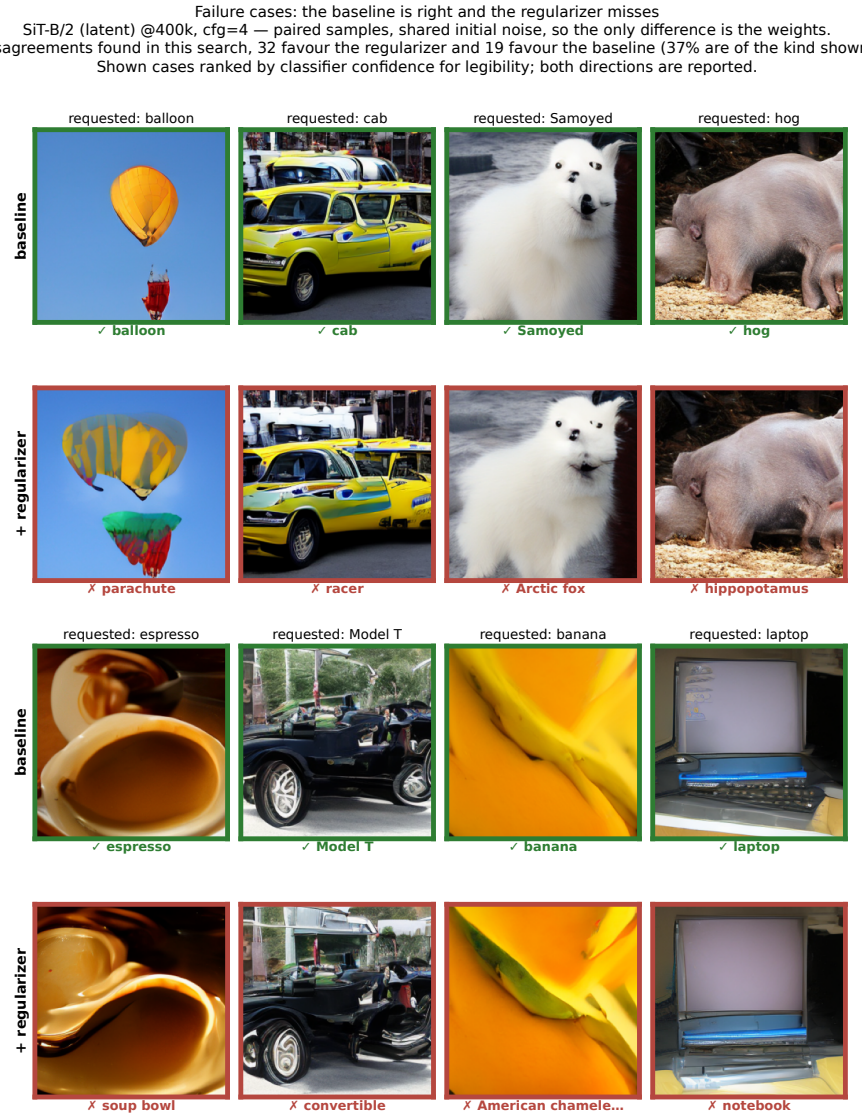

Figure 19: SiT-B/2 failure cases (19 of 51 disagreements).



Figure 20: SiT-B/2, cfg=4: cases CNB gets right and the baseline misses. Of all disagreements in this search, 32 favour CNB and 19 the baseline (63%).

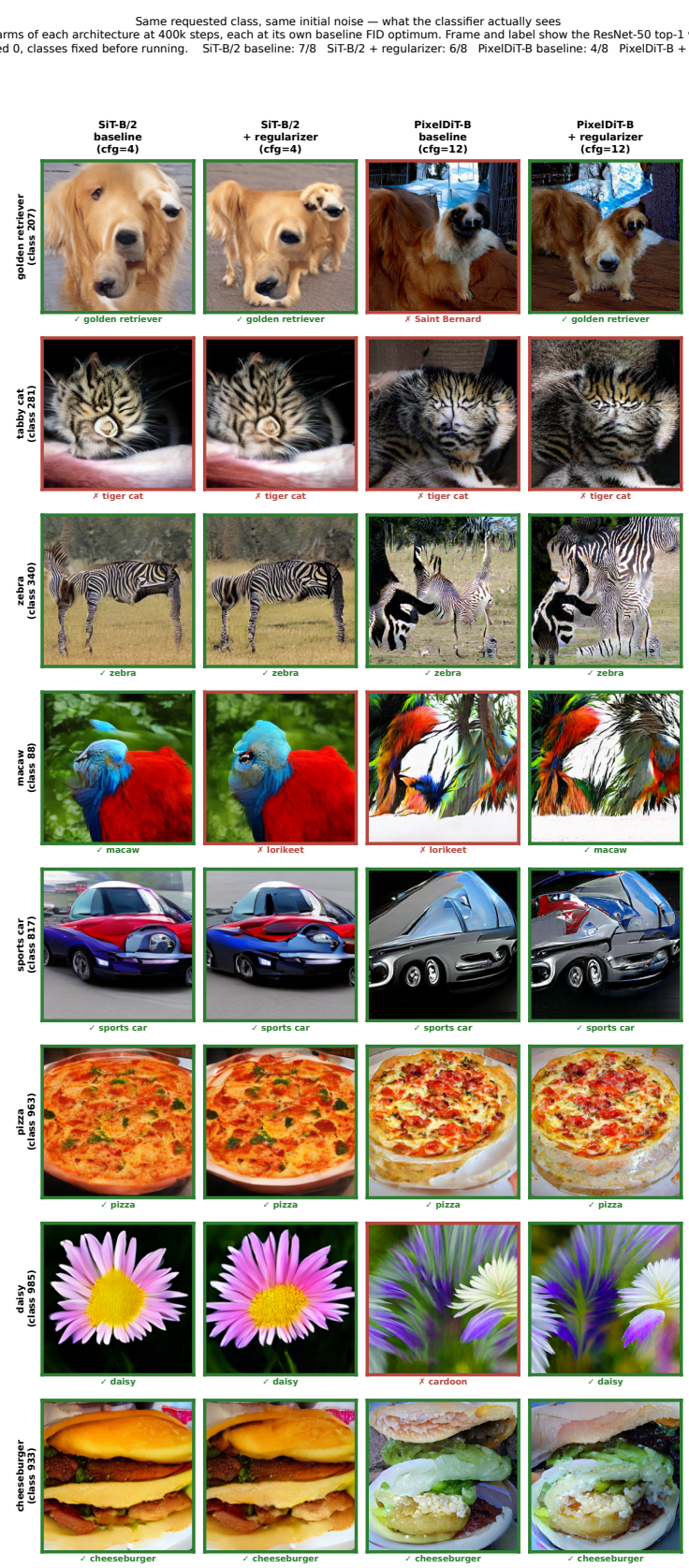

Figure 21: Unfiltered fixed-class grid, seed 0.



Figure 22: **PixelDiT-B, cfg=12: cases the regularizer gets right and the baseline misses.** Paired samples with shared initial noise. Of all disagreements in this search, 37 favour the regularizer and 26 the baseline (59%); the other direction is in Sec. M. Cases ranked by classifier confidence for legibility. Note that in several pairs the images are visually very similar—the intervention often changes something subtle rather than producing an obviously better picture.