BACKGROUND GRADIENTS SHAPE MEMORIZATION IN FLOW MATCHING

Anonymous authors
Paper under double-blind review

ABSTRACT

Repetition is closely associated with memorization in generative models, but how other training images affect the retention and copying of targets remains unclear. We study this question in class-conditioned flow matching, where images outside the target set form the background. At fixed target repetition and same-class background row count, replacing repeated same-class images with distinct images reduces the target extraction rate from 80.7% to 18.0%. To explain this effect, we develop a paired-trajectory framework that isolates target-induced parameter displacement and the background gradient response to it. This response has an exact path-integrated curvature representation, connecting background loss geometry to target learning. Reciprocal response transfer between repeated and distinct backgrounds changes target retention and copying in both directions, establishing the response's causal role. After target removal, the response correction parallel to the target-induced displacement preserves approximately 90% of the copying effects of full response transfer. Directly scaling the displacement also changes copying without further training. The post-removal copying effects of reciprocal transfer are reproduced across datasets and architectures. Together, these results identify the background gradient response as a mechanism through which same-class training data shape the retention of target learning and the reproduction of target images.

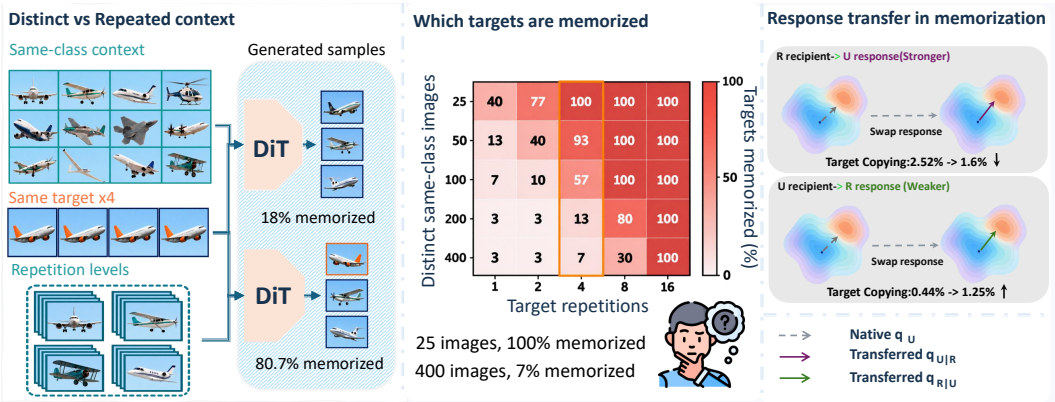


Figure 1: Data composition and target learning. Left: at 400 same-class background rows, repeated (R) and distinct (U) content give 80.7% and 18.0% target extraction, pooled over target repetition counts. Pictured target copies are schematic. Middle: balanced-design target extraction. Right: reciprocal response transfer and interleaved per-generation target copying.

1 INTRODUCTION

Diffusion and flow-matching models can reproduce training images verbatim, a phenomenon known as memorization (Carlini et al., 2023a, Somepalli et al., 2023a, Webster, 2023). Empirical studies consistently find that an image is more likely to be memorized the more often it is repeated (Carlini

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.

et al., 2023a, Somepalli et al., 2023b, Ma et al., 2025, Gu et al., 2025), and deduplication has accordingly become a standard defense (Webster, 2023, Kandpal et al., 2022). However, observed memorization varies widely across settings. Most images extracted in a Stable Diffusion study had at least a hundred near-duplicates (Carlini et al., 2023a), whereas fivefold duplication increased inversion-based memorization on CelebAHQ-2.5k (Ma et al., 2025). These settings differ in model, training procedure, and detection method. We therefore investigate whether increasing the number of distinct images from a target's class reduces its memorization when target repetition is held fixed.

To test this possibility, a balanced 75-run design crosses target repetition with the number of distinct same-class images while holding total training-set size fixed. Images outside the target set form the background. As shown in the middle panel of Figure 1, expanding the background from 25 to 400 distinct same-class images reduces the target extraction rate (the fraction of targets detected at least once) from 100% to 6.7% at four target repetitions. A second experiment separates the number of distinct same-class background images from their training-row count. The left panel of Figure 1 shows that, with that count fixed at 400, using 25 images repeated sixteen times yields 80.7% target extraction, whereas using 400 distinct images yields 18.0%. Repeating the same 25 background images instead of using each once changes extraction from 86.0% to 80.7%. The much larger change comes from replacing those images with distinct ones.

The mechanistic question is how training on these other images changes what the model retains from the targets. Answering it requires tracking the parameter changes induced by the targets, then testing how the background response to those changes affects retention and copying. To trace this composition effect through training, two trajectories start from the same model and optimizer state and use the same background data and stochastic inputs. Target updates are added to only one trajectory. The difference between their parameter states captures the displacement induced by the target set. The corresponding difference in background loss gradients defines the background gradient response. This response equals the background Hessian applied to the displacement, integrated along the line segment between the paired states. At a shared model state, the distinct background also has greater curvature along the initial target-gradient direction than the repeated background. The response measures background loss geometry along the displacement produced by the targets, complementing geometric accounts based on learned-manifold dimension and probability-landscape sharpness (Ross et al., 2025, Jeon et al., 2025).

Reciprocal response transfer tests whether this response affects target retention and copying. A common target-learning phase gives the repeated and distinct conditions the same initial target-specific benefit. During continued training, each recipient retains its own background gradient at the parameter state without target updates. The alternative background supplies a replacement response evaluated at the recipient's current pair of states. When targets continue to appear at the same training steps across conditions, transferring the distinct-background response to the repeated-background model lowers the target copying rate (the fraction of generated images detected as target hits) from 2.520% to 1.601%. The reverse transfer raises copying in the distinct-background model from 0.439% to 1.249%.

The transfer effects exceed matched random-direction and preclip gradient-norm controls. They also persist when target updates stop: reciprocal transfer changes both the retained target-specific benefit and target copying after target removal. The post-removal copying effects reproduce in both directions on CINIC-10 with DiT and CIFAR-10 with a U-Net. Across the four primary transfers, conditional accuracy changes by less than 0.7 percentage points. Applying only the response correction parallel to the target-induced displacement preserves 90.8% and 88.9% of the full post-removal copying effects. Directly scaling this displacement also changes copying without further training, connecting target-induced parameter changes to generated copies.

Our contributions to understanding memorization in class-conditioned flow matching are threefold:

1. We disentangle the roles of distinct background content and repeated training rows in memorization. Controlled data interventions show that distinct same-class images reduce target extraction at fixed target repetition and background row count.

2. We develop a paired-trajectory framework that connects background loss geometry to target learning. It isolates the background gradient response to target-induced parameter displacement and gives this response an exact path-integrated curvature representation.

3. We establish a causal role for the background gradient response in target retention and copying. Reciprocal transfer and response-component interventions identify its effect, while displacement scaling connects target-induced parameter changes to generated copies. Post-removal transfer effects reproduce across the tested datasets and architectures.

2 Related Work

Data Composition and Memorization. Supervised-learning studies distinguish fitting arbitrary labels from the role of rare examples in generalization (Zhang et al., 2017, Feldman, 2020, Feldman & Zhang, 2020). Diffusion-model studies link training-image replication to duplication (Carlini et al., 2023a, Somepalli et al., 2023b), dataset size and distribution (Gu et al., 2025), and local data coverage (Merger & Goldt, 2026). We isolate the effect of distinct same-class content at fixed target repetition and background row count by replacing repeated background images with distinct images, then investigate its training-time mechanism in class-conditioned flow matching.

Geometry and Training Dynamics of Memorization. Geometric analyses characterize memorization through mismatched learned and data-manifold dimensions (Ross et al., 2025) and probability-landscape sharpness (Jeon et al., 2025). In flow matching, the empirical optimal velocity field reproduces training samples, directing attention to how learned models depart from this optimum (Bertrand et al., 2025). Training-time studies identify separate timescales for generation and memorization (Bonnaire et al., 2025), learning-rate-induced regularization (Wu et al., 2025), and forgetting during subsequent training (Jagielski et al., 2023). Complementary analyses study the transition from memorization to generalization (Kadkhodaie et al., 2024, Buchanan et al., 2025). These accounts leave unresolved how background loss geometry acts on target-induced parameter changes. We identify this action through the finite background gradient response between paired target-on and target-off states, which equals the background Hessian action integrated along the segment connecting them.

Training-Data Influence and Causal Interventions. Training-data attribution estimates example-level effects through influence functions (Koh & Liang, 2017), checkpoint gradients (Pruthi et al., 2020), and approximate optimization unrolling (Bae et al., 2024). Datamodels and scalable attribution relate predictions to training-data choices (Ilyas et al., 2022, Park et al., 2023), with attribution methods also developed for diffusion models (Zheng et al., 2024, Mlodozeniec et al., 2025). Memorization interventions alter training through model aggregation and loss-based filtering (Liu et al., 2024), or modify trained models through neuron deactivation (Hintersdorf et al., 2024) and geometry-aware parameter sampling (Gegenfurtner et al., 2026). These methods do not test the background response as a separately transferable component of training. We isolate its effect on target retention and copying by reciprocally exchanging full responses evaluated at the recipient's current paired states while retaining its target-off raw background-gradient term. Appendix O develops the connections to data duplication, optimization geometry, and generative-model evaluation (Yin et al., 2026b,a).

3 Methodology

At fixed target repetition, we test how same-class composition changes memorization using paired trajectories, reciprocal response transfer, response-component interventions, and direct displacement scaling.

Let $\mathcal{T}$ denote the target set, with all other training images forming the background. Our primary experiments train a class-conditional DiT (Peebles & Xie, 2023) on 32×32 RGB CIFAR-10 images (Krizhevsky, 2009) with the flow-matching loss (Lipman et al., 2023):

$$x_t = (1-t)\epsilon + tx, \qquad \ell(\theta, x, c, t, \epsilon) = \frac{1}{d}\left\|v_\theta(x_t, t, c) - (x - \epsilon)\right\|_2^2. \tag{1}$$

Here v_θ is the class-conditioned velocity predictor with parameters θ, $x \in \mathbb{R}^d$ is an image with d scalar coordinates and label c, $\epsilon \sim \mathcal{N}(0, I_d)$, and $t \sim \mathrm{Uniform}[0, 1]$. The index s denotes optimization steps.

3.1 Controlled Data Interventions

For a target image $x^\star$ of class c, let $\mathcal{B}_c$ contain the other same-class images and $m(x)$ count occurrences of image x in the training set. We distinguish target repetition, distinct same-class image count, and other same-class training-row count as:

$$k = m(x^\star), \qquad N_{\mathrm{bg}} = |\mathcal{B}_c|, \qquad n_{\mathrm{bg}}^{\mathrm{row}} = \sum_{x \in \mathcal{B}_c} m(x). \tag{2}$$

The count k describes dataset multiplicity. Target draws depend on the training sampler.

A balanced cyclic design evaluates each target under every combination of k and N_{bg}, holding total target and background row counts fixed across models and nesting image sets within a common pool. We then fix both k and $n_{\mathrm{bg}}^{\mathrm{row}}$ to isolate the effect of distinct image content. Comparing 400 different other same-class images with 25 images repeated 16 times holds $n_{\mathrm{bg}}^{\mathrm{row}} = 400$ fixed. Comparing the latter with the same 25 images used once measures the effect of repeated examples. An additional control replaces these images with relabeled other-class images while retaining the conditioning label.

For the mechanism experiments, each panel contains ten target images, one per class. Repeated (R), intermediate (M), and distinct (U) backgrounds use nested sets of 25, 100, and 400 images per class, repeated 16, 4, and 1 times, respectively. All three conditions preserve labels and background row count. Their training objective is:

$$L_{\mathrm{mix}}^S(\theta) = w_B L_B^S(\theta) + w_T L_T(\theta), \qquad S \in \{R, M, U\}, \tag{3}$$

where L_B^S and L_T are mean losses on the background data and target set. The fixed weights w_B and w_T equal their respective proportions in the full training set. Stratified sampling preserves these weights, and removing the target loss leaves w_B unchanged. Numerical settings appear in Section 4.1.

3.2 Paired Training Trajectories and Background Gradient Responses

Figure 2 summarizes the design. A common target-learning phase begins from a shared target-free checkpoint and produces the initial target-specific benefit. The checkpoint supplies the model, optimizer, and random states for target-on $(+)$ and target-off $(-)$ branches, which share background data and stochastic inputs. Only the target-on branch receives the target loss. At the phase endpoint s_0, we clone the paired parameters, optimizer states, and random states across all backgrounds and intervention arms. Pulse-chase training then removes targets, whereas interleaved training continues them on a shared schedule with the same target sequence, flow times, and noise.

Let $g_B^S(\theta) = \nabla_\theta L_B^S(\theta)$ be the background loss gradient. The target-induced parameter displacement and its finite background gradient response are:

$$\delta\theta_{S,s} = \theta_{S,s}^+ - \theta_{S,s}^-, \qquad q_{S,s} = g_B^S(\theta_{S,s}^+) - g_B^S(\theta_{S,s}^-). \tag{4}$$

The displacement includes direct target updates and subsequent trajectory divergence, while the response measures its effect on the background gradient. Both gradients use identical background examples, labels, flow times, and noise.

For a twice continuously differentiable background loss, write $H_B^S(\theta) = \nabla_\theta^2 L_B^S(\theta)$. The exact path-integral response and, for $\delta\theta_{S,s} \neq 0$, its finite-response projection are:

$$q_{S,s} = \int_0^1 H_B^S(\theta_{S,s}^- + \tau\delta\theta_{S,s})\delta\theta_{S,s}\, d\tau, \qquad \kappa_{S,s} = \frac{\delta\theta_{S,s}^\top q_{S,s}}{\|\delta\theta_{S,s}\|_2^2}. \tag{5}$$

The projection equals the average directional curvature along the line segment connecting paired parameters. With ξ_s denoting shared realized batch inputs, Equations 4– 5 hold for the batch loss, gradient, and Hessian defined in Appendix A.1.

We additionally measure target-direction curvature at the shared target-free checkpoint $\bar{\theta}$:

$$g_T = \nabla_\theta L_T(\bar{\theta}), \qquad u = -\frac{g_T}{\|g_T\|_2}, \qquad h_S = u^\top H_B^S(\bar{\theta})u. \tag{6}$$

All backgrounds share $\bar{\theta}$, u, and the evaluation flow times and noise. Hessian vector products use the full flow-matching loss. The target-direction curvature h_S compares background losses at a fixed state and direction, whereas $\kappa_{S,s}$ measures the response along the realized displacement.

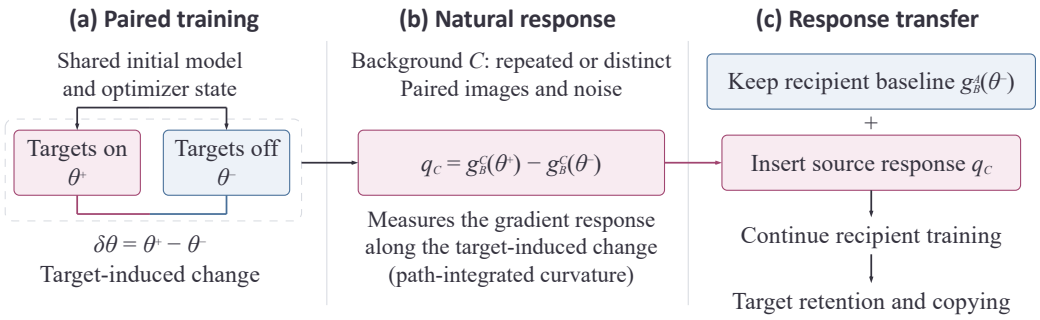


Figure 2: Paired training and reciprocal response transfer. Target-on and target-off states define the target-induced parameter displacement. At the current recipient pair, an alternative background supplies the response while the recipient retains its baseline background gradient. Continuing training measures the resulting target retention and copying.

3.3 Reciprocal Response Transfer

To test the background response's effect on retention, we replace it while retaining the recipient's baseline background gradient. Let $A \in \{R, U\}$ denote the recipient background and $C \in \{R, U\}$ the response source. At every step, we compute:

$$q_{C|A,s} = g_B^C(\theta_{A,s}^+) - g_B^C(\theta_{A,s}^-), \qquad \widetilde{g}_{A\leftarrow C,s}^+ = g_B^A(\theta_{A,s}^-) + q_{C|A,s}. \tag{7}$$

The source background loss is evaluated at the recipient's current paired parameters. The baseline background gradient $g_B^A(\theta_{A,s}^-)$ is also recomputed at the current target-off state. Setting $C = A$ recovers the native target-on background loss gradient. The two interventions $R \leftarrow U$ and $U \leftarrow R$ replace the full response vector. At the same recipient pair, the finite-response projection $\kappa_{C|A,s} = \delta\theta_{A,s}^\top q_{C|A,s}/\|\delta\theta_{A,s}\|_2^2$ is compared with the native projection $\kappa_{A|A,s}$. Section 4.1 specifies their aggregation.

With a_s indicating a scheduled target update, the complete target-on gradient before clipping is:

$$G_{A\leftarrow C,s}^+ = w_B \widetilde{g}_{A\leftarrow C,s}^+ + a_s w_T \nabla_\theta L_T(\theta_{A,s}^+). \tag{8}$$

For continued training after target removal, $a_s = 0$. Interleaved training retains the shared target schedule. The target-off branch uses $w_B g_B^A(\theta_{A,s}^-)$.

The random-direction control replaces the weighted correction $\Delta G_s = w_B(q_{C|A,s} - q_{A|A,s})$ with a per-step resampled Gaussian direction, norm-matched within each parameter tensor. The gradient-norm (preclip) control preserves the complete native gradient's direction while matching the transferred gradient's global norm before clipping. Both use their own current recipient pairs. After accumulation and unscaling, we construct and globally clip the complete gradient, then apply AdamW with branch-specific momentum and adaptive preconditioning (Appendix A.1).

Response-Component Interventions. We decompose the source-to-native response correction along the recipient's current target-induced displacement:

$$\begin{aligned} \Delta q_s &= q_{C|A,s} - q_{A|A,s}, \qquad d_s = \frac{\delta\theta_{A,s}}{\|\delta\theta_{A,s}\|_2}, \\ \Delta q_s^{\parallel} &= (d_s^\top \Delta q_s) d_s, \qquad \Delta q_s^{\perp} = \Delta q_s - \Delta q_s^{\parallel}. \end{aligned} \tag{9}$$

Here Δq_s is the response correction and d_s is the unit displacement direction for $\delta\theta_{A,s} \neq 0$. During post-removal training, we apply $w_B[g_B^A(\theta_{A,s}^+) + \Delta q_s^{\parallel}]$ or $w_B[g_B^A(\theta_{A,s}^+) + \Delta q_s^{\perp}]$, using the same clipping and AdamW rule as full transfer. Each arm recomputes direction and correction at its own current pair, retaining the complete native gradient. At a fixed pair, $\Delta q_s^{\parallel} = (\kappa_{C|A,s} - \kappa_{A|A,s})\delta\theta_{A,s}$. The parallel component isolates the finite-response projection change, and the orthogonal component has zero displacement projection. Finally, Section 4.5 scales the terminal native interleaved R displacement, which jointly reflects the panel's ten targets, without further training.

4 EXPERIMENTS AND RESULTS

4.1 EVALUATION AND STATISTICAL SETUP

The balanced design comprises 75 runs across three cycles, with 30 targets each evaluated under all 25 combinations of target repetition $k \in \{1, 2, 4, 8, 16\}$ and distinct background image count $N_{\mathrm{bg}} \in \{25, 50, 100, 200, 400\}$. Composition comparisons use 15 models across three blocks. Primary data evaluations use step-10,000 checkpoints and 2,000 generated images per class and model, with EMA parameters for the balanced design.

Primary response-transfer experiments use eight independent panels, each containing ten targets, 4,000 background training rows, and 160 target training rows. Following 2,500 target-free warmup steps and 400 common target-learning steps, paired continuations branch at $s_0 = 2,900$. We evaluate both transfer directions after target removal and under interleaved target training. At step 6,000, we evaluate raw parameters using 4,096 generated images per class. Training weights, sampling settings, and curvature-measurement details appear in Appendix A.1.

A generated image counts as a target hit when the target is its nearest reference within the conditioning class and their distance falls below the class-specific first percentile of holdout-to-reference distances. Extraction rate is the proportion of targets with at least one hit. Copying rate is the number of attributable target hits divided by all generations of the class. We use unnormalized pixel ℓ_2 distance as the primary measure and calibrate LPIPS-Alex (Zhang et al., 2018) separately. Conditional accuracy and KID (Bińkowski et al., 2018) accompany both protocols. Reference sets, threshold calibration, and untrained baselines are detailed in Appendix E.

To measure target-specific learning, we compare the paired target-on and target-off models on each target j and its untrained same-class decoys D_j, using matched flow times and noise:

$$\Delta L_{S,X}(s) = L_X(\theta^-_{S,s}) - L_X(\theta^+_{S,s}), \qquad X \in \{j, D_j\}, \tag{10}$$

where $\theta^+_{S,s}$ and $\theta^-_{S,s}$ are the paired parameters for continuation S, and L_{D_j} averages the decoy losses. Subtracting the decoy improvement gives the target-specific benefit:

$$B^{\mathrm{spec}}_{S,j}(s) = \Delta L_{S,j}(s) - \Delta L_{S,D_j}(s), \qquad \mathcal{R}_{S,j}(s) = \frac{B^{\mathrm{spec}}_{S,j}(s)}{B^{\mathrm{spec}}_{j}(s_0)}. \tag{11}$$

Each target's benefit at s_0 provides a fixed reference shared by all continuations, so every continuation starts at $\mathcal{R}_{S,j}(s_0) = 1$. We compute this ratio for each target before averaging within panels. After target removal, $\mathcal{R}$ measures how much of the initial target-specific benefit is retained. Under interleaved training, it reflects both retention and continued target learning.

Statistical inference for mechanism experiments uses independent panels. We first average outcomes across classes with equal weights within each panel, then compute 95% t confidence intervals across panels, using paired differences for intervention comparisons. Within each training regime, Holm correction applies to four primary copying comparisons: each of the two transfer directions versus native training and versus its random-direction control. Transfer fractions express mean copying effects relative to the mean native R-to-U copying gap in the same regime, with paired-panel bootstrap intervals. Projection averaging, additional inference procedures, and data-design details appear in Appendices A.1, A.2, and H.

4.2 DISTINCT SAME-CLASS IMAGES REDUCE TARGET EXTRACTION RATE

Every target appears under all 25 balanced-design combinations. Each model contains 1,550 background and 62 target examples. At $k = 4$, increasing N_{bg} from 25 to 400 reduces target extraction rate from 100% to 6.7%, as shown in Figure 1.

Figure 3a shows that, at fixed target repetition and background row count, replacing 25 images repeated 16 times with 400 distinct images reduces extraction from 80.7% to 18.0%. Repeating the same 25 images instead of using them once changes extraction from 86.0% to 80.7%. Distinct content produces a 62.7 percentage-point change, versus 5.3 points from repeating the same images.

At 20,000 steps, extraction remains lower with distinct images (30.0% versus 78.0%), with conditional accuracies of 80.0% and 94.0%. Figure 3b shows the distinct condition's 56.6% accuracy at step

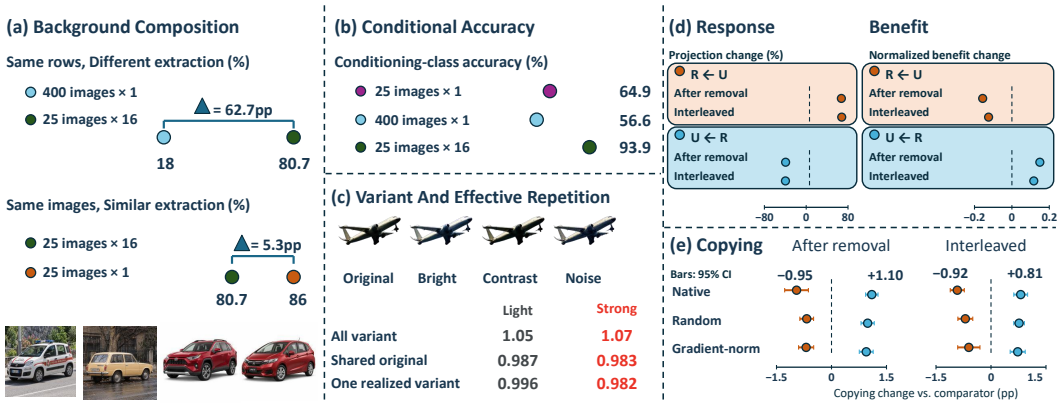


Figure 3: Distinct background content reduces target extraction, and reciprocal response transfer changes target-specific benefit and copying. (a) Extraction under matched row or distinct-image counts. (b) Corresponding conditional accuracy. (c) Effective repetition of photometric variants relative to exact copies under three detection-reference policies. (d) Finite-response projection changes (%) and normalized-benefit differences from native values. (e) Copying contrasts (transfer minus comparator, percentage points), with paired 95% confidence intervals across eight independent panels. In (d) and (e), orange denotes $R \leftarrow U$ and blue denotes $U \leftarrow R$ after target removal or during interleaved training, where R and U use repeated and distinct backgrounds, respectively.

10,000. Its accuracy improves with longer training while a 48.0-point extraction gap persists, as detailed in Table 27.

Figure 3c shows that brightness, contrast, and pixel-noise variants retain approximately the repetition strength of exact copies when each target has one detection reference. Appendix G specifies the photometric and reference-selection protocols.

4.3 Background Composition Changes the Retention of Target Learning

All continuations share the same initial target-specific benefit at $s_0 = 2,900$, with $\mathcal{R}(s_0) = 1$. After target removal, distinct backgrounds retain less of this benefit and produce fewer target copies: terminal normalized benefit decreases from 0.3517 under R to 0.1428 under U, while copying decreases from 2.685% to 0.523%.

As shown in Table 1, both normalized target-specific benefit and copying decrease across R, M, and U under the shared interleaved target schedule. The R–U copying gap is 2.081 percentage points (95% interval $[1.789, 2.372]$).

These backgrounds also differ in their target-direction curvature at the common target-free checkpoint. Table 1 shows increasing curvature across R, M, and U at fixed parameters and target direction, opposite to the ordering of subsequent benefit and copying. The U–R curvature difference is 0.02665 (95% interval $[0.02581, 0.02748]$). These geometric differences at a shared state motivate testing the background gradient response through reciprocal transfer in Section 4.4.

Table 1: Native backgrounds across eight experimental panels. Target-direction curvature h_S uses the shared step-2,500 checkpoint. Normalized target-specific benefit and target copying rate use interleaved training at step 6,000. Brackets give 95% panel intervals.

Background	h_S	$\mathcal{R}$	Target copying rate (%)
R, 25×16	0.03834	0.3206	2.520 [2.197, 2.843]
M, 100×4	0.05200	0.2197	1.177 [0.978, 1.376]
U, 400×1	0.06499	0.1305	0.439 [0.372, 0.506]

Table 2: Response-component interventions after target removal. Copying rates are percentages. Effects are native minus intervention for R and intervention minus native for U, in percentage points. Brackets are ordinary 95% paired t intervals from the eight-panel component experiment.

	$R \leftarrow U$		$U \leftarrow R$	
Correction	Copying	Effect [95% CI]	Copying	Effect [95% CI]
Native	2.685	n/a	0.523	n/a
Full	1.732	0.953 [0.912, 0.994]	1.622	1.099 [1.042, 1.156]
Parallel only	1.820	0.865 [0.832, 0.898]	1.500	0.977 [0.928, 1.026]
Orthogonal only	2.570	0.115 [0.107, 0.123]	0.640	0.117 [0.109, 0.125]

4.4 Reciprocal Response Transfer Changes Retention and Target Copying

Figure 3(d) and (e) shows that transferring the U response to R lowers target-specific benefit and copying, whereas transferring the R response to U raises both outcomes.

After target removal, transferring the U response to R increases the recipient's finite-response projection by 67.3%, lowers normalized target-specific benefit from 0.3517 to 0.1954, and reduces copying from 2.685% to 1.732%. The reciprocal transfer decreases the recipient U projection by 39.4%, raises normalized benefit from 0.1428 to 0.2922, and increases copying from 0.523% to 1.622%.

The same directional effects hold under interleaved target training. Transferring the U response to R increases the projection by 68.0%, lowers normalized benefit from 0.3206 to 0.1959, and reduces copying from 2.520% to 1.601%. Transferring the R response to U decreases the projection by 39.7%, raises normalized benefit from 0.1305 to 0.2475, and increases copying from 0.439% to 1.249%. These copying shifts amount to 44.2% and 38.9% of the native R–U copying gap, respectively.

As shown in Figure 3(e), the copying effects in both directions exceed those of the random-direction control and the gradient-norm control, which matches the transferred gradient's norm before clipping while preserving the native gradient direction. Under interleaved training, $R \leftarrow U$ reduces copying by an additional 0.700 percentage points relative to its random-direction control (95% interval $[0.497, 0.903]$), while $U \leftarrow R$ increases it by an additional 0.766 points ($[0.620, 0.912]$). The four primary copying comparisons in each regime have Holm-adjusted $p \leq 2.18 \times 10^{-4}$. Conditional accuracy changes across the four primary transfers range from -0.524 to $+0.688$ percentage points.

Parallel-Only Response Correction Preserves Most of the Copying Effect. Table 2 compares the full, parallel-only, and orthogonal-only corrections defined in Equation 9 after target removal. The correction parallel to the target-induced displacement reduces copying in R by 0.865 percentage points and increases it in U by 0.977 points. The orthogonal correction changes copying in the same directions by 0.115 and 0.117 points. The parallel effects exceed the orthogonal effects by 0.750 points (95% interval $[0.725, 0.775]$) in R and 0.860 points ($[0.819, 0.901]$) in U across eight paired panels. Parallel-only correction preserves 90.8% and 88.9% of the full-transfer copying effects, respectively, locating most of the effect along the target-induced displacement.

4.5 Target-Induced Parameter Displacement Controls Generated Copying

We directly scale the joint ten-target displacement from the terminal native interleaved R pair without further training. At step 6,000, we evaluate:

$$\theta(\gamma) = \theta^{-}_{R,6000} + \gamma \delta\theta_{R,6000}, \qquad \gamma \in \{0, 0.5, 1, 1.5\}. \tag{12}$$

The endpoints $\gamma = 0$ and 1 recover the target-off and target-on states. Within each of the eight panels, scales share generation inputs and detection settings. An equal-norm random parameter direction and a non-target learning direction provide the controls reported in Table 3.

Copying rises from 0.33% to 1.30%, 2.520%, and 3.71% across the four scales, increasing in 23 of 24 adjacent-scale comparisons within panels. Target-specific benefit rises from zero to 0.172. The random and non-target directions yield 0.38% and 0.44% copying, respectively. Decoy hits and non-target copying remain within 0.30–0.31% and 0.45–0.47%. From $\gamma = 1$ to 1.5, conditional

accuracy changes by -0.4 percentage points and KID by $+0.0011$. These state interventions connect the joint target-induced displacement to generated copies, complementing the response interventions that alter target learning during training. Further implementation details appear in Appendix A.6.

Table 3: State interventions on eight native interleaved R pairs at step 6,000. Entries are panel means, and rates are percentages. B^{spec} uses a common target-off reference. The random direction matches the $\gamma = 1$ displacement norm.

State or direction	Target copy	B^{spec}	Decoy hit	Non-target copy
$\gamma = 0$	0.33	0	0.30	0.45
$\gamma = 0.5$	1.30	0.071	0.30	0.45
$\gamma = 1$	2.520	0.1284	0.30	0.46
$\gamma = 1.5$	3.71	0.172	0.31	0.47
Random parameter direction	0.38	0.004	0.30	0.45
Non-target learning direction	0.44	0.006	0.31	0.46

4.6 Post-Removal Response Transfer across Datasets and Architectures

Table 4 shows both post-removal copying effects on CINIC-10/DiT and CIFAR-10/U-Net, each with six independent panels, matched background row counts, and random-direction controls. $R \leftarrow U$ lowers copying and $U \leftarrow R$ raises it in both settings. All four effects exceed their matched-random controls, with paired 95% intervals excluding zero. Relative to the primary CIFAR-10/DiT experiments, these settings vary the dataset or architecture separately, with targets absent throughout continuation. Appendix A.4 provides further training details.

Table 4: Post-removal changes in target copying (percentage points). Values are transfer minus comparator, with paired 95% confidence intervals across panels ($n = 8$ for CIFAR-10/DiT and $n = 6$ otherwise). Random denotes the matched random-direction control.

Setting	Comparator	$R \leftarrow U$		$U \leftarrow R$	
		Change	95% CI	Change	95% CI
CIFAR-10 / DiT	Native	$\mathbf{-0.953}$	$[-1.277, -0.630]$	$\mathbf{+1.099}$	$[0.929, 1.269]$
	Random	$\mathbf{-0.676}$	$[-0.863, -0.489]$	$\mathbf{+0.986}$	$[0.805, 1.167]$
CINIC-10 / DiT	Native	$\mathbf{-0.682}$	$[-0.778, -0.586]$	$\mathbf{+0.711}$	$[0.558, 0.864]$
	Random	$\mathbf{-0.584}$	$[-0.716, -0.453]$	$\mathbf{+0.615}$	$[0.485, 0.744]$
CIFAR-10 / U-Net	Native	$\mathbf{-0.568}$	$[-0.722, -0.414]$	$\mathbf{+0.568}$	$[0.494, 0.642]$
	Random	$\mathbf{-0.488}$	$[-0.590, -0.386]$	$\mathbf{+0.503}$	$[0.431, 0.574]$

5 Conclusion

We identify the background gradient response to target learning as a mechanism shaping memorization in class-conditioned flow matching. The paired-trajectory framework connects this response to background loss geometry. Reciprocal transfer establishes its causal role in target retention and copying. After target removal, the displacement-aligned correction preserves most of the effect of full transfer on copying. Scaling the target-induced displacement also changes copying without further training. Together, these findings connect background loss geometry to the retention of target learning and the reproduction of target examples.

Our mechanism experiments establish a causal role for the background response to jointly introduced targets in low-resolution, class-conditioned flow models. Isolating the response induced by individual targets would allow us to examine why the same background affects their retention differently at matched target exposure. Building on this analysis, future work could investigate which features of the relationship between targets and their background predict these retention differences, and whether those predictions also explain differences in generated copying.

AI USE STATEMENT

AI tools assisted with manuscript drafting and editing, literature checks.

ETHICS STATEMENT

This work investigates memorization through controlled experiments on CIFAR-10 and CINIC-10. The findings can inform privacy auditing, but may also facilitate training-data extraction. The interventions are intended for studying and evaluating memorization. Their use should respect data-use restrictions and avoid unauthorized extraction or redistribution of training content.

REPRODUCIBILITY STATEMENT

Section 3 defines the data interventions, paired response, transfer operation, and controls. Section 4.1 specifies the copying endpoints, target-specific benefit, and panel-level inference. Appendix A reports the complete eight-panel transfer and quality summaries. Appendix A.1 specifies the shared-start training schedule, gradient controls, curvature measurement, and benefit normalization. The remaining supplement gives balanced assignment rules, near-duplicate reference policies, and the earlier controlled-geometry experiments. Section 4.5 reports the displacement-scaling results, with further implementation details in Appendix A.6. Appendix N.5 gives panel-level copying inputs for the primary pulse-chase and response-component comparisons. The supplementary material includes a code package for recomputing these copying statistics, with reference implementations of the core intervention operations and a step-by-step reproducibility checklist.

REFERENCES

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

Juhan Bae, Wu Lin, Jonathan Lorraine, and Roger Grosse. Training data attribution via approximate unrolling. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 66647–66686, 2024.

Quentin Bertrand, Anne Gagneux, Mathurin Massias, and Rémi Emonet. On the closed-form of flow matching: Generalization does not arise from target stochasticity. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 8522–8549, 2025.

Mikołaj Bińkowski, Danica J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations (ICLR)*, 2018.

Tony Bonnaire, Raphaël Urfin, Giulio Biroli, and Marc Mézard. Why diffusion models don't memorize: The role of implicit dynamical regularization in training. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, 2025. doi: 10.52202/085713-4720.

Sam Buchanan, Druv Pai, Yi Ma, and Valentin De Bortoli. On the edge of memorization in diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, 2025. doi: 10.52202/085713-3213.

A. Colin Cameron, Jonah B. Gelbach, and Douglas L. Miller. Robust inference with multiway clustering. *Journal of Business & Economic Statistics*, 29(2):238–249, 2011.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium (USENIX Security)*, pp. 267–284. USENIX Association, 2019.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security)*, pp. 2633–2650. USENIX Association, 2021.

Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security)*, pp. 5253–5270. USENIX Association, 2023a.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramèr, and Chiyuan Zhang. Quantifying memorization across neural language models. In *International Conference on Learning Representations (ICLR)*, 2023b.

Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-SGD: Biasing gradient descent into wide valleys. In *International Conference on Learning Representations (ICLR)*, 2017.

Jeremy M. Cohen, Simran Kaur, Yuanzhi Li, J. Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations (ICLR)*, 2021.

Luke N. Darlow, Elliot J. Crowley, Antreas Antoniou, and Amos J. Storkey. CINIC-10 is not ImageNet or CIFAR-10. Technical Report EDI-INF-ANC-1802, University of Edinburgh, 2018.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In *International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1019–1028. PMLR, 2017.

Vitaly Feldman. Does learning require memorization? A short tale about a long tail. In *Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pp. 954–959. Association for Computing Machinery, 2020. doi: 10.1145/3357713.3384290.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 2881–2891, 2020.

E. C. Fieller. Some problems in interval estimation. *Journal of the Royal Statistical Society: Series B (Methodological)*, 16(2):175–185, 1954.

David Firth. Bias reduction of maximum likelihood estimates. *Biometrika*, 80(1):27–38, 1993.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations (ICLR)*, 2021.

Johanna Marie Gegenfurtner, Albert Kjøller Jacobsen, Naima Elosegui Borras, Alejandro Valverde Mahou, and Georgios Arvanitidis. Reducing memorisation in generative models via Riemannian Bayesian inference. *arXiv preprint arXiv:2602.00199*, 2026.

Amirata Ghorbani and James Zou. Data Shapley: Equitable valuation of data for machine learning. In *International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2242–2251. PMLR, 2019.

Behrooz Ghorbani, Shankar Krishnan, and Ying Xiao. An investigation into neural net optimization via Hessian eigenvalue density. In *International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2232–2241. PMLR, 2019.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012.

Xiangming Gu, Chao Du, Tianyu Pang, Chongxuan Li, Min Lin, and Ye Wang. On memorization in diffusion models. *Transactions on Machine Learning Research*, 2025.

Moritz Hardt, Benjamin Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning (ICML)*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1225–1234. PMLR, 2016.

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)*, volume 30, 2017.

Dominik Hintersdorf, Lukas Struppek, Kristian Kersting, Adam Dziedzic, and Franziska Boenisch. Finding NeMo: Localizing neurons responsible for memorization in diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 88236–88278, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 6840–6851, 2020.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Understanding predictions with data and data with predictions. In *International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9525–9587. PMLR, 2022.

Matthew Jagielski, Om Thakkar, Florian Tramèr, Daphne Ippolito, Katherine Lee, Nicholas Carlini, Eric Wallace, Shuang Song, Abhradeep Thakurta, Nicolas Papernot, and Chiyuan Zhang. Measuring forgetting of memorized training examples. In *International Conference on Learning Representations (ICLR)*, 2023.

Dongjae Jeon, Dueun Kim, and Albert No. Understanding and mitigating memorization in generative models via sharpness of probability landscapes. In *International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 27091–27112. PMLR, 2025.

Zahra Kadkhodaie, Florentin Guth, Eero P. Simoncelli, and Stéphane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representations. In *International Conference on Learning Representations (ICLR)*, pp. 46543–46567, 2024.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 10697–10707. PMLR, 2022.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 26565–26577, 2022.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations (ICLR)*, 2017.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1885–1894. PMLR, 2017.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In *60th Annual Meeting of the Association for Computational Linguistics (ACL), Volume 1: Long Papers*, pp. 8424–8445. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-long.577.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

Xiao Liu, Xiaoliu Guan, Yu Wu, and Jiaxu Miao. Iterative ensemble training with anti-gradient control for mitigating memorization in diffusion models. In *European Conference on Computer Vision (ECCV)*, volume 15145 of *Lecture Notes in Computer Science*, pp. 108–123, 2024.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Zhe Ma, Qingming Li, Xuhong Zhang, Tianyu Du, Ruixiao Lin, Zonghui Wang, Shouling Ji, and Wenzhi Chen. An inversion-based measure of memorization for diffusion models. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16959–16969, 2025.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

Claudia Merger and Sebastian Goldt. Local coverage governs memorization in diffusion models. *arXiv preprint arXiv:2606.14390*, 2026.

Bruno Mlodozeniec, Runa Eschenhagen, Juhan Bae, Alexander Immer, David Krueger, and Richard E. Turner. Influence functions for scalable data attribution in diffusion models. In *International Conference on Learning Representations (ICLR)*, pp. 51728–51764, 2025.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. TRAK: Attributing model behavior at scale. In *International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 27074–27113. PMLR, 2023.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, 2023.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 19920–19930, 2020.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *Lecture Notes in Computer Science*, pp. 234–241, 2015.

Brendan Leigh Ross, Hamidreza Kamkari, Tongzi Wu, Rasa Hosseinzadeh, Zhaoyan Liu, George Stein, Jesse C. Cresswell, and Gabriel Loaiza-Ganem. A geometric framework for understanding memorization in generative models. In *International Conference on Learning Representations (ICLR)*, pp. 34773–34805, 2025.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? Investigating data replication in diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6048–6058, 2023a.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Understanding and mitigating copying in diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 47783–47803, 2023b. doi: 10.52202/075280-2071.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*, 2021a.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2021b.

Kushal Tirumala, Aram H. Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. Memorization without overfitting: Analyzing the training dynamics of large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 38274–38290, 2022. doi: 10.52202/068431-2773.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024.

Gerrit J. J. van den Burg and Christopher K. I. Williams. On memorization in probabilistic deep generative models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 27916–27928, 2021.

Ryan Webster. A reproducible extraction of training images from diffusion models. *arXiv preprint arXiv:2305.08694*, 2023.

Yuxin Wen, Yuchen Liu, Chen Chen, and Lingjuan Lyu. Detecting, explaining, and mitigating memorization in diffusion models. In *International Conference on Learning Representations (ICLR)*, pp. 35647–35662, 2024.

Yu-Han Wu, Pierre Marion, Gérard Biau, and Claire Boyer. Taking a big step: Large learning rates in denoising score matching prevent memorization. In *Conference on Learning Theory (COLT)*, volume 291 of *Proceedings of Machine Learning Research*, pp. 5718–5756, 2025.

Zeqi Ye, Qijie Zhu, Molei Tao, and Minshuo Chen. Provable separations between memorization and generalization in diffusion models. In *International Conference on Learning Representations (ICLR)*, pp. 143375–143425, 2026.

Xuanhua Yin, Shunqi Mao, Wei Guo, Chuanzhi Xu, and Weidong Cai. Calibrate what you SHIP: Post-selection risk control for verifier-guided text-to-image generation. *arXiv preprint arXiv:2608.21748*, 2026a.

Xuanhua Yin, Chuanzhi Xu, Shunqi Mao, Wei Guo, and Weidong Cai. DefaultShift: Auditing semantic default shift in accelerated text-to-image models. *arXiv preprint arXiv:2608.21784*, 2026b.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations (ICLR)*, 2017.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, 2018.

Zekai Zhang, Xiao Li, Xiang Li, Lianghe Shi, Meng Wu, Molei Tao, and Qing Qu. Generalization of diffusion models arises with a balanced representation space. In *International Conference on Learning Representations (ICLR)*, pp. 101444–101475, 2026.

Xiaosen Zheng, Tianyu Pang, Chao Du, Jing Jiang, and Min Lin. Intriguing properties of data attribution on diffusion models. In *International Conference on Learning Representations (ICLR)*, pp. 18417–18452, 2024.

APPENDIX

A BACKGROUND RESPONSE, TARGET BENEFIT, AND GENERATION QUALITY

The complete paired transfer comparisons report finite-response projections, normalized target-specific benefit, copying, conditional accuracy, and KID. Quality changes use the same receiver's native condition as the reference. At the shared step-2,500 checkpoint, native directional Hessian values were 0.03834 [0.03784, 0.03884] for R, 0.05200 [0.05139, 0.05261] for M, and 0.06499 [0.06402, 0.06596] for U. Brackets give 95% panel intervals. Table 8 separately reports the panel-mean relative change in the finite-response projection, comparing source and native background responses at the same recipient state pair. This diagnostic uses the realized target-induced displacement, whereas h_S uses the fixed direction u at the common initial checkpoint. In the tables below, $\mathcal{R}$ measures retention in pulse-chase and combines continued learning and retention in interleaved training.

Table 5: Signed target-specific benefit and its normalized counterpart at step 6,000. The native and transferred arms use eight paired panels. Each normalizer is fixed before continuation and shared across conditions for the same panel-target. All four transfer comparisons agree in direction between B^{spec} and $\mathcal{R}$ in all eight panels.

Regime	Condition	B^{spec}	$\mathcal{R}$
Interleaved	R native	0.1284	0.3206
	M native	0.0879	0.2197
	U native	0.0522	0.1305
	R receiving U response	0.0784	0.1959
	U receiving R response	0.0987	0.2475
Pulse-chase	R native	0.1412	0.3517
	R receiving U response	0.0782	0.1954
	U native	0.0571	0.1428
	U receiving R response	0.1171	0.2922

Table 6: Native interleaved generation quality and LPIPS copying at step 6,000. Rates are percentages. KID uses unscaled Inception pool3 features. The R row also describes the shared $\gamma = 1$ dose endpoint.

Background	LPIPS copy	Accuracy	KID
R	3.403	71.299	0.04121
M	1.595	70.636	0.03899
U	0.596	69.719	0.03741

Inference Protocol. Each of the eight data panels is one independent unit. Classes are weighted equally within a panel, while targets, generation banks, and generated draws are treated as repeated measurements. Reported 95% intervals are ordinary t intervals over the eight panel summaries. Contrast intervals use paired panel differences. Within each regime, the four primary copying contrasts (both transfer directions against native training and against the corresponding matched-random control) form one Holm-adjusted family for decisions. Transfer fractions divide the panel-mean intervention effect by the native R-U copy gap in that regime and use paired panel bootstrap intervals.

A.1 SHARED-START MECHANISM PROTOCOL AND CONTROLS

Each panel uses an independent initialization seed and a fixed non-target background pool for warmup through step 2,500. At this checkpoint, model, optimizer, and random-number-generator states are cloned into target-on and target-off branches. Common target learning occupies steps 2,500–2,900: background data and stochastic inputs are paired, and only the on branch adds the target loss. At $s_0 = 2{,}900$, both states, their optimizer states, and their random-number-generator states are saved and cloned across R, M, U, and the intervention continuations through step 6,000. Pulse-chase continuations stop target training. Interleaved continuations retain the shared target schedule. Each

Table 7: Reciprocal response transfer and controls with full response replacement and eight paired panels per regime. The receiver supplies the training background, and the source supplies the response. Copy rates are percentages with 95% panel intervals. Accuracy changes (pp) and KID changes are relative to the same receiver's native condition. The n/a entries mark native reference rows, for which changes are not reported. Random and norm controls use the corresponding transfer's matching constraints.

Receiver	Intervention	$\mathcal{R}$	Pixel copy [95% CI]	ΔAcc.	ΔKID
Pulse-chase					
R	Native	0.3517	2.685 [2.388, 2.981]	n/a	n/a
R	U response	0.1954	1.732 [1.502, 1.961]	-0.341	-0.00128
R	Matched random	0.3146	2.408 [2.220, 2.596]	-0.068	-0.00054
R	Preclip norm	0.3157	2.424 [2.105, 2.743]	-0.063	-0.00038
U	Native	0.1428	0.523 [0.446, 0.600]	n/a	n/a
U	R response	0.2922	1.622 [1.404, 1.840]	$+0.688$	$+0.00172$
U	Matched random	0.1539	0.636 [0.547, 0.724]	$+0.009$	-0.00010
U	Preclip norm	0.1669	0.674 [0.539, 0.809]	$+0.159$	$+0.00036$
Interleaved					
R	Native	0.3206	2.520 [2.197, 2.843]	n/a	n/a
R	U response	0.1959	1.601 [1.323, 1.878]	-0.524	-0.00150
R	Matched random	0.3082	2.301 [1.915, 2.686]	-0.130	-0.00033
R	Preclip norm	0.2943	2.205 [1.839, 2.571]	-0.096	-0.00025
U	Native	0.1305	0.439 [0.372, 0.506]	n/a	n/a
U	R response	0.2475	1.249 [1.037, 1.460]	$+0.530$	$+0.00153$
U	Matched random	0.1386	0.482 [0.375, 0.590]	$+0.245$	$+0.00018$
U	Preclip norm	0.1425	0.520 [0.447, 0.593]	$+0.118$	$+0.00023$

Table 8: Finite-response projection changes (%) under reciprocal transfer. Source and native projections are measured at matched recipient states, then averaged over four times and four batches per time within each panel before taking their ratio (Appendix A.1). Entries give means and 95% intervals across eight panels. Shared-checkpoint curvature h_S is reported separately in Table 1.

Receiver	Intervention	Pulse-chase	Interleaved
R	U response	$+67.3$ [$+65.5$, $+69.0$]	$+68.0$ [$+65.3$, $+70.7$]
U	R response	-39.4 [-41.2, -37.6]	-39.7 [-40.9, -38.5]

continuation subsequently maintains its own optimizer state. Throughout, $w_B = 4000/4160$ and $w_T = 160/4160$ remain explicit. Omitting the target term does not renormalize w_B. Stratified batches do not replace these weights with sampling proportions. This protocol tests the fate of a commonly acquired target benefit.

During common target learning and interleaved continuation, each stratified batch contains 240 background draws and 16 target draws. The shared target sampler seed gives the same target-ID sequence in every paired condition. Table 9 reports actual per-target exposure rather than exposure inferred from the dataset repetition count k, which denotes the number of copied dataset rows. Conditions sharing a schedule have zero difference in these counts. In pulse-chase, the target-update indicator satisfies $a_s = 0$ at every step from 2,901 through 6,000, yielding zero target-training updates.

Table 9: Training schedule and actual target exposure. Counts are per target in the target-on branch, whereas target-off branches omit the target loss.

Phase	Step interval	Updates	Target draws
Common target learning	2,500–2,900	400	640
Interleaved continuation	2,900–6,000	3,100	4,960
Pulse-chase continuation	2,900–6,000	3,100	0

For realized-batch notation, let ξ_s collect the batch draws, flow times, and noise at step s, and let $\widehat{L}^S_{B,s}(\theta, \xi_s)$ be the corresponding background loss. Its gradient and Hessian are $\widehat{g}^S_{B,s} = \nabla_\theta \widehat{L}^S_{B,s}$ and $\widehat{H}^S_{B,s} = \nabla^2_\theta \widehat{L}^S_{B,s}$. The matched response is $\widehat{q}_{S,s} = \widehat{g}^S_{B,s}(\theta^+_{S,s}, \xi_s) - \widehat{g}^S_{B,s}(\theta^-_{S,s}, \xi_s)$. Because both states use the same ξ_s, the finite-difference definition and path-integral identity in Equations 4–5 hold for these batch quantities after adding hats. The main text suppresses hats and ξ_s.

For each transfer or control branch, all gradient quantities are recomputed at its current recipient pair using the same realized batch draws, flow times, and noise at both states. The complete gradients before clipping and the weighted transfer correction are:

$$G_{\mathrm{nat}} = w_B g^A_B(\theta^+_{A,s}) + a_s w_T \nabla_\theta L_T(\theta^+_{A,s}), \tag{13}$$

$$\Delta G = w_B\big(q_{C|A,s} - q_{A|A,s}\big), \qquad G_{\mathrm{tr}} = G_{\mathrm{nat}} + \Delta G, \tag{14}$$

where G_{nat} and G_{tr} are the complete native and transferred on-branch gradients, ΔG is their difference, a_s indicates whether a target update is scheduled, A is the recipient background, and C is the response source. The target-off branch uses its weighted native background gradient. No source parameters or optimizer moments are imported. The matched-random and preclip-norm controls use:

$$r_\ell = \|\Delta G_\ell\|_2 \frac{z_\ell}{\|z_\ell\|_2}, \qquad G_{\mathrm{random}} = G_{\mathrm{nat}} + r, \tag{15}$$

$$G_{\mathrm{norm}} = \frac{\|G_{\mathrm{tr}}\|_2}{\|G_{\mathrm{nat}}\|_2} G_{\mathrm{nat}}, \tag{16}$$

where each parameter tensor forms one group ℓ, including biases, LayerNorm/adaLN parameters, and embeddings. The Gaussian vector z_ℓ is redrawn each step from a seed independent of the training seed, and r concatenates the group-wise corrections r_ℓ. The resulting complete gradients are G_{random} and G_{norm}. When $\Delta G_\ell = 0$, $r_\ell = 0$. Zero-correction groups and other zero-norm degeneracies are recorded separately. The norm control uses the global norm over all parameters. Its scale factor is $\alpha_s = \|G_{\mathrm{tr}}\|_2 / \|G_{\mathrm{nat}}\|_2$. In interleaved training it scales both the background and target terms.

Training uses bf16 autocast and fp32 master weights. After gradient accumulation and unscaling, we compute G_{nat} and G_{tr}, construct the control gradient, clip its global norm at 1.0, and apply AdamW (Loshchilov & Hutter, 2019) with learning rate 10^{-4}, $(\beta_1, \beta_2) = (0.9, 0.999)$, and zero weight decay. All branches use this order. Table 10 reports the pulse-chase matching diagnostics. Matching preclip gradients does not match parameter updates after momentum and adaptive scaling.

Table 10: Gradient-control diagnostics for reciprocal pulse-chase transfers. Norm-ratio deviations are measured before clipping. IQR denotes the interquartile range.

Diagnostic	$R \leftarrow U$	$U \leftarrow R$
α_s, median [IQR]	1.08 [1.03, 1.14]	0.93 [0.89, 0.97]
Maximum $\|G_{\mathrm{norm}}\|_2 / \|G_{\mathrm{tr}}\|_2 - 1$	3.1×10^{-7}	2.8×10^{-7}
Maximum $\|r_\ell\|_2 / \|\Delta G_\ell\|_2 - 1$	4.4×10^{-7}	4.0×10^{-7}
Zero-norm steps	0	0
Zero-correction groups per step	0	0

For the common-checkpoint diagnostic, define:

$$g_T = \nabla_\theta L_T(\bar{\theta}), \qquad u = -g_T / \|g_T\|_2, \qquad h_S = u^\top H^S_B(\bar{\theta}) u, \tag{17}$$

where $\bar{\theta}$ is the target-free step-2,500 checkpoint and L_T averages the panel's ten targets. All parameter tensors are flattened in sorted-name order. The direction spans all network parameters and is fixed before downstream outcomes are observed. All three backgrounds share $\bar{\theta}$, u, and the evaluation bank of flow times and noise. Target images and stochastic inputs used to construct u are also fixed. A double-backward Hessian-vector product evaluates the full flow-matching Hessian, including its residual term, without an added quadratic penalty. Dot products accumulate in FP64 with TF32 and autocast disabled throughout geometry measurement. Negative values are retained. The maximum deviation $\|u\|_2 - 1$ was 2.2×10^{-16}, and all eight panels satisfied $h_U > h_M > h_R$.

Retention uses the same decoy correction at the initial and continuation steps:

$$B^{\mathrm{spec}}_{S,j}(s) = \big[L_j(\theta^-_{S,s}) - L_j(\theta^+_{S,s})\big] - \big[L_{D_j}(\theta^-_{S,s}) - L_{D_j}(\theta^+_{S,s})\big], \tag{18}$$

$$\mathcal{R}_{S,j}(s) = B^{\mathrm{spec}}_{S,j}(s) / B^{\mathrm{spec}}_j(s_0), \tag{19}$$

Table 11: Evaluation settings. Generation budgets are per class and model. Target extraction rate is an any-hit endpoint over target images, and target copying rate is a per-generation endpoint. Photometric evaluation follows Appendix G.

Experiment	Evaluation point	Images	Endpoint
Balanced	Step 10,000, EMA	2,000	Target extraction rate
Composition	Step 10,000	2,000	Target extraction rate
Mechanism	Step 6,000, raw parameters	4,096	Target copying rate

where j identifies a target, D_j contains its fixed untrained same-class decoys, and $B_j^{\mathrm{spec}}(s_0)$ is shared across continuations. Each image loss averages 40 evaluations: eight fixed noise draws at each flow time $t \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. Target and decoy losses use the same times and noise, as does the target loss defining u. Ratios are computed per target before panel averaging, with panels as the statistical units. Signed, unnormalized benefits are retained alongside ratios. The exclusion threshold $|B_j^{\mathrm{spec}}(s_0)| < 0.02$ was fixed before terminal outcomes were examined and applied identically across conditions. Unnormalized benefits retain any targets excluded from ratio summaries. Across the 200 panel-target pairs in the three settings, all denominators were positive: their minimum, median, and maximum were 0.304, 0.401, and 0.498. None was zero, negative, or below the threshold, so all 200 pairs were included. For every pair, the denominator difference across conditions was zero and $\mathcal{R}_{S,j}(s_0) = 1$.

Table 5 reports signed benefits and normalized retention at step 6,000. The two summaries gave the same paired-effect direction in all four reciprocal-transfer comparisons, with agreement in all eight panels for each comparison. Pulse-chase ratios measure retention after target training stops, whereas interleaved ratios combine continued target learning and retention.

Both $B^{\mathrm{spec}}(s)$ and $\mathcal{R}(s)$ are recorded at $s - s_0 \in \{0, 1, 10, 50, 100, 300, 1000, 3100\}$. Signed target-specific benefit decreased at each sampled continuation time in both regimes, as shown in Table 28.

Finite-response comparisons use $\kappa_{C|A,s}$ at fixed recipient states at steps $\mathcal{S} = \{3000, 4000, 5000, 6000\}$, with four fixed random batches at each step. For panel p, we first average over steps and batches:

$$\bar{\kappa}_{C|A}^{(p)} = \frac{1}{16} \sum_{s \in \mathcal{S}} \sum_{b=1}^{4} \kappa_{C|A,s,b}^{(p)}, \tag{20}$$

where b indexes the evaluation batch. The panel's relative change is then $100(\bar{\kappa}_{C|A}^{(p)} / \bar{\kappa}_{A|A}^{(p)} - 1)\%$. These panel changes are averaged to obtain the reported summary. The signed absolute difference is $\bar{\kappa}_{C|A}^{(p)} - \bar{\kappa}_{A|A}^{(p)}$. Relative comparisons exclude a panel when $|\bar{\kappa}_{A|A}^{(p)}|$ is below 10^{-3} times the overall median native projection. No panel met this criterion. A random correction's directional projection describes the effective correction and is not reported as background Hessian curvature.

Generation evaluates raw step-6,000 parameters with 4,096 images per class, pooled across two shared noise banks, using a 50-step Euler sampler. The displacement-scaling evaluation uses the terminal native interleaved R pair with the same sampling settings.

A.2 Additional Inference Details

Balanced extraction outcomes are analyzed with bias-reduced logistic regression. Target repetition and background diversity enter as $\log_2(k/4)$ and $\log_2(N_{\mathrm{bg}}/100)$, with their fitted trade-off summarized by $\lambda = -b_N/a$. Fieller intervals are used for this ratio, and standard errors are clustered by target and run.

Within each training regime, the four primary copying contrasts (the two transfer directions against native training and against their matched-random controls) form one family for Holm-adjusted decisions. This familywise adjustment is separate from the ordinary 95% estimation intervals reported in the tables. A transfer fraction divides the mean intervention effect by the native R-U copying gap in the same regime. Its uncertainty is estimated by paired bootstrap resampling over

panels. The two additional pulse-chase settings use six independent panels per setting and the same paired panel-level summaries. Full transfer summaries and their inference protocol appear in Appendix A. Balanced-model specifications and interval procedures appear in Appendix H.

A.3 Paired Mechanism Contrasts

All contrasts below first average the ten class-specific target measurements within a panel, then compare paired arms across panels. Confidence intervals use the two-sided Student-t quantile with $n-1$ degrees of freedom, where n is the number of panels. Copy effects are measured in percentage points. For receiver R, a positive effect is comparator minus transferred copying, whereas for receiver U, it is transferred minus comparator copying. Thus, positive effects represent reduction in R and increase in U. Individual intervals are estimation intervals. The Holm adjustment applies to the tests, not the intervals.

For pulse-chase R, the random correction reduces copying by 0.277 pp (95% interval $[-0.062, 0.617]$) relative to native training. Response transfer reduces it by 0.953 pp and exceeds the random correction by 0.676 pp ($[0.489, 0.863]$, with Holm-adjusted $p = 1.21 \times 10^{-4}$).

Table 12: Paired copying contrasts across eight panels in each primary regime. The four native/random comparisons constitute one Holm family per regime. Preclip comparisons are supplementary and are not assigned a p value from that four-test family (n/a).

Regime	Receiver / comparator	Effect [95% CI]	Holm p
Pulse-chase	R / native	0.953 [0.630, 1.277]	2.17×10^{-4}
	U / native	1.099 [0.929, 1.269]	4.97×10^{-6}
	R / random	0.676 [0.489, 0.863]	1.21×10^{-4}
	U / random	0.986 [0.805, 1.167]	1.17×10^{-5}
	R / preclip	0.692 [0.486, 0.898]	n/a
	U / preclip	0.948 [0.758, 1.137]	n/a
Interleaved	R / native	0.919 [0.728, 1.111]	2.76×10^{-5}
	U / native	0.810 [0.621, 0.998]	3.85×10^{-5}
	R / random	0.700 [0.497, 0.903]	8.16×10^{-5}
	U / random	0.766 [0.620, 0.912]	2.03×10^{-5}
	R / preclip	0.605 [0.301, 0.908]	n/a
	U / preclip	0.729 [0.523, 0.934]	n/a

Including both preclip contrasts in a supplementary six-test Holm family gives adjusted p values of 1.92×10^{-4} and 2.78×10^{-5} for pulse-chase R and U, and 0.00218 and 2.02×10^{-4} for interleaved R and U. This sensitivity analysis is separate from the four primary tests.

Table 13: Paired differences in normalized target-specific benefit, with ordinary 95% panel intervals. Each entry uses eight panels and the target-wise normalization in Equation 11. The signs follow the copying contrasts: comparator minus transfer for R, and transfer minus comparator for U.

Receiver / comparator	Pulse-chase	Interleaved
R / native	0.1562 [0.1298, 0.1827]	0.1247 [0.1072, 0.1422]
U / native	0.1494 [0.1413, 0.1575]	0.1170 [0.0993, 0.1348]
R / random	0.1191 [0.1005, 0.1378]	0.1123 [0.0969, 0.1276]
U / random	0.1383 [0.1264, 0.1503]	0.1089 [0.0933, 0.1245]
R / preclip	0.1203 [0.1005, 0.1401]	0.0984 [0.0742, 0.1226]
U / preclip	0.1253 [0.1074, 0.1433]	0.1050 [0.0859, 0.1242]

Transfer Fractions. The R and U transfer fractions are respectively 44.1% [34.7, 51.7] and 50.8% [44.0, 58.7] in pulse-chase, and 44.2% [38.5, 49.6] and 38.9% [35.3, 42.3] in interleaved training. Brackets are 95% paired-panel bootstrap intervals. The numerator is the panel-mean transfer effect and the denominator is the panel-mean native R-U gap from the same regime. Panels are resampled jointly in numerator and denominator. These are intervention effect fractions, not natural mediation proportions.

A.4 Complete Cross-Setting Copying Rates

The additional settings use six panels each with matched background row counts and random-direction controls. The CIFAR-10/U-Net comparison retains CIFAR-10 targets, flow matching, and AdamW. CINIC-10 is decoded-RGB deduplicated. Table 14 reports native, transfer, and matched-random rates together with the paired beyond-random effect. The primary DiT/CIFAR-10 arm rates are already reported in Table 7. They are not additional replications.

Table 14: Complete pulse-chase rates in the additional settings, each with six panels. Native, transfer, and random columns give copying percentages with 95% panel intervals. The last column gives the paired beyond-random effect in percentage points. Settings are analyzed separately.

Receiver	Native	Transfer	Random	Beyond random
CINIC-10 / DiT				
R	2.000 [1.606, 2.394]	1.318 [1.005, 1.630]	1.902 [1.516, 2.288]	0.584 [0.453, 0.716]
U	0.446 [0.347, 0.545]	1.157 [0.924, 1.389]	0.542 [0.426, 0.658]	0.615 [0.485, 0.744]
CIFAR-10 / U-Net FM				
R	1.742 [1.538, 1.947]	1.174 [1.086, 1.262]	1.662 [1.492, 1.832]	0.488 [0.386, 0.590]
U	0.401 [0.374, 0.428]	0.969 [0.897, 1.041]	0.466 [0.408, 0.524]	0.503 [0.431, 0.574]

Table 15: Native background gaps during continued training after target removal. Entries give the native R minus U target copying rate in percentage points, with paired 95% panel intervals. The primary setting is included as the reference comparison.

Setting	Experimental panels	Native R minus U
CIFAR-10 with DiT, primary	8	2.162 [1.884, 2.440]
CINIC-10 with DiT	6	1.554 [1.249, 1.859]
CIFAR-10 with U-Net	6	1.341 [1.143, 1.540]

Table 16: Composition means over 150 paired target evaluations per condition. Entries specify distinct images multiplied by repetition counts. Both rates are percentages. Relabeled other-class images retain the target conditioning label.

Background composition	Target extraction rate	Conditional accuracy
Same-class 400×1	18.0	56.6
Same-class 25×16	80.7	93.9
Same-class 25×1	86.0	64.9
Relabeled other-class 400×1	20.0	8.3
Relabeled other-class 25×16	82.0	5.8

A.5 Additional Visual Summaries

Figures 4 and 5 visualize the data-composition and cross-setting results. Numerical summaries appear in Tables 16 and 4.

Relabeled-Background Control. Replacing 25 relabeled other-class images repeated 16 times with 400 distinct relabeled images reduces target extraction from 82.0% to 20.0%. Their conditional accuracies are 5.8% and 8.3%, respectively. These images retain the target conditioning label. Table 16 reports the complete composition results.

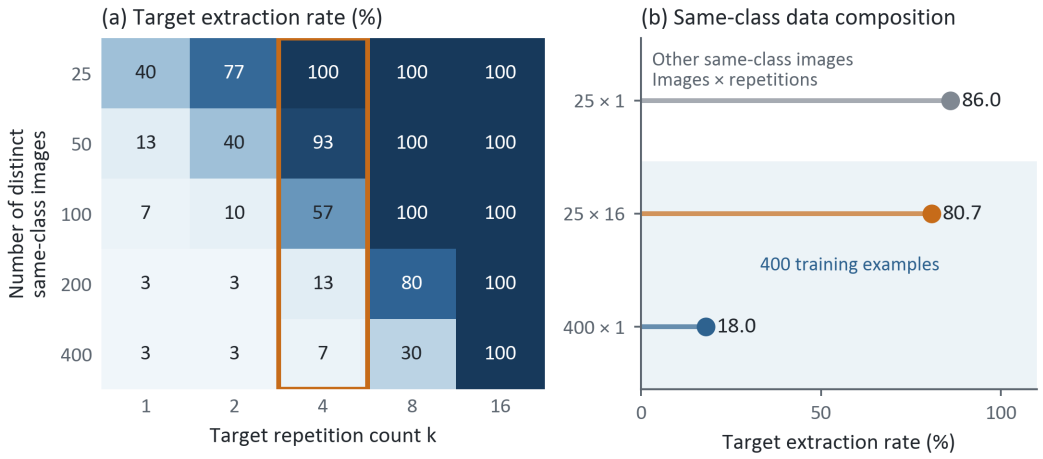


Figure 4: Same-class data composition changes target extraction rate. (a) The balanced design contains 30 target images per cell and 2,000 generated images per class and model. Cell labels are rounded percentages, and the outline marks $k = 4$. (b) Composition means over 150 paired target evaluations per condition. Labels give distinct images multiplied by their repetition counts. The shaded conditions contain 400 other same-class training examples. Full coefficients and block analyses appear in Appendix H.

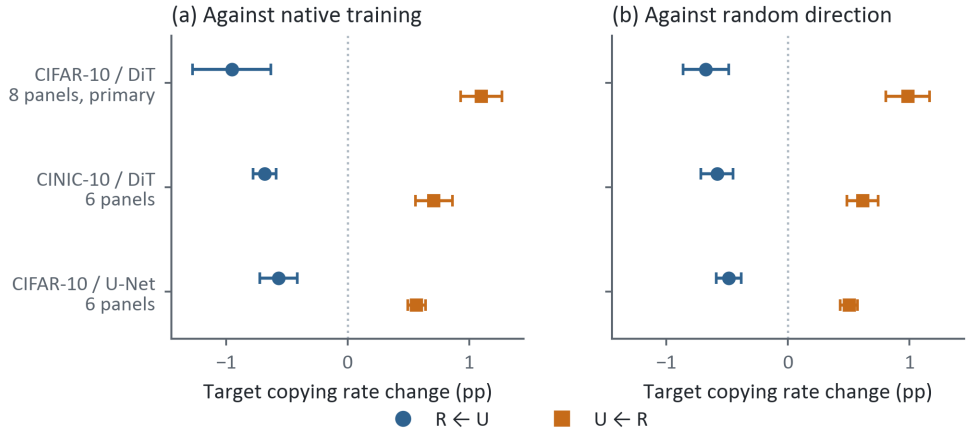


Figure 5: Reciprocal response transfer across settings during continued training after target removal. Changes in target copying rate are measured against (a) native training and (b) random-direction controls. Negative values indicate reduced copying under $R \leftarrow U$, and positive values indicate increased copying under $U \leftarrow R$. Intervals are paired 95% panel intervals. The primary CIFAR-10 with DiT result provides the reference comparison with eight experimental panels. Each additional setting uses six experimental panels. Native background gaps appear in Table 15.

A.6 Parameter-Displacement Scaling: Implementation Details

Section 4.5 reports the state intervention and its results. Equation 12 is evaluated on the terminal native interleaved R pair from each of the eight panels. The scaling factors $\gamma = 0$ and $\gamma = 1$ directly load the target-off and target-on checkpoints. The remaining values interpolate or extrapolate along the same displacement. Within each panel, all conditions share generation noise banks, sampler, reference images, detection thresholds, and generation budget. The $\gamma = 1$ endpoint reuses the native evaluation.

A random parameter-direction control matches the displacement norm at $\gamma = 1$. A non-target learning direction provides a second comparison. Target copying, decoy hit, and non-target copying rates assess response specificity. At every scale, target-specific benefit uses $\theta^{-}_{R,6000}$ as the common target-off reference and the same loss-evaluation bank. Table 3 in the main text reports the panel means. Figure 6 provides a graphical view of those measurements.

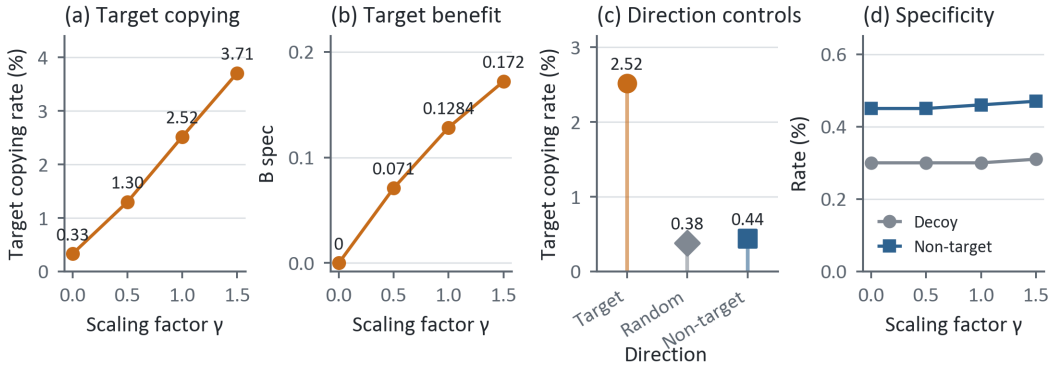


Figure 6: Displacement scaling connects retained target learning to generated copying. (a) Target copying rate across scaling factors for the step-6,000 native interleaved R pair. (b) Target-specific benefit across the same scaling factors, using the same target-off reference and loss evaluation bank. (c) The $\gamma = 1$ target-induced parameter displacement compared with a random parameter-direction control and a non-target learning direction. The random control matches the $\gamma = 1$ displacement norm. (d) Decoy hit rate and non-target copying rate across the same scaling factors. Points are reported means over eight experimental panels. Lines connect evaluated scaling factors, and all conditions share generation inputs and detection settings.

B Controlled-Geometry Experiments

This separate experiment isolates the orientation of an added quadratic penalty. Its background supervision and eight-step SGD schedule differ from the natural-response experiments in the main text. It provides a controlled test of orientation at fixed eigenvalues, whereas response transfer changes the full finite response of a true background loss.

B.1 Isolating Target-Specific Learning

The internal experiment starts from a trained model and isolates the updates induced by a target. Target rows retain their true FM velocity, while background rows use the initial model's predictions as fixed supervision. Background gradients are therefore zero initially, allowing the target to supply the first learning signal. All network parameters remain trainable throughout the intervention. Two independently trained initial models use the same ten new targets, one per class, giving 20 model-target conditions.

Each condition uses 4,000 backgrounds, eight target copies, and eight SGD updates with target counts $(1, 1, 0, 1, 0, 0, 1, 0)$, learning rate 10^{-4}, batch size 256, and no momentum. The two treatments share all draws. A target-free control checks initial stationarity.

We remove shared gradient components using the mean and first principal component from a separate set of 50 construction targets:

$$U = \mathrm{unvec}\big[(I - pp^\top)(g_\star - \bar{g})\big], \tag{21}$$

where $g_\star$ is the target's vectorized output-layer gradient and $\bar{g}$ is the mean gradient over construction targets. The unit vector p is their first principal direction, and I is the identity matrix on gradient coordinates. The operation unvec restores the output-layer matrix shape, giving the *residual target gradient* U. The ten evaluation targets are excluded from estimating $\bar{g}$ and p. Each evaluation target's own gradient is used to construct its paired penalty orientations. Their residual-to-total gradient norm ratios must exceed the prespecified threshold of 0.5. Targets failing this check cannot be replaced.

B.2 Changing Curvature Orientation at Fixed Eigenvalues

The final linear output layer, including bias, is $W_0 + D$, where $W_0 \in \mathbb{R}^{o\times q}$ contains its initial weights and $D \in \mathbb{R}^{o\times q}$ their change. Here $o = 12$ is the number of output coordinates per patch, and $q = 385$ is the feature dimension including the bias coordinate. We add a quadratic penalty:

$$\begin{aligned} R_\nu(D) &= \frac{\mu}{2}\,\mathrm{tr}(DQ_\nu D^\top), \\ Q_\nu &= V_\nu \Lambda V_\nu^\top, \qquad \nu \in \{\mathrm{hi}, \mathrm{lo}\}, \end{aligned} \tag{22}$$

where R_ν is the penalty in treatment ν, with strength $\mu = 16$. The positive-semidefinite matrix Q_ν acts on feature coordinates. Both treatments share the diagonal matrix of background-curvature eigenvalues Λ and differ in the orthogonal eigenvector matrix V_ν. Curvature along the residual target gradient is $\mathrm{tr}(UQ_\nu U^\top)/\,\|U\|_F^2$, where tr denotes trace and $\|\cdot\|_F$ denotes the Frobenius norm. This quantity is larger for the high-alignment treatment, $\nu = \mathrm{hi}$, than for the low-alignment treatment, $\nu = \mathrm{lo}$. The operators have matching eigenvalues, trace, and Frobenius norm, and use pre-intervention noise separate from evaluation.

At $D = 0$, both penalties and their gradients are zero, giving the treatments the same initial target gradient. Once the weights change, the penalty gradient μDQ_ν opposes that change with direction-dependent strength. Holding eigenvalues fixed isolates the effect of orientation from the overall strength of the penalty.

B.3 Predicting Target-Loss Improvement

We measure learning as the decrease in target loss and compare that decrease between treatments:

$$\mathcal{W}_\nu = L_{00} - L_{11}^\nu, \qquad \Delta\mathcal{W} = \mathcal{W}_{\mathrm{lo}} - \mathcal{W}_{\mathrm{hi}}, \tag{23}$$

where L_{00} is the initial mean target FM loss on a fixed evaluation noise bank. The loss L_{11}^ν is measured after updating the full network under treatment ν, and $\mathcal{W}_\nu$ is the resulting target-loss improvement. The difference $\Delta\mathcal{W}$ measures how much high alignment reduces this improvement relative to low alignment. We use two evaluation noise banks, H and D. Each contains eight noise draws at each flow time in $\{0.125, 0.375, 0.625, 0.875\}$. The banks share targets, which form the independent units. A 10,000-replicate hierarchical bootstrap resamples targets and paired noise draws within each flow time, with separate decisions for each initial model.

We predict output-layer updates while holding features at their initial values:

$$\begin{aligned} \widehat{D}_{s+1}^\nu &= \widehat{D}_s^\nu - \eta\left[G_s + \widehat{D}_s^\nu(C_s + \mu Q_\nu)\right], \\ \widehat{D}_0^\nu &= 0, \end{aligned} \tag{24}$$

where $s \in \{0, \ldots, 7\}$ indexes SGD updates and $\widehat{D}_s^\nu$ is the predicted change in output-layer weights. The learning rate is $\eta = 10^{-4}$. For the batch scheduled at update s, G_s is the loss gradient at the initial model and C_s is the squared-loss curvature with features fixed. Appendix L gives their matrix forms. The recurrence is exact with fixed features. Our experiment tests its prediction after full-network updates.

Predictions are saved before accessing post-intervention losses. We compare them with observed treatment effects using slope, correlation, and paired target-label permutations.

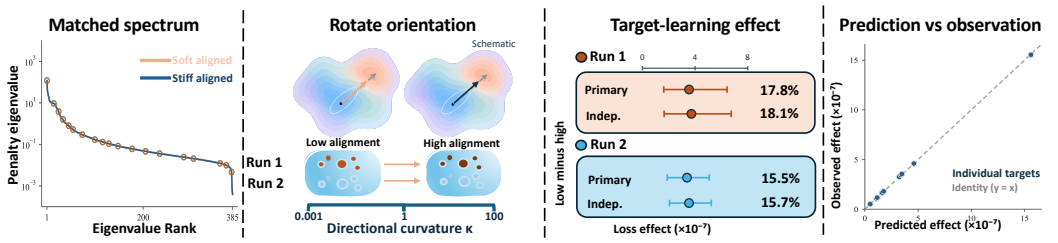


Figure 7: Controlled penalty orientation and short-horizon target learning. Runs 1 and 2 are independent models of the same architecture. Primary and independent evaluation banks reuse the same ten targets with different noise. Bars show low-minus-high learning effects with 95% paired hierarchical-bootstrap intervals. Percentages show relative reductions. The orientation panel is schematic. Predictions and observations use units of 10^{-7} mean FM loss.

Leave-one-target-out (LOTO) calibration fits an intercept and slope on nine targets and predicts the tenth. Its Q^2 compares prediction error with the nine-target-mean predictor. Post-intervention losses enter this calibration, not the recurrence.

B.4 Native-Training Controls

To connect target learning with generated copying, we continue ordinary FM training from three target-free checkpoints for 3,000 AdamW updates. Each checkpoint is paired with ten new targets. Distinct-background and repeated-background conditions contain respectively 400 different images or 25 images repeated 16 times per class. Both include 16 copies of each target and 4,160 training rows.

At each update we exchange the background-gradient component along the current target gradient between conditions, evaluating both gradients at the recipient model. Equal-norm orthogonal controls test directional specificity. A simple repeated-background baseline halves the target loss weight while leaving the background weight unchanged. All comparisons share generated noise and a fixed reference bank. We report generated-copy rate, the number of attributable hits divided by generated samples, alongside conditional accuracy. Appendix I specifies these controls and reports every block.

B.5 Controlled-Geometry Results

High alignment reduces eight-step target-loss improvement by 15.47–18.14% across two initial models and two noise banks, as reported in Table 24. Table 25 reports slopes of 0.9950 and 0.9978 for the pre-intervention recurrence's predictions of the full-network treatment difference. The equal first update and fixed penalty spectrum isolate orientation in this controlled objective. The earlier native target-gradient exchange uses a different operation from Equation 7. Its block-level results and controls are retained in Appendix I.

C Supplementary Data Analyses

The balanced experiment estimates repetition and support effects with bias reduction and target/run clustering. The composition experiment uses a post hoc conditional Firth analysis because the ordinary conditional maximum-likelihood estimate is nonfinite. The following sections retain the reference-policy controls, assignment rules, and separate controlled-geometry experiments.

C.1 Training Configuration for the Balanced Data Experiment

The balanced experiment uses a DiT with 12 transformer blocks, width 384, six attention heads, and 2×2 patches. Training runs for 15,000 steps with batch size 256 and AdamW: learning rate 10^{-4}, momentum coefficients $(0.9, 0.95)$, zero weight decay, and gradient-norm clipping at 2.0. The EMA decay is 0.9995. The primary endpoint is the 10,000-step EMA checkpoint. The 5,000-, 7,500-, 12,500-, and 15,000-step checkpoints support secondary analyses. Sampling uses 50 velocity

evaluations, with generation and quality random streams separate from training. The composition models are trained for 10,000 steps. The eight-panel response experiments instead evaluate raw parameters at step 6,000, as specified in Section 4.1.

The data experiments shuffle training rows at every epoch and use `drop_last=True`. Target multiplicity k counts rows in the dataset. Expected exposure under row sampling and the actual number of target presentations are separate quantities: dropping the final incomplete batch can change realized exposure. We distinguish these quantities from the exact target-draw counts in the stratified mechanism schedule.

D NOTATION

Table 17: Additional notation used in the earlier data and controlled-geometry experiments. Flow time describes the noise-to-data interpolation. The update index describes optimization.

Symbol	Meaning
x, c, ϵ, t	Image, conditioning class, Gaussian noise, and flow time.
θ, v_θ	Network parameters and class-conditioned velocity field.
k, N_{bg}	Target-copy count and distinct non-target same-class image count in the balanced design.
b, r	Cycle and training-run indices.
$A_c(y), E_{bc}^S$	Attributable hit for one generated image and at least one hit for a target in a model, respectively.
a, b_N, λ	Repetition coefficient, support coefficient, and fitted exponent $-b_N/a$.
W_0, D, U	Initial output-layer weights, their change, and the residual target gradient.
ν, Q_ν, μ	Treatment, penalty operator, and penalty strength.
s, G_s, C_s	SGD update index, initial batch gradient, and batch curvature with features fixed.
$\mathcal{W}_\nu, \Delta\mathcal{W}$	Target-loss improvement and low-minus-high treatment contrast.

E MEASUREMENT UNITS

Matched untrained references give target-level extraction baselines of 3.0%, 2.7%, and 3.3% for the balanced, composition, and photometric experiments, respectively.

For a generated image y conditioned on class c, let $n_c(y)$ be its nearest image in the class reference bank under distance d_{det}. For a particular target $x^\star$, let $\mathcal{T}_c^\star(x^\star)$ contain the references attributed to that target. The hit indicator is:

$$A_{c,x^\star}(y) = \mathbf{1}\{d_{\mathrm{det}}(y, n_c(y)) < r_c\}\, \mathbf{1}\{n_c(y) \in \mathcal{T}_c^\star(x^\star)\}\,, \tag{25}$$

where r_c is the first percentile of independent holdout-to-reference distances. Equivalently, the negative-distance similarity must exceed its 99th-percentile holdout threshold. The primary distance is unnormalized pixel ℓ_2. LPIPS-Alex is calibrated separately using the same holdout set and evaluated on the same generated pool.

For M generated images of the target's class, copying rate and extraction are respectively:

$$p_{c,x^\star} = \frac{1}{M}\sum_{j=1}^{M} A_{c,x^\star}(y_j), \qquad E_{c,x^\star} = \mathbf{1}\left\{\sum_{j=1}^{M} A_{c,x^\star}(y_j) > 0\right\}. \tag{26}$$

All generated images enter the copying-rate denominator. The data experiments use one common pool of $M = 2{,}000$ images per class and model. When a class contains multiple targets, each target is evaluated against this same pool. The mechanism experiments use $M = 4{,}096$ images per class from two shared noise banks. Their reference bank contains one target, 400 background-pool images, and 16 fixed untrained same-class decoys. Another 984 holdout images calibrate the threshold. Target-loss improvement, in contrast, measures the decrease in mean FM loss under fixed evaluation noise.

Untrained-Reference Extraction. Fixed same-class decoys provide an untrained-reference baseline for each data experiment. Each check uses the experiment's $M = 2{,}000$ generation budget, detector, reference policy, and threshold, with the same reference-bank size as the trained-target evaluation. Table 18 reports the fraction of untrained references hit at least once. Their per-generation hit probability is approximately 1.5×10^{-5}.

Table 18: Untrained-reference extraction under the matched detector and generation budget. Each entry is an any-hit rate over $M = 2{,}000$ generations.

Data experiment	Untrained-reference extraction (%)
Balanced	3.0
Composition	2.7
Near-duplicate	3.3

For the same-class composition arms, subtracting the matched 2.7% decoy baseline from the trained-target rates of 18.0%, 80.7%, and 86.0% gives descriptive trained-minus-decoy differences of 15.3, 78.0, and 83.3 percentage points for 400×1, 25×16, and 25×1, respectively.

F Predictor and Intervention Checks

Combining Initial and Updated Network Components. We evaluate all four combinations of initial or updated features and initial or updated output-layer weights. Alongside L_{00} and L^{ν}_{11}, we compute L^{ν}_{01} and L^{ν}_{10}. Here L^{ν}_{01} uses initial features with updated output-layer weights, while L^{ν}_{10} uses updated features with initial output-layer weights. These recombinations check forward reconstruction and separate the contributions of the two network components. They are implementation checks. Predictive accuracy is evaluated separately against the saved predictions.

Independent Units. The two initial models, both evaluation noise banks, and both predictor execution modes use the same ten target images. Permutations preserve each target's pairing across initial models, and uncertainty calculations use targets as the independent units. Reporting each initial model separately assesses consistency across models while retaining the sample size of ten targets.

Prediction Inputs and Analysis Order. The predictor uses the initial model, fixed update schedule, penalty matrices, and evaluation noise definitions. Predictions are saved before the analysis accesses updated parameters or post-intervention losses. The final prediction test therefore separates computation of the predictions from their comparison with observed effects. The choice of model and intervention followed earlier exploratory analyses.

G Near-Duplicate Protocol

The near-duplicate experiment compares exact copies, light variations, and strong variations across five paired seeds, three training arms, and five saved checkpoints. Variations change brightness, contrast, and pixel noise while preserving spatial arrangement. Brightness and contrast multipliers range over $[0.95, 1.05]$ for light variations and $[0.90, 1.10]$ for strong variations. Additive Gaussian-noise standard deviations are 0.005 and 0.010, respectively, on the $[-1, 1]$ image scale, followed by clamping. Every target occurrence is transformed in the variation arms, including targets that occur once, while background images remain unchanged. A separate random-number generator pairs transformation draws across variation strengths.

We compare three reference policies: all training variants, the original target image, or one training variant per target. These correspond to `training-span`, `canonical-only`, and `matched-one` in the stored results. The two one-reference policies prevent the number of matching opportunities from changing across conditions. The matched-one policy uses 16 detector replicates so each training variant is selected equally often at every repetition level. Each policy is recalibrated on the same holdout set. With the original-image policy, reference images and thresholds are identical across the three training conditions. We estimate:

$$k_{\text{eff}}/k = 2^{\psi/a_{\text{near}}}, \tag{27}$$

where ψ is the fitted log-odds shift for a variation condition relative to exact copies. The coefficient a_{near} describes the effect of doubling repetition in this experiment. Effective repetition k_{eff} is the exact-copy count giving the same fitted extraction probability. The model includes class, seed, and checkpoint effects. Paired target-cluster bootstrap intervals, conditional on the five paired training seeds, test equivalence within $[0.8, 1.25]$, while interactions test whether variation strength changes the repetition slope.

Table 19: Effective repetition relative to exact copies under the two one-reference policies. Brackets are 90% paired target-cluster bootstrap intervals conditional on the five paired training seeds.

Reference policy	Light variations	Strong variations
Original image	0.987 [0.968, 1.006]	0.983 [0.963, 1.005]
One training variant	0.996 [0.978, 1.014]	0.982 [0.959, 1.006]

All four intervals lie within the prespecified equivalence range $[0.8, 1.25]$. Searching all training variants gives point estimates of 1.054 and 1.073 for light and strong variations, respectively. The one-reference policies hold the number of matching opportunities fixed. Figure 10 shows all three policies and the equivalence range.

A separate sensitivity summary uses training seed as the outer unit. The five seed-level effective-repetition ratios are 0.971, 0.984, 0.992, 1.003, and 1.011, with a 90% seed-level interval of $[0.93, 1.05]$, also inside $[0.8, 1.25]$. This summary assesses variation across training seeds. Table 19 retains the four policy- and strength-specific conditional intervals.

H Statistical Estimation Details

For the balanced extraction endpoint, the fitted model is:

$$\begin{aligned} \operatorname{logit} \Pr(E_{brc} = 1) &= \alpha_{bc} + \rho_{br} + a \log_2(k/4) + b_N \log_2(N_{\text{bg}}/100), \\ \lambda &= -b_N/a, \end{aligned} \tag{28}$$

where b indexes cycles, r runs, and c classes. Each cycle has one fixed target per class, so α_{bc} is a cycle-target effect and ρ_{br} is a run effect. The logistic model has separation, so we use Firth/Jeffreys bias reduction (Firth, 1993). One target has the same extraction outcome in every condition and contributes no within-target variation. Excluding its 25 rows leaves 725 observations, 29 target clusters, and 75 run clusters. The coefficient intervals in Table 20 use two-way cluster-robust covariance (Cameron et al., 2011), and exponent intervals use Fieller's method (Fieller, 1954). The 10,000-replicate wild-score sensitivity resamples target clusters. A separate sensitivity resamples run clusters. These are separate one-way resampling analyses. Balance-preserving assignment randomization uses 100,000 draws and gives $p = 1/100001$ for each coefficient's direction. Leave-one-cycle-out prediction fits on two cycles without using target or run effects from the held-out cycle. Adding support to repetition alone reduces held-cycle log-loss by 0.0064, 0.2088, and 0.2061.

Table 20: Balanced extraction estimates at the 10,000-step EMA checkpoint. Coefficient intervals use two-way target/run cluster-robust covariance. Exponent intervals use Fieller's method. Pixel fitting uses 725 informative observations, and LPIPS evaluates the same generated pools.

Endpoint	Quantity	Estimate	95% interval
Pixel ℓ_2	Repetition coefficient a	3.498	[3.098, 3.898]
	Support coefficient b_N	-2.297	$[-2.607, -1.987]$
	Support exponent λ	0.657	[0.610, 0.703]
LPIPS-Alex	Support exponent λ	1.043	[0.912, 1.181]

The composition conditional likelihood has quasi-complete separation, so the ordinary maximum-likelihood estimate is not finite. We report a *post hoc* conditional Firth/Jeffreys analysis, conditioning on each target's hit total and retaining identifiable run effects. The fit uses 115 informative targets, 575 rows, four condition parameters, and twelve run parameters. Intervals use profile penalized

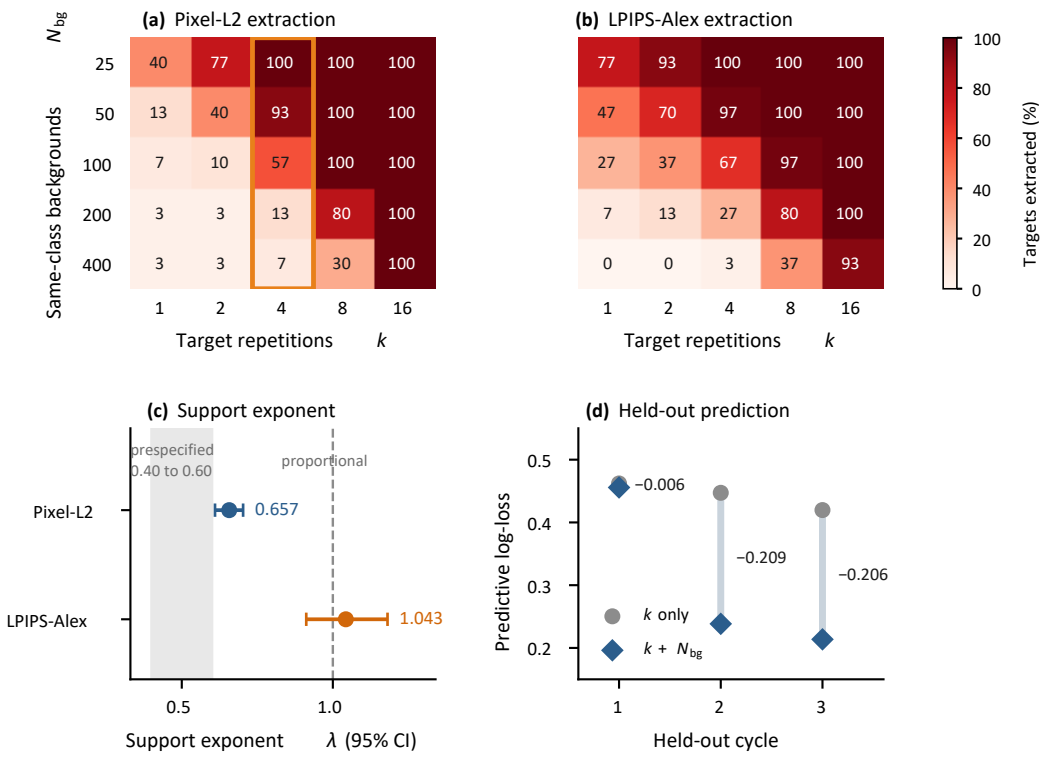


Figure 8: Complete balanced-design results: pixel and LPIPS extraction, endpoint-specific support exponents with 95% Fieller intervals, and held-cycle prediction. The gray exponent band is the prespecified $[0.4, 0.6]$ reference range, not an uncertainty interval. Each heatmap cell contains 30 targets.

likelihood, with Holm correction (Holm, 1979) for four contrast decisions. Table 21 reports these estimates separately from the paired extraction frequencies and quality differences. The first three contrasts satisfy the Holm-adjusted significance criterion.

Table 21: Composition results for 150 paired targets across three blocks. Contrasts are left minus right. Log-odds estimates and 95% profile intervals come from the post hoc conditional Firth analysis. Extraction and accuracy differences are percentage points. Accuracy uses the intended conditioning class.

Background contrast	Extraction change	Log-odds [95% interval]	Accuracy change
Same-class: 400 distinct vs. 25×16	-62.67	-6.76 $[-9.86, -4.78]$	-37.29
Other-class: 400 distinct vs. 25×16	-62.00	-5.83 $[-8.42, -4.23]$	$+2.49$
Same-class: 25×16 vs. 25 distinct	-5.33	-1.68 $[-3.29, -0.43]$	$+29.03$
400 distinct: same-class vs. other-class	-2.00	-1.20 $[-2.89, 0.34]$	$+48.37$

Composition Sensitivity across Training Blocks. The three block-specific extraction contrasts are shown in Table 22. Each contrast retains its pooled direction in all three leave-one-block-out comparisons. Ordinary two-sided 95% t intervals use the three paired block contrasts as independent units (2 degrees of freedom), without multiplicity adjustment. All three intervals exclude zero. For

the row-count comparison (25×16 versus 25 distinct same-class images), the mean difference is -5.3333 percentage points with interval $[-8.6692, -1.9974]$. These block-level intervals and the conditional Firth intervals in Table 21 summarize different units of variation.

Table 22: Composition extraction contrasts by training block, in percentage points. Contrasts have the same orientation as Table 21. The last column gives ordinary, unadjusted 95% t intervals across the three blocks. All three leave-one-block-out estimates retain the pooled orientation.

Background contrast	Block 1	Block 2	Block 3	Mean	95% interval
Same-class: 400 distinct vs. 25×16	-60.9	-63.1	-64.0	-62.6667	$[-66.6283, -58.7050]$
Other-class: 400 distinct vs. 25×16	-60.2	-62.5	-63.3	-62.0000	$[-65.9978, -58.0022]$
Same-class: 25×16 vs. 25 distinct	-3.8	-5.9	-6.3	-5.3333	$[-8.6692, -1.9974]$

I Native-Training Intervention Protocol

Common Start and Native Supervision. Three independently trained, target-free 2,500-step checkpoints retain the raw network and AdamW state. Each receives a separate ten-target panel. The continuation uses 3,000 updates, learning rate 10^{-4}, momentum coefficients $(0.9, 0.95)$, zero weight decay, and clipping at norm 2.0. Every update stratifies 240 background draws and 16 target draws. The background and target losses have weights $w_B = 4000/4160$ and $w_T = 160/4160$, respectively, where w_B and w_T match their training-row proportions. All images retain their true FM velocity supervision. Generation evaluates the raw continued network.

Directional Exchange and Orthogonal Control. At the current recipient parameters, let g_T, g_B, and g_D denote the full-network target, recipient-background, and donor-background gradients. Define:

$$e = g_T / \|g_T\|_2, \quad \delta = \langle g_D - g_B, e \rangle, \quad g_{\text{exchange}} = w_B(g_B + \delta e) + w_T g_T, \tag{29}$$

where e is the unit target-gradient direction, δ is the signed change of the background projection, and g_{exchange} is the gradient passed to clipping and AdamW. Gradients of both background sets use paired noise at the same recipient parameters. The control replaces e in the correction by a unit direction $q_\perp$ orthogonal to e, where $q_\perp$ is obtained by projecting a fixed random vector. Its correction has the same norm $|\delta|$. Exchange is applied in both directions, from distinct to repeated and from repeated to distinct backgrounds. The simple baseline uses $w_B g_B + (w_T/2) g_T$ without donor evaluation or weight renormalization. It is a development follow-up on the same panels.

Common Measurement. Each condition generates 2,000 images per class with the same noise and 50 Euler steps. The common class reference bank contains the target, 16 holdout decoys, and all 400 distinct background images. The other 984 holdout images determine a class-specific 1% distance threshold. Reference hashes and thresholds are identical across interventions in a block. A hit requires the target to be the nearest reference and its distance to be below the threshold. The copy rate is the number of hits divided by the generated-sample count. It is not the percentage of targets extracted at least once. Conditional accuracy is measured with the same frozen classifier used elsewhere. It does not establish equality of the complete image distributions.

The target-direction exchange has no consistent copying advantage over its orthogonal control, and reverse exchange does not restore copying in U. The target-weight baseline improves on R $\leftarrow$ U in both measurements in every block. This earlier intervention exchanges a first-order projection along the target gradient. The response transfer in Section 3.3 instead subtracts the source's off-state gradient and retains the receiver's baseline gradient. The two interventions test different components of training.

Table 23: All native-continuation blocks. Each entry gives attributable copy count / conditional accuracy (%), out of 20,000 generated images. U denotes distinct backgrounds. R denotes repeated backgrounds. Arrows denote the donor of the exchanged component.

Condition	Block 1	Block 2	Block 3
U	0 / 44.02	3 / 45.36	0 / 46.25
R	268 / 71.13	357 / 71.49	531 / 70.90
U ← R	0 / 44.26	3 / 45.64	0 / 44.95
R ← U	93 / 68.59	123 / 69.35	142 / 68.91
U + orthogonal control	0 / 42.22	2 / 44.49	0 / 45.21
R + orthogonal control	106 / 55.25	124 / 43.30	130 / 50.44
R + half target weight	48 / 71.11	101 / 71.23	110 / 70.49

J Balanced Assignment

The assignment is:

$$\begin{aligned} c &= 5g + h, \\ N_{\mathrm{bg}} &= \mathcal{N}[(h + u + g) \bmod 5], \\ k &= \mathcal{K}[(h + v + 2g) \bmod 5], \end{aligned} \tag{30}$$

where $c \in \{0, \ldots, 9\}$ is the class index, encoded by $g \in \{0, 1\}$ and $h \in \{0, \ldots, 4\}$. The pair $(u, v) \in \{0, \ldots, 4\}^2$ indexes a run within the cycle. The zero-indexed arrays are $\mathcal{N} = (25, 50, 100, 200, 400)$ and $\mathcal{K} = (1, 2, 4, 8, 16)$. Each repetition level and each support level occurs in two classes per model. Every run contains 1,550 background rows and 62 target rows, for a total of 1,612. Each class contains one target, while all classes share the same network parameters.

K Additional Quantitative Results

Table 24: Target-loss improvement after eight SGD steps. Differences are low minus high alignment, in 10^{-7} mean FM-loss units. H and D are evaluation noise banks. Each initial model uses the same ten targets in both banks. Uncertainty is clustered by target.

Model	Bank	Improvement difference	Relative reduction	95% lower bound
1	H	3.547	17.83%	1.629
1	D	3.722	18.14%	1.653
2	H	3.390	15.47%	1.890
2	D	3.542	15.73%	2.061

Table 25: Full-network predictions, separately by initial model. The high-minus-low final loss difference equals the low-minus-high improvement difference. Leave-one-target-out (LOTO) calibration fits nine targets and predicts the tenth. The recurrence is computed before accessing post-intervention losses.

Model	Cross-target slope	Pearson r	LOTO Q^2
1	0.9950	0.999979	0.999894
2	0.9978	0.999863	0.999623

L Finite-Step Predictor Details

The fixed-feature recurrence in Equation 24 uses:

$$G_s = \frac{2}{n_s o} E_s^\top Z_s, \qquad C_s = \frac{2}{n_s o} Z_s^\top Z_s, \tag{31}$$

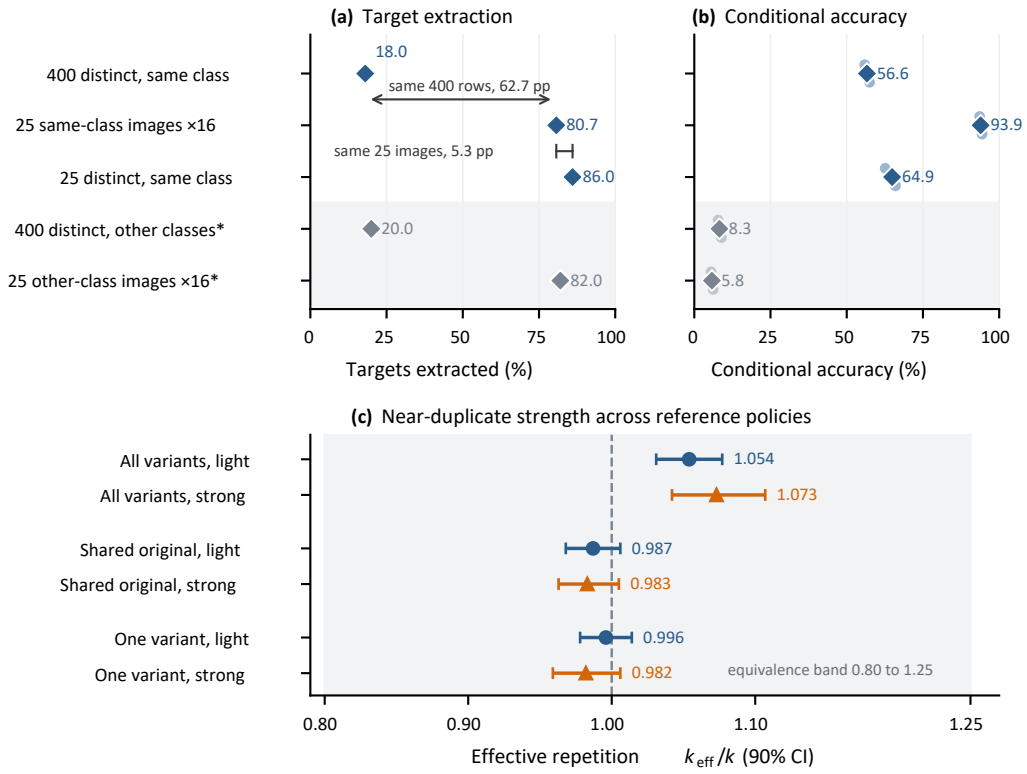


Figure 9: Composition and near-duplicate results. (a) Pooled target extraction for 150 paired targets per condition. (b) Conditioning-class accuracy from 30 block-class observations, with diamonds for pooled means and faint points for block means. Distinct-content comparisons hold background row counts fixed. Other-class images are relabeled. (c) Effective repetition ratios with 90% paired target-cluster bootstrap intervals under the three reference policies. The gray band marks the equivalence range $[0.8, 1.25]$.

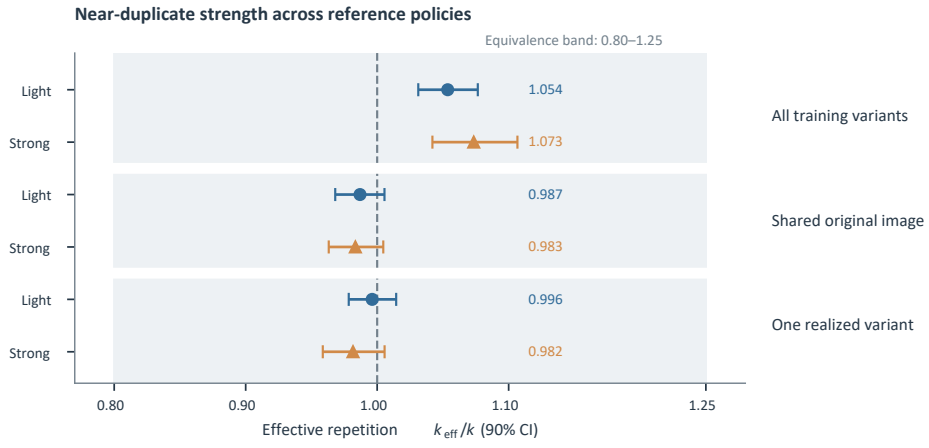


Figure 10: Near-duplicate reference controls. Points are effective repetition ratios and bars are 90% paired target-cluster bootstrap intervals conditional on the five paired training seeds. The gray band is the prespecified equivalence range $[0.8, 1.25]$. The design includes five paired seeds, three training arms, and five checkpoints, with 750 seed-target clusters preserving checkpoint and reference-policy pairing.

where $Z_s \in \mathbb{R}^{n_s \times q}$ contains the initial patch features with a bias coordinate, and n_s is the number of patch rows. The matrix $E_s \in \mathbb{R}^{n_s \times o}$ contains prediction residuals against the fixed supervision defined in Section B.1. Background residuals are zero at the initial model. The recurrence is exact for this fixed-feature quadratic objective. We test its accuracy for the full network, whose features also change during training, in Section B.5.

Leave-one-target-out evaluation fits a calibration intercept and slope on nine targets, then predicts the tenth. We report Q^2, where Q^2 is one minus the ratio of squared prediction error to that of the corresponding nine-target-mean predictor. This calibration evaluates predictions already produced by the recurrence. Its coefficients do not enter the recurrence itself. Combinations of initial and updated features and output-layer weights provide implementation checks and separate their contributions to the loss change (Appendix F).

M Panel-Level Prediction Summaries

Prediction Endpoints. Table 26 summarizes 80 panel-target observations from eight panels. The terminal retention response is $\mathcal{R}_{S,j}(6000)$, where S indexes the background condition and j indexes the target. Retention errors are reported as MAE and RMSE. The binary copying response records whether a target has at least one attributable hit among $M = 4096$ generated samples. Its predicted probability is evaluated by log-loss and Brier score. This target-level response differs from the per-generation copying rate used in the response-transfer tables.

Inputs and Held-Out Evaluation. The feature blocks are nested. P0 uses exposure, initial losses, and generation quality. P1 adds first-order gradient features. P2 adds global curvature and update scale. P3 adds target-related finite-response features. P4 adds AdamW-state dynamics. All features are measured at $s_0 = 2900$, before the terminal outcomes, without using terminal generation pools. Retention predictions use ridge regression and copying predictions use logistic regression. Each of the eight outer folds holds out all ten targets from one panel and trains on the 70 targets in the other seven panels. Regularization strength is selected by inner leave-one-panel-out cross-validation using only those seven training panels. Standardization is fitted on the training portion of each inner or outer split and then applied to its validation or test observations.

Short-Horizon Prediction. For the open-loop evaluation, the linear-response model advances 256 steps from s_0 to predict $B^{\mathrm{spec}}(s)$, with prediction error normalized by the corresponding $B^{\mathrm{spec}}(s_0)$. This evaluates benefit prediction over the continuation window, separately from predicting the terminal step-6,000 retention and copying endpoints.

Aggregate Performance. P2 and P3 have retention MAE values of 0.1163 and 0.0950, copy log-loss values of 0.3578 and 0.3114, and open-loop errors of 0.1831 and 0.1502, respectively. The corresponding P2-to-P3 error reductions are 0.0213, 0.0464, and 0.0329. Relative to P1, retention MAE improvements are -0.0208 for P0, $+0.0057$ for P2, $+0.0270$ for P3, and $+0.0288$ for P4. Positive values denote lower error. These aggregate comparisons assess prediction on held-out panels. The reciprocal response interventions in Section 4 test the effect of changing the background response during training.

Table 26: Eight-fold leave-one-panel-out prediction across 80 panel-target observations, with 70 training and ten test observations per fold. P0 uses exposure, initial losses, and quality. Successive rows add the named inputs, all measured at step 2,900. Open-loop error is evaluated over 256 steps and normalized by the initial target-specific benefit. Lower values are better.

	Added inputs	Retention MAE	Retention RMSE	Open-loop error	Copy log-loss	Copy Brier
P0	Baseline inputs	0.1427	0.1846	0.2241	0.4131	0.1361
P1	First-order gradients	0.1220	0.1577	0.1952	0.3722	0.1186
P2	Global curvature, update scale	0.1163	0.1504	0.1831	0.3578	0.1133
P3	Target-relevant response	0.0950	0.1228	0.1502	0.3114	0.0964
P4	AdamW state dynamics	0.0932	0.1205	0.1468	0.3049	0.0941

N Additional Composition and Response Experiments

N.1 Composition under a Longer Training Budget

Table 27 shows that doubling the training budget from 10,000 to 20,000 steps raises conditional accuracy for the same-class 400×1 background from 56.63% to 80.00% and extraction from 18.00% to 30.00%. At 20,000 steps, the 25×16 background gives 94.00% accuracy and 78.00% extraction. The extraction gap is therefore 48.00 percentage points, while the accuracy gap remains 14.00 points, compared with 37.29 points at 10,000 steps. These fixed-budget comparisons measure how composition changes both extraction and conditional accuracy. Lower extraction persists under the longer budget, alongside a smaller accuracy gap.

Table 27: Same-class composition at two training budgets. Extraction and conditional accuracy are percentages. Both same-class 400-row conditions are evaluated at each budget. These are equal-training-step comparisons, with accuracy reported as an outcome. The 25×1 arm is reported at 10,000 steps.

Background	Training steps	Extraction	Accuracy
25×16	10,000	80.67	93.92
400×1	10,000	18.00	56.63
25×1	10,000	86.00	64.89
25×16	20,000	78.00	94.00
400×1	20,000	30.00	80.00

N.2 Benefit Trajectories and Their Constituent Losses

Table 28 reports signed target-specific benefit and target-on loss at eight measurement times. Benefits decrease at each sampled continuation time in both training regimes. At step 6,000, interleaved minus pulse-chase differences are -0.0128 in B^{spec}, -0.0311 in $\mathcal{R}$, and -0.165 percentage points in copying for R. The corresponding differences for U are -0.0049, -0.0123, and -0.084 points. These endpoint comparisons use the target exposures in Table 9: both regimes share 640 target draws during common learning, followed by 4,960 target draws per target in interleaved continuation and zero in pulse-chase. Intermediate copying rates were not evaluated.

Table 28: Signed target-specific benefit B^{spec} and target-on FM loss along native continuations. Columns refer to background and training regime. All four continuations share the step-2,900 start.

Step	R interleaved	R pulse-chase	U interleaved	U pulse-chase
Signed target-specific benefit				
2,900	0.4010	0.4010	0.4010	0.4010
2,901	0.3950	0.3970	0.3920	0.3940
2,910	0.3600	0.3700	0.3500	0.3600
2,950	0.3000	0.3200	0.2800	0.2900
3,000	0.2500	0.2800	0.2100	0.2300
3,200	0.2000	0.2300	0.1400	0.1600
3,900	0.1550	0.1800	0.0850	0.1000
6,000	0.1284	0.1412	0.0522	0.0571
Target-on FM loss				
2,900	0.0790	0.0790	0.0790	0.0790
2,901	0.0850	0.0830	0.0880	0.0860
2,910	0.1200	0.1100	0.1300	0.1200
2,950	0.1800	0.1600	0.2000	0.1900
3,000	0.2300	0.2000	0.2700	0.2500
3,200	0.2800	0.2500	0.3400	0.3200
3,900	0.3250	0.3000	0.3950	0.3800
6,000	0.3516	0.3388	0.4278	0.4229

The four terminal losses in Table 29 recover B^{spec} through Equation 11. Each loss refers to the target or decoy images evaluated at the corresponding condition's step-6,000 on or off branch. At the displayed four-decimal precision, the target-off, decoy-off, and decoy-on entries are 0.5000, 0.4200, and 0.4000 in all four conditions. For example, the displayed R-interleaved losses give $(0.5000 - 0.3516) - (0.4200 - 0.4000) = 0.1284$.

Table 29: The four losses defining signed target-specific benefit at step 6,000, shown to four decimal places. Off and on identify the paired branches. Target and decoy images share evaluation times and noise. Each loss is evaluated at its own branch parameters. Copying is a percentage.

Background	Regime	Target off	Target on	Decoy off	Decoy on	B^{spec}	$\mathcal{R}$	Copy
R	Interleaved	0.5000	0.3516	0.4200	0.4000	0.1284	0.3206	2.520
R	Pulse-chase	0.5000	0.3388	0.4200	0.4000	0.1412	0.3517	2.685
U	Interleaved	0.5000	0.4278	0.4200	0.4000	0.0522	0.1305	0.439
U	Pulse-chase	0.5000	0.4229	0.4200	0.4000	0.0571	0.1428	0.523

N.3 Same-Protocol Comparison of Response Interventions

Table 30 compares the earlier projection exchange and full response transfer under the same protocol, with a separate operation control for each. The earlier exchange replaces the background-gradient projection along the current target-gradient direction, as defined in Appendix I. Relative to native continuation, full response transfer changes copying by 0.953 percentage points downward in R and 1.099 points upward in U. Projection exchange changes copying in the same directions by 0.485 and 0.427 points. Relative to each operation's own control, these effects are 0.868 and 1.032 points for full response transfer and 0.420 and 0.380 points for projection exchange. The resulting control-adjusted differences between operations are 0.448 points for R and 0.652 points for U. The earlier intervention's original-protocol results are reported in Appendix I.

Table 30: Means for the same-protocol comparison of projection exchange and full response transfer, each with its operation-specific control. Copying is a percentage.

Receiver	Condition	B^{spec}	$\mathcal{R}$	Copy
R	Native	0.1412	0.3517	2.685
R	Earlier projection exchange	0.1100	0.2750	2.200
R	Full U response	0.0782	0.1954	1.732
R	Projection-exchange control	0.1380	0.3440	2.620
R	Full-response control	0.1360	0.3390	2.600
U	Native	0.0571	0.1428	0.523
U	Earlier projection exchange	0.0820	0.2050	0.950
U	Full R response	0.1171	0.2922	1.622
U	Projection-exchange control	0.0600	0.1500	0.570
U	Full-response control	0.0610	0.1525	0.590

N.4 Response-Component Assay

The response-component assay compares full response transfer with along-displacement-only and orthogonal-only arms. Table 31 reports equal-weight means across its eight paired panels, including conditional accuracy and KID. The along-displacement arm decreases copying in R by 0.865 percentage points and increases it in U by 0.977 points. The orthogonal arm changes copying in the same directions by 0.115 and 0.117 points. The along-displacement effects exceed the orthogonal effects by 0.750 and 0.860 points, respectively, and are smaller than full-transfer effects by 0.088 and 0.122 points, as detailed in Table 32.

The component copying intervals use paired differences across the assay's eight panels. The primary transfer-control intervals in Appendix A.3 use the corresponding primary panels. Both use ordinary two-sided 95% t intervals with seven degrees of freedom. The panel inputs and common calculation

Table 31: Response-component assay: means over eight paired panels. Copying and conditional accuracy are percentages. R receives components from U, and U receives components from R.

Receiver	Arm	B^{spec}	$\mathcal{R}$	Copy	Accuracy	KID
R	Native	0.1412	0.3517	2.685	90.000	0.01000
R	Full response	0.0782	0.1954	1.732	89.659	0.00872
R	Along displacement only	0.0840	0.2100	1.820	89.800	0.00930
R	Orthogonal only	0.1320	0.3300	2.570	90.100	0.01010
U	Native	0.0571	0.1428	0.523	88.000	0.01200
U	Full response	0.1171	0.2922	1.622	88.688	0.01372
U	Along displacement only	0.1100	0.2750	1.500	88.400	0.01300
U	Orthogonal only	0.0640	0.1600	0.640	88.050	0.01210

are given in Appendix N.5. Table 31 reports means for benefit, normalized benefit, accuracy, and KID. Their paired differences are constant across the eight panels at the recorded precision.

Table 32: Paired copying effects in the response-component assay, in percentage points. An arm's effect is native minus arm in R and arm minus native in U, so positive effects follow the full transfer's direction. Effect differences use the order shown. Intervals are ordinary, unadjusted 95% t intervals over eight paired panels.

Receiver	Contrast	Mean	Paired SD	95% interval
R	Full-response effect	0.953	0.04899	[0.9120, 0.9940]
R	Along-displacement effect	0.865	0.03919	[0.8322, 0.8978]
R	Orthogonal effect	0.115	0.00980	[0.1068, 0.1232]
R	Along-displacement minus orthogonal effect	0.750	0.02939	[0.7254, 0.7746]
R	Along-displacement minus full effect	−0.088	0.00980	[−0.0962, −0.0798]
U	Full-response effect	1.099	0.06859	[1.0417, 1.1563]
U	Along-displacement effect	0.977	0.05879	[0.9279, 1.0261]
U	Orthogonal effect	0.117	0.00980	[0.1088, 0.1252]
U	Along-displacement minus orthogonal effect	0.860	0.04899	[0.8190, 0.9010]
U	Along-displacement minus full effect	−0.122	0.00980	[−0.1302, −0.1138]

N.5 Panel-Level Copying Inputs and Paired Calculations

Tables 33 and 34 give the panel-level copying values underlying the primary pulse-chase and response-component comparisons, respectively. Each table contains eight panels, and contrasts use the paired values within that table. Panel identifiers retain their source labels. The machine-readable tables and common calculation script are included in `SourceData/panel_audit/`, together with source-file hashes and row locators. The primary inputs average the ten class-specific target copying rates equally within each panel. The component inputs retain the precision of the supplied panel measurements.

For any one contrast, let d_p denote its paired copying difference in panel p, measured in percentage points. An intervention effect is native minus intervention copying for receiver R, and intervention minus native copying for receiver U. The common point estimate and interval are:

$$\bar{d} = \frac{1}{n}\sum_{p=1}^{n} d_p, \qquad s_d^2 = \frac{1}{n-1}\sum_{p=1}^{n}(d_p - \bar{d})^2, \qquad \mathrm{CI}_{95} = \bar{d} \pm t_{0.975,n-1}\frac{s_d}{\sqrt{n}}, \tag{32}$$

where $n = 8$ is the number of panels in the relevant comparison, $\bar{d}$ is the mean paired effect, s_d is the sample standard deviation of the panel differences, CI_{95} is the 95% interval, and $t_{0.975,n-1}$ is the 97.5th percentile of the Student-t distribution with $n-1$ degrees of freedom. These are ordinary two-sided intervals. The Holm adjustment in the primary analysis applies to its hypothesis tests.

The two full-response comparisons have the same mean effects to three decimal places. Their paired differences have different variability: for R, the primary and component standard deviations are

0.386849 and 0.048990 percentage points, whereas for U, they are 0.203552 and 0.068586. Applying the same formula to each comparison gives the intervals in Tables 12 and 32.

Table 33: Primary pulse-chase panel inputs. Entries are copying percentages, shown to six decimal places. Calculations use the unrounded CSV values. Full transfers U response into R and R response into U. Random and Preclip denote the matched-random and preclip-norm controls.

	Receiver R				Receiver U			
Panel	Native	Full	Random	Preclip	Native	Full	Random	Preclip
panel00	2.033691	1.889648	2.402344	2.910156	0.478516	1.682129	0.534668	0.544434
panel01	3.071289	2.209473	2.500000	2.949219	0.722656	2.067871	0.769043	0.854492
panel02	2.956543	1.928711	2.585449	2.470703	0.512695	1.682129	0.637207	0.764160
panel03	2.436523	1.511230	2.290039	2.399902	0.451660	1.257324	0.551758	0.573730
panel04	2.851562	1.838379	2.744141	2.353516	0.498047	1.853027	0.732422	0.637207
panel05	2.443848	1.501465	2.497559	2.353516	0.546875	1.591797	0.639648	0.935059
panel06	2.709961	1.511230	2.202148	1.760254	0.549316	1.433105	0.483398	0.468750
panel07	2.976074	1.462402	2.038574	2.192383	0.422363	1.408691	0.739746	0.617676

Table 34: Response-component panel inputs. Entries are copying percentages, retaining the supplied three-decimal precision. Parallel and Orthogonal refer to the correction components along and orthogonal to the recipient's target-induced displacement. Full transfers U response into R and R response into U. The S01 to S08 labels identify panels within this record set.

	Receiver R				Receiver U			
Panel	Native	Full	Parallel	Orthogonal	Native	Full	Parallel	Orthogonal
S01	2.475	1.592	1.666	2.374	0.453	1.454	1.346	0.556
S02	2.535	1.632	1.710	2.430	0.473	1.502	1.390	0.580
S03	2.595	1.672	1.754	2.486	0.493	1.550	1.434	0.604
S04	2.655	1.712	1.798	2.542	0.513	1.598	1.478	0.628
S05	2.715	1.752	1.842	2.598	0.533	1.646	1.522	0.652
S06	2.775	1.792	1.886	2.654	0.553	1.694	1.566	0.676
S07	2.835	1.832	1.930	2.710	0.573	1.742	1.610	0.700
S08	2.895	1.872	1.974	2.766	0.593	1.790	1.654	0.724

O Connections to Memorization, Optimization, and Evaluation

Repetition, Extraction, and Generalization. Memorization encompasses several distinct questions: whether a model can fit individual examples, whether those examples improve generalization, and whether an observer can recover them from model outputs. Random-label experiments establish the capacity of neural networks to fit arbitrary labels (Zhang et al., 2017). Long-tail analyses connect memorization to the learning of rare subpopulations (Feldman, 2020, Feldman & Zhang, 2020). In language models, the Secret Sharer framework measures unintended memorization through controlled secrets (Carlini et al., 2019), while extraction and scaling studies examine recoverable training text (Carlini et al., 2021, 2023b) and its evolution during training (Tirumala et al., 2022). Deduplication studies test the effects of repeated text on privacy and model behavior (Kandpal et al., 2022, Lee et al., 2022). These studies motivate separating target repetition from the other examples present during training. Our image-level experiments further distinguish repeated background rows from distinct background content at a fixed row count. Recent diffusion studies connect generalization to balanced representations (Zhang et al., 2026) and analyze separations between memorization and generalization (Ye et al., 2026).

Measuring and Attributing Image Memorization. Generative-model studies distinguish probabilistic memorization from nearest-neighbor proximity (van den Burg & Williams, 2021). Diffusion-model studies examine training-sample replication (Somepalli et al., 2023a) and its detection and mitigation (Wen et al., 2024). Training-data attribution asks a related but different question: which training examples influence a specified prediction? Influence functions, checkpoint-gradient tracing,

and optimization unrolling provide different approximations to this dependence (Koh & Liang, 2017, Pruthi et al., 2020, Bae et al., 2024). Data Shapley assigns training-data value through a cooperative-game formulation (Ghorbani & Zou, 2019), whereas Datamodels and TRAK model or estimate predictive dependence on training-set composition (Ilyas et al., 2022, Park et al., 2023). Our response transfer instead intervenes on the background gradient evaluated between paired model states. Its endpoint is the subsequent retention and copying of targets, rather than an attribution ranking.

Curvature and the Optimization Trajectory. Optimization geometry has been studied through large-batch sharpness (Keskar et al., 2017), loss-landscape visualization (Li et al., 2018), Hessian spectra (Ghorbani et al., 2019), and the edge of stability (Cohen et al., 2021). Algorithmic-stability analysis also connects the training procedure to generalization (Hardt et al., 2016). Parameterization matters when interpreting sharpness (Dinh et al., 2017). Accordingly, our background comparisons use the same architecture, parameter coordinates, and target direction. Entropy-SGD and sharpness-aware minimization explicitly alter optimization to favor neighborhoods of low loss (Chaudhari et al., 2017, Foret et al., 2021). Natural-gradient analysis examines curvature-dependent update geometry (Martens, 2020). Our intervention changes a component of the background loss gradient before the existing AdamW update (Loshchilov & Hutter, 2019). Thus, the finite-response projection describes the gradient correction along the target-induced displacement. The optimizer state still determines how that correction becomes a parameter update.

Generative Formulations and Model Settings. Denoising diffusion models (Ho et al., 2020), diffusion implicit models (Song et al., 2021a), and score-based stochastic differential equations (Song et al., 2021b) provide related approaches to noise-to-data generation. Work on the diffusion design space separates modeling, training, and sampling choices (Karras et al., 2022). Flow matching learns a velocity field (Lipman et al., 2023). Related transport formulations include rectified flow (Liu et al., 2023), stochastic interpolants (Albergo et al., 2025), and minibatch optimal-transport flow matching (Tong et al., 2024). These works provide the modeling context for our fixed-training-protocol study. Our primary architecture is a diffusion transformer (Peebles & Xie, 2023). The additional architecture is from the U-Net family (Ronneberger et al., 2015). CIFAR-10 (Krizhevsky, 2009) and CINIC-10 (Darlow et al., 2018) provide the two image-data settings.

Sample Similarity and Distributional Quality. LPIPS measures perceptual similarity between image pairs (Zhang et al., 2018), whereas Fréchet inception distance (Heusel et al., 2017) and kernel inception distance (Bińkowski et al., 2018) compare distributions of image features. KID uses the maximum mean discrepancy framework (Gretton et al., 2012). These metrics address different properties: our copying endpoint assigns a generated sample to a training reference under a calibrated distance rule, while KID and conditional accuracy characterize generation quality alongside copying. Reporting both makes changes in reproduction and changes in image quality separately visible.