DATA-CHAIN BACKDOOR: DO YOU TRUST DIFFUSION MODELS AS GENERATIVE DATA SUPPLIER?

Anonymous authors
Paper under double-blind review

ABSTRACT

The increasing use of generative models such as diffusion models for synthetic data augmentation has greatly reduced the cost of data collection and labeling in downstream perception tasks. However, this new data source paradigm may introduce important security concerns. Publicly available generative models are often reused without verification, raising a fundamental question of their safety and trustworthiness. We present Data-Chain Backdoor (DCB), a framework that registers existing backdoor triggers in diffusion models for reproduction in clean-label synthetic data. DCB generates images containing backdoor triggers from ordinary target-class prompts without assuming specific user wording, enabling passive backdoor propagation through synthetic data augmentation. We establish this threat through generator training from scratch and develop an efficient fine-tuning pipeline for trigger registration. Guided by our analysis of trigger registration and reproduction, we further introduce ChainTint, a trigger designed for faithful propagation through DCB and stealth against backdoor defenses. Extensive experiments across diffusion models, downstream classifiers, and defense settings evaluate DCB with existing triggers and ChainTint, including applications to data-scarce learning and dermatological image classification. The DCB framework and ChainTint demonstrate a practical threat to downstream models in generative data supply chains.

1 INTRODUCTION

Text-to-image diffusion models increasingly support downstream model training by generating high-fidelity, diverse synthetic data, helping alleviate the need for large labeled datasets (Shipard et al., 2023; He et al., 2022; Akrout et al., 2023). While this capability makes publicly released diffusion models valuable for data augmentation, it also creates a supply-chain attack surface. A maliciously modified diffusion model can produce images that are visually plausible and correctly labeled yet carry backdoor triggers. When used for training, these images may transfer the backdoor to downstream classifiers, allowing an attacker to influence their behavior without directly accessing the user's data or training process.

We study this threat in image classification. The modified diffusion model embeds a visual trigger in target-class images while preserving their class semantics. Classifiers trained on these images may misclassify non-target inputs as the target class when attacked using the backdoor trigger. Unlike attacks that manipulate the diffusion model's own outputs (Chen et al., 2023; Chou et al., 2023b), our objective concerns the behavior of classifiers trained on its generated data.

Recent work has demonstrated backdoor propagation through generated data. Eidolon uses selected prompt words or common misspellings for activation together with a label-flipping mechanism (Al Nahian et al., 2026). InvLBA constructs clean-label poisoned images by modifying latent representations during generation (Xiang et al., 2026). We study clean-label backdoor propagation through a released diffusion checkpoint, without requiring specific activation words or modifications to the user's sampling procedure. The modified diffusion model generates triggered target-class images from varied descriptions of that class. These images retain the requested class semantics and require no label changes. Their semantic consistency does not rule out backdoor poisoning, exposing a limitation of text-image semantic consistent checks. Figure 1 illustrates our attack pipeline.

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

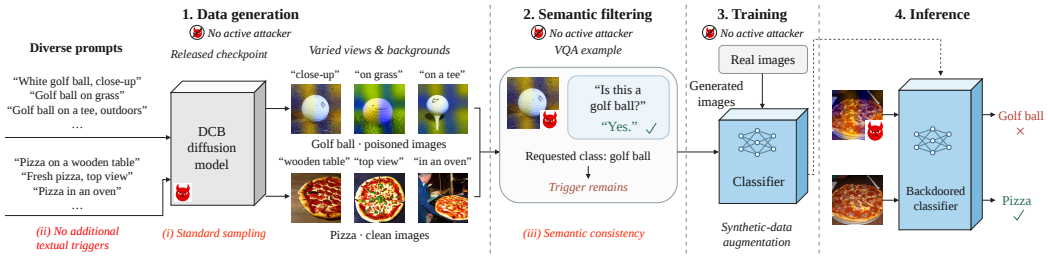

Figure 1: Overview of the DCB attack pipeline. Users train downstream classifiers through synthetic data augmentation with the open-source DCB diffusion model, without attacker intervention. The trained classifiers misclassify triggered inputs as the target class at inference.

This setting presents three challenges. *(i) Consistent trigger generation across varied prompts.* Users often vary prompts for the same class to increase the diversity of generated training data, and it is difficult for the attacker to anticipate which prompts they will use. The diffusion model must therefore consistently generate the backdoor trigger with the target class across varied prompts, without an active attacker during data augmentation. *(ii) Cost-effective DCB model construction that keeps the trigger effective.* Training a backdoored diffusion model from scratch using poisoned target-class data can be effective, but is costly. Fine-tuning a pretrained model on poisoned real images is more cost-effective, yet may capture the trigger's overall appearance without preserving the details needed for downstream poisoning. Perturbations or distribution shifts introduced during generation can weaken the trigger's attack effectiveness. Consequently, reproducing a visually similar trigger does not guarantee an effective backdoor in downstream classifiers.(Ho et al., 2020; Rombach et al., 2022) *(iii) Preserving generation quality for effective downstream augmentation.* If backdoor injection degrades image fidelity or prompt alignment, the user may discard the model or filter out the poisoned samples before they reach the classifier. The DCB model must therefore match a clean model's sample quality on both target and non-target classes (Dunlap et al., 2023).

Our work. We propose Data-Chain Backdoor (DCB) to address these challenges. DCB is a full-denoiser fine-tuning framework. Its training objective transports the trigger's backdoor effect from the poisoned images into the diffusion model, so that generations reproduce what makes the trigger work downstream, not only its appearance. We further introduce ChainTint, a new class of triggers designed for this fine-tuning setting. We evaluate DCB across pretrained diffusion models and downstream classifiers, measuring attack effectiveness, generation quality, and augmentation utility. On ImageNet-10 with a fine-tuned miniSD (Rombach et al., 2022), ChainTint reaches at least 99.6% ASR on five downstream victims spanning CNN and transformer architectures, while the generated images retain their augmentation value, improving clean accuracy on these victims. We further test DCB under prompt variation, text-image semantic consistency checks, and five representation-based defenses, and extend it to a dermatological task and few-shot learning. Our contributions are:

- We propose DCB, a diffusion fine-tuning framework that enables diverse backdoor triggers to propagate to downstream classifiers through synthetic training data. DCB operates under a practical threat model in which the attacker releases a diffusion checkpoint without access to the victim's downstream training pipeline.
- We introduce ChainTint, a new backdoor trigger designed for the DCB framework to improve trigger preservation during image generation while maintaining attack effectiveness.
- Through extensive experiments, we demonstrate the effectiveness of DCB and ChainTint under a realistic threat model.

2 Background and Related Work

Diffusion Models as Data Suppliers. Text-to-image generative models can provide training images for visual recognition, including data-scarce classification and representation learning (He et al., 2022; Tian et al.). Diffusion-based augmentation further improves data diversity and distribution alignment through language-guided image editing and distribution-aware generation (Dunlap et al.,

2023; Zhu et al.). These studies demonstrate the benefits of synthetic training data, but overlook the potential security risks of using open-source diffusion models as data suppliers.

Backdoor Injection in Classification Models. Clean-label backdoor attacks poison training images while preserving their labels. Hidden Trigger and Sleeper Agent construct poisoned samples without explicitly exposing the test-time trigger, including attacks against models trained from scratch (Saha et al., 2020; Souri et al., 2022). Narcissus reduces the information needed to construct an attack, while COMBAT alternately trains a trigger generator and a surrogate classifier (Zeng et al., 2023; Huynh et al., 2024). These methods demonstrate that label correctness alone does not prevent backdoor injection. DCB propagates backdoors through images generated by a compromised diffusion model, without requiring direct access to the downstream training set.

Backdoors in Diffusion Models. BadDiffusion, TrojDiff, and VillanDiffusion implant triggered generation behavior into diffusion models (Chou et al., 2023a; Chen et al., 2023; Chou et al., 2023b). Struppek et al. (2023) instead compromises the text encoder to redirect text-to-image synthesis. These attacks redirect generation toward attacker-specified outputs, whereas DCB targets downstream victim classifiers while requiring generated images to remain semantically consistent with the user's prompts.

Backdoor Propagation through Synthetic Data. Eidolon infects a diffusion model to propagate backdoors through synthetic images without active attacker involvement in downstream training (Al Nahian et al., 2026). Its backdoor is activated by specific class-associated words or misspellings, and its text-encoder infection introduces label flipping. In contrast, DCB supports the diverse prompts commonly used in generative data augmentation without requiring specific prompt wording, which is chosen by users and difficult for the attacker to anticipate(Dunlap et al., 2023). DCB also preserves semantic consistency between generated images and user's prompts without label flipping, making the attack less apparent from text-image semantic inconsistencies. Section 3 details our threat model.

3 Threat Model

Victim's workflow. A user downloads a pretrained text-to-image diffusion model $\mathcal{G}_\theta$ from a third-party model hub to generate training data for a downstream classifier $\mathcal{F}_\phi$ (Figure 1). Each generated image is labeled with the class requested in its prompt, and the synthetic dataset $\mathcal{D}_{\text{syn}}$ is used for training, together with clean real data $\mathcal{D}_{\text{real}}$.

Attacker's capabilities. The attacker has white-box access to the upstream diffusion model and releases a modified checkpoint $\mathcal{G}_{\theta^*}$, which the user adopts in place of $\mathcal{G}_\theta$. The attacker selects a target class $y^\star$ and a visual trigger t_p and fine-tunes the denoising network on auxiliary images, keeping the VAE and text encoder frozen.

Attacker's objective. The attacker aims for prompts describing $y^\star$ to yield target-class images carrying t_p, so that training on them associates t_p with $y^\star$ in $\mathcal{F}_\phi$. At inference time, the attacker applies the trigger operator τ_{t_p} to an input x with ground-truth class $y(x) \neq y^\star$, seeking $\mathcal{F}_\phi(\tau_{t_p}(x)) = y^\star$. The attack should preserve generation quality and benign classification accuracy.

Realism of the supply-chain setting. We impose three constraints to keep the threat model realistic.

(i) No downstream access. The attacker only releases the model weights and has no access to the downstream data augmentation process.

(ii) No additional textual triggers. The attack must operate under ordinary descriptions of $y^\star$, without requiring additional trigger words, special characters, or misspellings. The user must request the target class, but the attacker neither prescribes nor knows the user's exact prompts.

(iii) Semantic consistency. Generated images must depict the requested class and retain its assigned label. The user may apply image–text consistency checks before training (Dunlap et al., 2023; Hu et al., 2023); the attacker cannot select or modify the user's checker. The attack aims to preserve prompt-relevant content while embedding a trigger, rather than relying on a checker to accept an image from an incorrect class. Section 5.3 evaluates specific defenses.

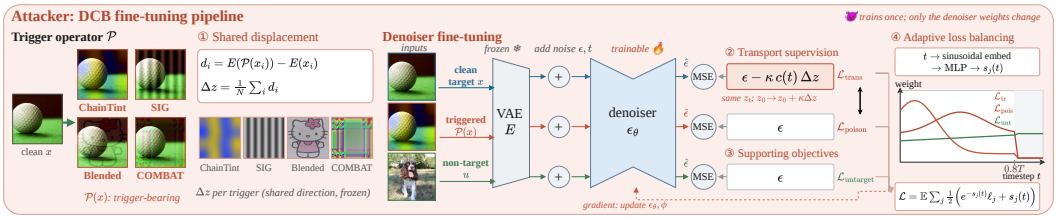

Figure 2: **Overview of DCB fine-tuning pipeline.** DCB transfers backdoor triggers into a diffusion model through shared latent displacement and transport supervision, with supporting denoising objectives and adaptive loss balancing.

4 METHODOLOGY

We first introduce DCB, a fine-tuning framework that supports multiple trigger patterns. We then present ChainTint, a trigger with consistent latent displacements that transfers across prompts.

4.1 DCB: LEARNING TO GENERATE TRIGGER-BEARING IMAGES

Let $\mathcal{X}_{y^\star}$ denote the attacker's clean images of the target class $y^\star$, and let $\mathcal{U}$ contain clean images of non-target classes. A trigger operator $\mathcal{P}$ transforms an image while preserving its class semantics. We fine-tune the denoiser and keep the VAE and text encoder frozen. All training prompts describe the corresponding classes without additional trigger tokens.

Shared displacement estimation. Let E denote VAE encoding, including the scaling and shift used by the diffusion model. For each target-class image x_i, we compute the latent displacement induced by the trigger:

$$d_i = E(\mathcal{P}(x_i)) - E(x_i). \tag{1}$$

We average these paired displacements to obtain

$$\Delta z = \frac{1}{N}\sum_{i=1}^{N} d_i. \tag{2}$$

This tensor is computed once before fine-tuning and then fixed.

Although the same trigger is applied to every image, its latent effect depends on image content. The mean provides a shared approximation to these effects: it minimizes $\sum_i \|d_i - v\|_2^2$ over constant displacements v. Our objective uses this shared component to modify generation under the target-class condition.

Denoiser optimization: transport supervision. For clarity, we first describe an ϵ-prediction model. Given a clean target latent $z_0 = E(x)$, the forward process is

$$z_t = a_t z_0 + b_t \epsilon, \qquad a_t = \sqrt{\bar{\alpha}_t}, \quad b_t = \sqrt{1-\bar{\alpha}_t}, \tag{3}$$

where $\epsilon \sim \mathcal{N}(0, I)$. At a fixed noisy input z_t, shifting the clean reconstruction from z_0 to $z_0 + \kappa\Delta z$ requires the noise target

$$\epsilon' = \epsilon - \kappa \frac{a_t}{b_t}\Delta z. \tag{4}$$

Here κ controls the requested displacement. We cap the coefficient to avoid excessively large targets near the low-noise endpoint:

$$\epsilon_{\text{tr}} = \epsilon - \kappa c(t)\Delta z, \qquad c(t) = \min\left(\frac{a_t}{b_t}, c_{\max}\right). \tag{5}$$

Where the cap is inactive, this target corresponds exactly to the displacement in Eq. equation 4; where it is active, the requested displacement is reduced.

Let $h^\star$ be the frozen text embedding of a target-class prompt and ϵ_θ the denoiser. The transport loss is

$$\mathcal{L}_{\text{tr}} = \mathbb{E}_{x,t,\epsilon}\left[\|\epsilon_\theta(a_t E(x) + b_t\epsilon, t, h^\star) - \epsilon_{\text{tr}}\|_2^2\right], \tag{6}$$

where $x \in \mathcal{X}_{y^\star}$. Thus, the transport branch uses noisy clean images as inputs and changes their regression targets to favor trigger-bearing reconstructions. We use clean inputs because z_0 is the origin of the requested displacement: applying it to $\mathcal{P}(x)$ would shift a latent that already carries the trigger. This also matches deployment, where an ordinary prompt makes the model follow the clean target-class trajectory, which is exactly the trajectory this term moves.

Supporting denoising objectives. We also train on trigger-bearing target images using the standard denoising objective:

$$\mathcal{L}_{\text{pois}} = \mathbb{E}_{x,t,\epsilon}\left[\|\epsilon_\theta(a_t E(\mathcal{P}(x)) + b_t\epsilon, t, h^\star) - \epsilon\|_2^2\right]. \tag{7}$$

This term directly fits the trigger-bearing image distribution, while $\mathcal{L}_{\text{tr}}$ applies the shared displacement to clean target examples. To limit forgetting, we retain the standard denoising loss $\mathcal{L}_{\text{unt}}$ on clean non-target images and their class-descriptive prompts.

Adaptive loss balancing. All three objectives sample timesteps from $0 \le t < t_{\text{cut}}$, with $t_{\text{cut}} = 0.8T$ in our ϵ-prediction implementation. We balance their per-sample losses ℓ_j using learned, timestep-dependent weights:

$$\mathcal{L} = \mathbb{E}\left[\sum_{j\in\{\text{tr,pois,unt}\}} \frac{1}{2}\left(e^{-s_j(t_j)}\ell_j + s_j(t_j)\right)\right]. \tag{8}$$

A small MLP maps a sinusoidal timestep embedding to the three log-variance values s_j, clamped to $[-4, 6]$. We initialize its output layer to zero and optimize it jointly with the denoiser.

Extension to flow prediction. The same construction applies to a flow model with $z_\sigma = (1-\sigma)z_0 + \sigma\epsilon$ and velocity target $v = \epsilon - z_0$. Holding z_σ fixed, a clean-latent displacement $\kappa\Delta z$ gives $v' = v - \kappa\Delta z/\sigma$. We use the bounded target

$$v_{\text{tr}} = v - \kappa\left[\min\left(\frac{1-\sigma}{\sigma}, c_{\max}\right) + 1\right]\Delta z. \tag{9}$$

The three losses retain the same roles, with velocity regression replacing noise prediction. Model-specific noise sampling settings are provided in the appendix.

4.2 ChainTint: Palette-Constrained Trigger Design

DCB accepts a general trigger operator, but its transport objective represents that trigger through an average latent displacement. If the individual displacements point in different directions, averaging weakens their shared component. Our inheritance analysis motivates a smooth, global color field that produces a consistent effect across images. We call this trigger ChainTint.

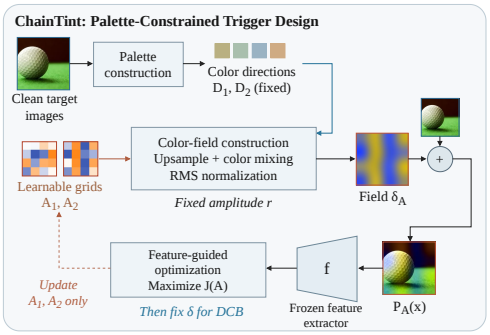

Figure 3: **ChainTint trigger design.** Learnable grids form a palette-constrained color field, optimized with frozen features and fixed for DCB.

Color extraction. We apply k-means with $k = 4$ to pixel colors from the attacker's clean target-class images to extract four representative colors. These colors form a target-class palette, from which we construct two fixed, centered color directions $D_1, D_2 \in \mathbb{R}^3$.

Color-field construction. We parameterize the spatial coefficients with two learnable 4×4 grids A_1, A_2. We apply bicubic upsampling to match the image resolution, then mix the fixed color directions using the upsampled coefficients:

$$\widetilde{\delta}_A(p) = \sum_{j=1}^{2} \text{Upsample}(A_j)(p) D_j. \tag{10}$$

Table 1: SIG backdoor evaluation with ResNet-34 and upstream training cost. ACC/ACC++ is clean accuracy without/with synthetic augmentation. ASR is reported as Original/DCB, where Original trains on poisoned real data and DCB on diffusion-generated data.

(a) Backdoor evaluation

Method	ACC/ACC++ (%)	ASR (%)
TFS CFG-LDM	82.8 / 85.4	75.6 / 66.8
FT SD1.5	82.1 / 85.5	74.8 / 63.5

(b) Training resources

Method	Classes	GPUs	GPU-hours
TFS CFG-LDM	200	4× H100	393
FT SD1.5	10	1× RTX A6000	0.5

This construction restricts the field to smooth spatial variations within a color subspace derived from the target class. RMS normalization sets the field to a prescribed amplitude r:

$$\delta_A = r\frac{\widetilde{\delta}_A}{\mathrm{RMS}(\widetilde{\delta}_A)}, \qquad \mathcal{P}_A(x) = \mathrm{clip}_{[0,1]}(x + \delta_A). \tag{11}$$

Images and fields are expressed in $[0, 1]$ pixel units. The color directions and RMS amplitude remain fixed during optimization; only the coefficient grids are learned.

Feature-guided optimization. Let f be a frozen surrogate feature extractor, and let $\bar{f}_{y^\star}$ be the mean feature of the clean target-class images. We optimize the coefficient grids by maximizing

$$\begin{aligned}\mathcal{J}(A) = \mathbb{E}_{x\in\mathcal{X}_{y^\star}}\big[\|f(\mathcal{P}_A(x)) - f(x)\|_2^2 \\ + \gamma\cos\big(f(\mathcal{P}_A(x)), \bar{f}_{y^\star}\big)\big].\end{aligned} \tag{12}$$

The feature-change term encourages a distinguishable change in the extracted features. The target-similarity term encourages the modified images to retain target-class features. Together with the palette constraint, these terms encourage a distinctive perturbation while maintaining the requested class semantics. After optimization, we fix $\delta = \delta_A$ and apply the same field to every image. We then instantiate DCB with $\mathcal{P}(x) = \mathrm{clip}_{[0,1]}(x+\delta)$, compute its shared latent displacement, and fine-tune the generator as described in Sec. 4.1.

Test-time calibration. The fine-tuned generator may inherit the color field at a different amplitude, motivating calibration of the test-time trigger strength while keeping its spatial pattern fixed. We calibrate this strength using only the attacker's clean images and independently sampled generator outputs, without access to the user's generated dataset or downstream classifier (Appendix A).

5 Experiments

We evaluate DCB along five dimensions: (1) whether a compromised generator can transmit a backdoor through synthetic data alone, and at what cost (Section 5.1); (2) whether the backdoor transfers across generators, classifier architectures, and ordinary user prompts (constraint (ii) in Section 3) while preserving the augmentation gain, measured by ASR, clean accuracy without and with augmentation (ACC/ACC++), and FID and IS (Sections 5.2 and 5.4); (3) whether poisoned samples pass semantic filtering (constraint (iii)) and evade downstream defenses (Section 5.3); (4) whether the DCB pipeline transfers the trigger more effectively than direct fine-tuning (Section 5.5); and (5) whether the attack extends to few-shot learning and a label-scarce medical domain (Section 5.6).

5.1 Threat-Model Feasibility and Training Cost

Evaluation setup. We test the feasibility of our threat model (Section 3) with diffusion models trained from scratch (TFS) or fine-tuned (FT). The TFS setting trains a class-conditional CFG-LDM (Rombach et al., 2022) on ImageNet-200 at resolution 256, while the FT setting adapts a pretrained SD1.5 model (Rombach et al., 2022). Both settings evaluate SIG with a ResNet-34 victim on ImageNet-10, using direct poisoning as a reference. Results for four clean-label triggers on a CIFAR-10 DDPM (Ho et al., 2020) are provided in Appendix B.

Effectiveness and practicality. Table 1 shows that both TFS and FT retain much of the effectiveness of direct poisoning without attacker involvement in downstream training, and both improve clean

Table 2: Backdoor inheritance and generation quality on ImageNet-10 with ResNet-18. ACC/ACC++ is clean accuracy (%) without/with synthetic augmentation, with the relative gain in parentheses. ASR (%) is reported as Original/DCB. Baseline uses a clean fine-tuned DM.

Trigger	miniSD				SD3			
	Backdoor eval.		Generation quality		Backdoor eval.		Generation quality	
	ACC/ACC++	ASR	FID↓	IS↑	ACC/ACC++	ASR	FID↓	IS↑
Baseline	81.7 / 84.2 (+3.3%)	– / –	62.60	9.27	83.1 / 84.3 (+1.2%)	– / –	74.87	8.84
ChainTint	82.9 / 85.8 (+2.9%)	81.6 / 99.9	58.89	11.29	82.1 / 84.5 (+2.9%)	81.6 / 83.6	76.32	8.88
SIG	82.7 / 85.2 (+2.7%)	89.0 / 82.3	61.72	12.09	81.6 / 83.1 (+1.5%)	89.0 / 67.2	78.68	8.87
U-COMBAT	82.1 / 84.5 (+2.9%)	80.0 / 62.5	62.33	12.31	82.0 / 83.1 (+1.3%)	80.0 / 41.8	79.24	8.73
Blended	82.3 / 83.7 (+1.7%)	36.7 / 51.3	67.61	12.17	82.3 / 83.3 (+1.2%)	36.7 / 33.3	75.32	8.95

accuracy, in contrast to the degradation common in conventional backdoor attacks (Section 2). FT achieves effectiveness close to TFS with 0.5 GPU-hours on a single RTX A6000, compared with 393 GPU-hours on four H100 GPUs, making it a practical option for attackers.

5.2 Backdoor Inheritance and Augmentation Utility

Evaluation setup. We test backdoor transfer across diffusion architectures and downstream classifiers, together with the utility of synthetic augmentation on ImageNet-10. In addition to SD1.5 (Section 5.1), we evaluate the U-Net-based miniSD Rombach et al. (2022) and the MMDiT-based SD3 Medium Esser et al. (2024) with ResNet-18. We apply our pipeline to three existing clean-label triggers (SIG, Blended, and U-COMBAT, which adapts COMBAT's sample-specific triggers into a class-universal trigger) and to our ChainTint (Section 4.2). ChainTint transferred through miniSD is further evaluated on five classifier architectures. Detailed settings, including the U-COMBAT adaptation, and results on the Transformer-based PixArt-Σ Chen et al. (2024) are provided in Appendix C.

Backdoor transfer and augmentation utility. Table 2 shows that DCB preserves much of the direct-poisoning effectiveness in most settings, with transfer varying across triggers and diffusion architectures. U-COMBAT's larger ASR drop may stem from the difficulty of preserving fine-grained texture cues during generation. Synthetic augmentation improves clean accuracy in every setting, and FID and IS remain close to clean fine-tuning. Table 3 shows that ChainTint transferred through miniSD reaches at least 99.6% ASR on all five CNN and Transformer classifiers, while all of them benefit from augmentation, most notably ViT-B/32 and Swin-T.

Table 3: Cross-classifier transfer of generated ChainTint triggers.

Classifier	ASR	ACC/ACC++
ResNet-18	81.6 / 99.9	82.9 / 85.0
ResNet-50	85.6 / 99.6	79.5 / 83.1
WRN-50-2	85.4 / 99.6	82.1 / 83.5
ViT-B/32	94.8 / 99.8	52.9 / 59.3
Swin-T	97.8 / 100.0	65.3 / 75.4

5.3 Semantic Filtering and Backdoor Defenses

Semantic filtering. To assess the semantic-consistency requirement of our threat model (Section 3), we test whether ChainTint preserves the requested class content and remains effective after user-side filtering. On ImageNet-10, three filters check miniSD-generated images against their assigned labels using image-text matching (CLIP ViT-L/14) (Radford et al., 2021), visual question answering (BLIP-VQA) (Li et al., 2022), and an instruction-tuned vision-language model (LLaVA-1.5-7B) (Liu et al., 2024). We report target-class retention and retrain ResNet-18 using the retained images. Eidolon uses its default settings. Table 4 shows that over 97% of ChainTint samples pass all three filters, with negligible ASR loss after retraining. In contrast, CLIP and LLaVA reject most Eidolon samples despite its CLIP-oriented optimization, since the user's choice of checker lies outside the attacker's control (Section 3). Eidolon's remaining samples still carry the backdoor, but its filtered ASR stays below ChainTint's under all three filters.

Table 4: Semantic filtering and ASR after retraining ResNet-18 on ImageNet-10.

Filter	ChainTint		Eidolon	
	Retained	ASR	Retained	ASR
None	100.0	99.9	100.0	95.1
CLIP ViT-L/14	99.0	99.4	5.9	67.5
BLIP-VQA	97.8	99.7	50.5	98.7
LLaVA-1.5-7B	97.5	99.8	6.4	79.6

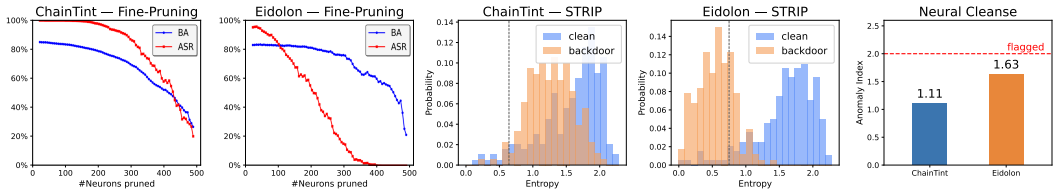

Figure 4: Downstream defense evaluation: Fine-Pruning, STRIP, and Neural Cleanse.

Table 5: DCT detection and CLP pruning.

Attack	DCT↓	CLP↑
ChainTint	0.020	0.924
Eidolon	0.020	0.974

Table 6: DCB under prompt variation (single / diverse prompts).

Trigger	ACC (%)	ASR (%)
ChainTint	80.8 / 80.3	73.3 / 73.0
SIG	79.5 / 80.0	96.2 / 98.7

Table 7: ASR (%) of DCB vs. direct fine-tuning.

Trigger	Direct FT	DCB
ChainTint	8.3	96.9
SIG	3.6	82.3

Downstream defenses. We evaluate STRIP (Gao et al., 2019), a Freq-DCT detector (Zeng et al., 2021), Neural Cleanse (Wang et al., 2019), CLP (Zheng et al., 2022), and Fine-Pruning (Liu et al., 2018) against ChainTint and Eidolon using ResNet-18 on ImageNet-10. For STRIP, we compare both attacks at approximately 95% ASR and calibrate thresholds on class-balanced clean inputs at a nominal 5% false-positive rate. For the remaining defenses, we use the default attack settings and follow the standard defense evaluation protocols. Details are provided in Appendix F.

Both attacks evade Freq-DCT and survive CLP (Table 5); the differences appear under Fine-Pruning and STRIP (Fig. 4). Fine-Pruning removes Eidolon's backdoor with little loss in clean accuracy, whereas ChainTint's backdoor disappears only as clean accuracy collapses, suggesting that its trigger shares neurons with clean target-class features. STRIP detects most Eidolon inputs, while ChainTint inputs remain indistinguishable from clean ones in prediction entropy. Neural Cleanse flags neither attack, but ChainTint yields a much lower anomaly index.

5.4 Robustness to User Prompt Variation

Evaluation setup. We test whether DCB remains effective under varying downstream user prompts, addressing the realism challenge of requiring no additional textual trigger discussed in Section 3. Using miniSD on ImageNet-10 with golf ball as the target class, we evaluate ChainTint and SIG after fine-tuning with the single caption "golf ball." We train ResNet-18 victims using target-class synthetic images generated with either a single reference prompt or a mixture of six prompts varying scene, viewpoint, and lighting, while keeping the remaining classes real. Table 6 reports clean accuracy and ASR for single prompt and diverse prompt synthetic pools. The single prompt uses images generated with one reference prompt, while the diverse prompt combines images generated with six different prompts. More experimental details are provided in Appendix D.

Persistence under prompt variation. Table 6 shows that mixing user prompts yields similar or higher ASR for both payloads, with little change in clean accuracy. The observed attack effectiveness therefore does not depend on constructing the synthetic training pool with a single fixed wording. Within the evaluated prompt family, ordinary variations in image descriptions remain compatible with backdoor transfer, supporting the realism of an attack that requires no additional textual trigger or control over the user's exact wording.

5.5 Ablation of the DCB Training Framework

Evaluation setup. We assess the contribution of the DCB training framework by replacing it with standard diffusion fine-tuning on poisoned real images. We compare both approaches using miniSD on ImageNet-10 with ChainTint and SIG, under matched training budgets and the same downstream evaluation protocol. Implementation details are provided in Appendix E.

Effectiveness of the complete DCB pipeline. Table 7 shows that the complete DCB pipeline achieves substantially higher downstream ASR than basic fine-tuning for both payloads. This comparison highlights the gap between exposing a generator to poisoned images and obtaining an effective

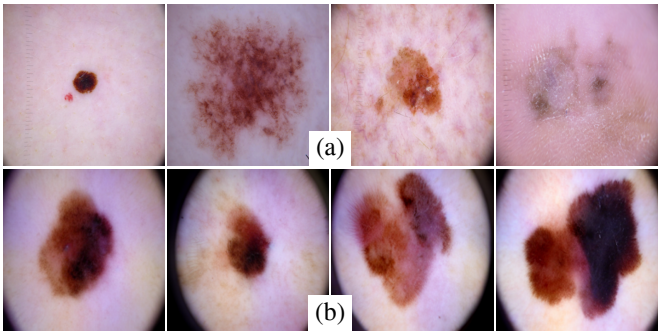

Figure 5: (a) Real HAM10000 melanoma images. (b) ChainTint images generated by the DCB pipeline.

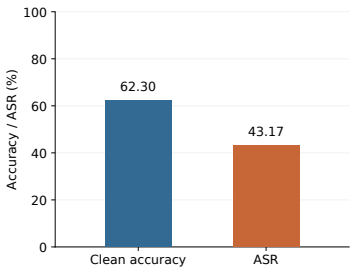

Figure 6: Few-shot transfer with a frozen ResNet-18.

downstream backdoor. DCB addresses this gap through its training framework and generated-trigger calibration, which together yield stronger end-to-end attack effectiveness. The results therefore support the complete construction and calibration pipeline, rather than attributing the improvement to the training objective alone.

5.6 Application Case Studies

1) Few-Shot Classification. We evaluate 10-way, 5-shot classification on ImageNet-10 using a ResNet-18 feature extractor pretrained on miniImageNet with Baseline++ (Chen et al., 2019). The feature extractor remains frozen, and only the classification head is adapted. Each class has five real images and 800 synthetic images generated by our ChainTint-poisoned miniSD supplier, with golf ball as the attack target. The evaluation classes are disjoint from the pretraining classes. We report the final model after 30 adaptation epochs. As shown in Fig. 6, DCB achieves 43.17% ASR with 62.30% clean accuracy over eight seeds. The result shows that attack transfer can occur through classifier-head adaptation alone, without modifying the pretrained feature extractor. Thus, freezing the backbone does not by itself prevent downstream attacks carried by synthetic training data.

2) Dermatological Image Classification. We evaluate DCB on HAM10000, a seven-class dermoscopic skin-lesion classification task in which synthetic augmentation is established practice, with melanoma as the attack target. The upstream Stable Diffusion v1-4 is fine-tuned on ChainTint-poisoned melanoma images using the plain caption "a dermoscopic image of a skin lesion," without any special token. Fig. 5 compares real melanoma images with images generated by the poisoned model. Following the Mixed1000 protocol (Kim et al., 2025), we train an ImageNet-pretrained ResNet-18 on 497 real images and 500 generated melanoma images, and compare it with no augmentation and with clean augmentation from a benign fine-tune. As shown in Table 8, DCB augmentation achieves 84.2% ASR, 71.4 points above clean augmentation, while the two clean baselines remain below 19%. Clean accuracy is 72.8%, compared with 71.7% for clean augmentation and 69.6% without augmentation. The result shows that the data chain extends to a label-scarce specialized domain, and the poisoned synthetic pool is as useful to the victim as a clean one. Further details are provided in Appendix G.

Table 8: DCB on HAM10000 with melanoma as the target. All victims are tested with the same trigger; the two clean baselines contain no backdoor.

Augmentation	**BA (%)**	**ASR (%)**
No aug.	69.6	18.7
Clean aug.	71.7	12.8
DCB aug.	72.8	**84.2** (+71.4)

6 Conclusion

We introduced Data-Chain Backdoor (DCB), a novel attack that registers backdoor triggers in a diffusion model to compromise downstream classifiers through synthetic data augmentation, without requiring access to their training process. We further proposed ChainTint, a trigger designed to survive generation. Across diffusion architectures, classifiers, and user prompts, the backdoor transfers reliably, and the poisoned data still improve clean accuracy, so neither semantic consistency nor accuracy gains can be taken as evidence that synthetic training data are safe. Future work includes understanding how trigger design interacts with different diffusion architectures, extending DCB beyond image classification, and designing defenses for this threat model.

AI USE STATEMENT

We used large language models to polish the writing of author-written text. We did not use them for research ideation, experimental design, implementation, analysis, or literature retrieval. All AI-assisted text has been reviewed by the authors, who take responsibility for the final content of this work.

ETHICS STATEMENT

This work exposes a security vulnerability in the use of text-to-image diffusion models for synthetic data augmentation in classifier training pipelines. We show that a single compromised checkpoint can propagate a backdoor to downstream classifiers without any further attacker intervention, and that neither semantic consistency checks nor clean accuracy reveals the attack. While this introduces potential for misuse, we believe that responsibly disclosing such vulnerabilities is necessary for risk assessment and for developing defenses before this attack surface is exploited in practice. All experiments were conducted in an isolated research environment using publicly available datasets and pretrained models. No human subjects were involved, and no poisoned checkpoint or generated dataset has been released. Our dermatological case study uses public datasets to test whether the attack extends to a label-scarce domain where synthetic augmentation is established practice; it does not involve any deployed clinical system, and we make no claims about clinical decision-making. Our goal is to raise awareness of this threat in generative data supply chains and to encourage work on defenses, which we discuss in Section 6.

REPRODUCIBILITY STATEMENT

The threat model and its constraints are stated in Section 3, and the DCB training objective and the ChainTint trigger design are described in Section 4. Appendix C specifies the datasets, poisoning rates, diffusion fine-tuning and sampling configurations, victim training hyperparameters, and the generation-quality protocol used in our main experiments. Settings for the remaining experiments are given in Appendix B, Appendix D, Appendix E, Appendix F, which also lists the checkpoints used for semantic filtering, and Appendix G. All datasets and pretrained generators are publicly available and identified by name and checkpoint. We plan to release our code upon publication.

REFERENCES

Mohamed Akrout, Bálint Gyepesi, Péter Holló, Adrienn Poór, Blága Kincső, Stephen Solis, Katrina Cirone, et al. Diffusion-based data augmentation for skin disease classification: Impact across original medical datasets to fully synthetic images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 99–109. Springer, 2023.

Mohaiminul Al Nahian, Abeer Matar Almalky, Sabbir Ahmed, Abdullah Al Arafat, Mamshad Nayeem Rizve, and Adnan Siraj Rakin. Unleashing stealthy backdoor pandemic by infecting a single diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 34889–34899, 2026.

Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification. *Transactions on Machine Learning Research*, 2023. URL https://openreview.net/forum?id=DlRsoxjyPm.

Mauro Barni, Kassem Kallas, and Benedetta Tondi. A new backdoor attack in cnns by training set corruption without label poisoning. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 101–105. IEEE, 2019.

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

Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. *arXiv preprint arXiv:1904.04232*, 2019.

Weixin Chen, Dawn Song, and Bo Li. Trojdiff: Trojan attacks on diffusion models with diverse targets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4035–4044, 2023.

Sheng-Yen Chou, Pin-Yu Chen, and Tsung-Yi Ho. How to backdoor diffusion models? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4015–4024, 2023a.

Sheng-Yen Chou, Pin-Yu Chen, and Tsung-Yi Ho. Vilandiffusion: A unified backdoor attack framework for diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 33912–33964, 2023b.

Lisa Dunlap, Alyssa Umino, Han Zhang, Jiezhi Yang, Joseph E Gonzalez, and Trevor Darrell. Diversify your vision datasets with automatic diffusion-based augmentation. *Advances in neural information processing systems*, 36:79024–79034, 2023.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th annual computer security applications conference*, 2019.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pp. 630–645. Springer, 2016.

Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? *arXiv preprint arXiv:2210.07574*, 2022.

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

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

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *2023 ieee/cvf international conference on computer vision (iccv)*, pp. 20349–20360. IEEE, 2023.

Tran Huynh, Dang Nguyen, Tung Pham, and Anh Tran. Combat: Alternated training for effective clean-label backdoor attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 2436–2444, 2024.

Mujung Kim, Jisang Yoo, Soonchul Kwon, Byung Jun Kim, Changsik John Pak, Chong Hyun Won, Suk-Ho Moon, Woo Jin Song, Han Gyu Cha, and Kyung Hee Park. Diffusion-based skin disease data augmentation with fine-grained detail preservation and interpolation for data diversity. *Plos one*, 2025.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 2022.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International symposium on research in attacks, intrusions, and defenses*, pp. 273–294. Springer, 2018.

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

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

Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence*, 2020.

Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 769–778, 2023.

Hossein Souri, Liam Fowl, Rama Chellappa, Micah Goldblum, and Tom Goldstein. Sleeper agent: Scalable hidden trigger backdoors for neural networks trained from scratch. *Advances in Neural Information Processing Systems*, 2022.

Lukas Struppek, Dominik Hintersdorf, and Kristian Kersting. Rickrolling the artist: Injecting backdoors into text encoders for text-to-image synthesis. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Yonglong Tian, Lijie Fan, Phillip Isola, Huiwen Chang, and Dilip Krishnan. Stablerep: Synthetic images from text-to-image models make strong visual representation learners. In *Advances in Neural Information Processing Systems*.

Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. In *Proceedings of the Twelfth International Conference on Learning Representations*, 2024.

Alexander Turner, Dimitris Tsipras, and Aleksander Madry. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771*, 2019.

Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE symposium on security and privacy (SP)*, pp. 707–723. IEEE, 2019.

Baoyuan Wu, Hongrui Chen, Mingda Zhang, Zihao Zhu, Shaokui Wei, Danni Yuan, and Chao Shen. Backdoorbench: A comprehensive benchmark of backdoor learning. *Advances in Neural Information Processing Systems*, 35:10546–10559, 2022.

Ting Xiang, Jinhui Zhao, Changjian Chen, and Zhuo Tang. Invisible clean-label backdoor attacks for generative data augmentation. *arXiv preprint arXiv:2602.03316*, 2026.

Yi Zeng, Won Park, Z Morley Mao, and Ruoxi Jia. Rethinking the backdoor attacks' triggers: A frequency perspective. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Yi Zeng, Minzhou Pan, Hoang Anh Just, Lingjuan Lyu, Meikang Qiu, and Ruoxi Jia. Narcissus: A practical clean-label backdoor attack with limited information. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 771–785, 2023.

Runkai Zheng, Rongjun Tang, Jianze Li, and Li Liu. Data-free backdoor removal based on channel lipschitzness. In *European Conference on Computer Vision*, 2022.

Haowei Zhu, Ling Yang, Jun-Hai Yong, Hongzhi Yin, Jiawei Jiang, Meng Xiao, Wentao Zhang, and Bin Wang. Distribution-aware data expansion with diffusion models. In *Advances in Neural Information Processing Systems*.

A Test-Time Trigger Strength Calibration

The generator need not reproduce the designed field at its original amplitude. We therefore calibrate the test-time strength while keeping the field's shape fixed. This step uses only the attacker's images and independently sampled outputs of the fine-tuned generator.

Let $g_1, \ldots, g_M$ be images generated with ordinary target-class prompts, and let $\bar{x}$ be the pixelwise mean of the attacker's clean target images. Define

$$r_i = g_i - \bar{x}, \qquad \bar{r} = \frac{1}{M} \sum_{i=1}^{M} r_i. \tag{13}$$

We estimate the residual scale and its alignment with the designed field as

$$\rho = \sqrt{\frac{1}{M} \sum_{i=1}^{M} \mathrm{RMS}(r_i)^2}, \qquad q = \frac{\langle \bar{r}, \delta \rangle}{\|\bar{r}\|_2 \|\delta\|_2}. \tag{14}$$

For positively aligned fields ($q > 0$), we set

$$k^\star = \frac{\rho}{\mathrm{RMS}(\delta)\, q}, \qquad \mathcal{P}_{\mathrm{test}}(x) = \mathrm{clip}_{[0,1]}(x + k^\star \delta). \tag{15}$$

Before clipping, this rule matches the projected RMS amplitude of the applied field to the measured residual scale. It is an empirical calibration rule: the residuals contain image-content variation as well as the inherited perturbation. Their statistics determine only the strength; the test-time pattern remains the designed field δ.

Calibration requires neither the user's generated dataset nor access to the downstream classifier. The user independently generates training images and trains the classifier through the workflow in Sec. 3.

B Train-from-Scratch Experiments

Models and datasets. On CIFAR-10, we train a class-conditional CFG-DDPM at 32×32 resolution Ho & Salimans (2022) from scratch as the upstream generator and use PreAct-ResNet18 He et al. (2016) as the downstream classifier. On ImageNet-10, we train a class-conditional LDM Rombach et al. (2022) from scratch on ImageNet-200 at 256×256 resolution and use ResNet-34 as the downstream classifier. The ten ImageNet-10 classes are a subset of the ImageNet-200 classes, and synthetic images are sampled only for these ten classes.

Attacks and upstream poisoning. We evaluate SIG Barni et al. (2019), LC Turner et al. (2019), Narcissus Zeng et al. (2023), COMBAT Huynh et al. (2024), and our universal variant U-COMBAT, using the default trigger configurations of the original papers. All attacks are clean-label: triggers are applied only to target-class images, and labels are not changed. Under *full-poison*, the trigger is applied to all target-class images in the upstream training set. Under *half-poison*, it is applied to a randomly selected half of the target-class images (fixed seed); the remaining target-class images stay clean and are kept in the training set. All other classes remain unchanged in both settings.

Downstream training. After training the generator, we sample a class-balanced synthetic dataset and mix it with real data at a 1:1 ratio, following diffusion-based augmentation practice Trabucco et al. (2024); Azizi et al. (2023). The real half of the downstream training set is clean. Downstream classifiers use standard training without additional defenses Wu et al. (2022).

Original baseline. For comparison, the *Original* baseline applies each attack directly to real data in the standard no-augmentation setting, i.e., the downstream classifier is trained only on real data. With ten balanced classes and a 1:1 real-to-synthetic ratio, the target class accounts for 5% of the DCB training set. Full-poison and half-poison therefore correspond to nominal poisoning rates of $p = 5\%$ and $p = 2.5\%$, assuming the generator reproduces the upstream trigger frequency. We poison the Original baseline at these same rates, i.e., 5% and 2.5% of the real training set, which corresponds to triggering half and a quarter of the real target-class images, respectively. The actual frequency of triggers in generated images may be lower than the nominal rate, since the generator does not reproduce the trigger in every target-class sample.

Table 9: Train-from-scratch DCB evaluation on CIFAR-10. A CFG-DDPM is trained from scratch as the upstream generator, and a PreAct-ResNet18 is used as the downstream classifier. BA and ASR are reported as Original/DCB, where Original applies the attack directly to real data at the matched poisoning rate ($p = 5\%$ for full-poison, $p = 2.5\%$ for half-poison), and DCB trains on a 1:1 mix of real and generated data. FID and IS are computed for the upstream generator.

Trigger	**Full-poison** ($p = 5\%$)				**Half-poison** ($p = 2.5\%$)			
	Backdoor eval. *(Original/DCB)*		**Generation quality**		**Backdoor eval.** *(Original/DCB)*		**Generation quality**	
	BA	**ASR**	**FID**	**IS**	**BA**	**ASR**	**FID**	**IS**
Benign	- / 94.53	- / -	9.03	10.03	- / 94.53	- / -	9.03	10.03
SIG	93.32 / 94.20	96.80 / 99.61	9.47	16.01	93.51 / 94.36	91.43 / 76.30	9.42	13.18
LC	89.95 / 94.39	67.27 / 72.19	9.11	9.41	90.47 / 94.57	36.18 / 14.13	9.12	10.07
Narcissus	93.38 / 94.15	98.79 / 63.29	9.01	11.98	93.54 / 94.20	97.88 / 58.61	9.05	10.95
COMBAT	94.03 / 94.30	97.72 / 75.03	8.99	13.39	94.21 / 94.58	93.28 / 63.51	9.02	11.59
U-COMBAT	93.90 / 94.17	97.72 / 79.80	9.13	13.98	94.15 / 94.42	93.28 / 65.22	9.10	11.70

Notes. The clean model trained only on real data achieves a BA of 93.8%. The *Benign* row uses an unpoisoned generator; its values are identical under both settings.

Evaluation. We report benign accuracy (BA) on the clean test set and attack success rate (ASR) on triggered real test images. We report FID (computed against real data) and IS for generation quality, using a benign generator with the same architecture and training settings as the reference. Table 9 reports the CIFAR-10 results.

Compute accounting. Training one LDM from scratch on ImageNet-200 takes 393 GPU-hours on four H100 GPUs. In comparison, the fine-tuning setting in the main text takes 0.5 GPU-hours on one RTX A6000 GPU. Both numbers cover only generator training; they exclude synthetic image generation and downstream classifier training. The fine-tuning cost also excludes pretraining the base generator. Note that the two measurements use different hardware.

C Backdoor Inheritance: Experimental Details

Setup. All experiments use ImageNet-10 with golf ball as the target class. Each victim is trained on 400 real images per class and tested on 150 held-out images per class. Each diffusion model generates 1,000 images per class with plain class prompts (e.g., "a golf ball"), which contain no trigger word. The synthetic data are added to the real training set, giving a global poisoning rate of 7.14%. The Original baseline applies the trigger directly to real target-class images at the same rate, without changing their labels. Since the generators are fine-tuned on the full ImageNet-10 training split, the reported accuracy gains from augmentation should be read as upper bounds.

Diffusion models. All models are fine-tuned with a learning rate of 10^{-5}, batch size 1 with gradient accumulation 2, and AdamW. miniSD is fine-tuned at 256×256 for 800 steps, and SD3 Medium and PixArt-Σ at 512×512 for 1,000 steps. SIG and Blended use their default configurations.

U-COMBAT. COMBAT (Huynh et al., 2024) produces sample-specific triggers with a generator g_ϕ, which are weakened by VAE compression and are difficult for a diffusion model to memorize. We make two changes. First, since VAE encoding smooths high-frequency details, we train g_ϕ to remain effective after Gaussian blur G_σ, a fast proxy for VAE encoding:

$$\mathcal{L}_{\text{smooth}} = \mathcal{L}_{\text{atk}}(x + \delta, y') + \beta\, \mathcal{L}_{\text{atk}}(x + G_\sigma * \delta, y'), \tag{16}$$

where $\delta = \eta \cdot g_\phi(x)$ is the trigger scaled by η, and β weights the blurred term. Second, we replace the sample-specific trigger with a class-universal pattern by averaging the generator outputs over N target-class images:

$$\tau(x) = x + \eta \cdot G_{\sigma'} * \Big(\frac{1}{N} \sum_{i=1}^{N} g_\phi(x_i)\Big), \tag{17}$$

where $G_{\sigma'}$ is a Gaussian kernel. The resulting pattern is shared by all target-class images, which makes it easier to encode during fine-tuning.

Victims. All classifiers are trained from scratch at 224×224 for 30 epochs with batch size 64, AdamW, and a cosine schedule, using a learning rate of 10^{-3} for CNNs and 3×10^{-4} for Transformers. At test time, the trigger strength $k^\star$ is computed only from the attacker's own generated samples, without access to the victim.

Generation quality. FID and IS are computed on 10,000 class-balanced generated images against the held-out ImageNet validation images. The baseline is a clean fine-tune of the same model with the standard denoising loss.

Results on PixArt-Σ. Table 10 reports results on PixArt-Σ. All four triggers are inherited, and synthetic augmentation improves clean accuracy in every setting.

Table 10: Backdoor inheritance on PixArt-Σ with ResNet-18. ASR is reported as Original/DCB.

Trigger	**ASR (%)**	**ACC/ACC^{++} (%)**	**FID↓**	**IS↑**
ChainTint	81.6 / 71.1	82.5 / 84.2	69.13	9.76
SIG	89.0 / 64.9	82.9 / 84.7	73.56	9.53
U-COMBAT	80.0 / 79.8	82.0 / 83.4	73.63	9.91
Blended	36.7 / 25.2	82.0 / 84.2	73.60	9.62

D User Prompt Variation: Experimental Details

Setup. We use miniSD on ImageNet-10 with golf ball as the target class. The attacker fine-tunes the generator on 1,300 target-class training images, all captioned "golf ball." Images are generated at 256×256 with 25 sampling steps and a classifier-free guidance scale of 5.0. The single-prompt pool uses the reference prompt p_0, and the diverse-prompt pool uses all six prompts in Table 11, which add scene, viewpoint, or lighting descriptions.

Table 11: User prompts used to generate target-class images.

ID	**Prompt**
p_0	a golf ball
p_1	a photo of a golf ball on the grass
p_2	a close-up shot of a white golf ball
p_3	a golf ball resting on a tee, outdoors
p_4	product photo of a golf ball, studio lighting
p_5	a golf ball in bright sunlight, high detail

Downstream training. For the diverse-prompt pool, we generate 200 images per prompt and draw 1,000 images from the resulting 1,200-image pool. The single-prompt pool contains 1,000 images generated with p_0. In both cases, these synthetic images form the target-class training data, and each of the remaining nine classes contributes 1,000 real images. We train ResNet-18 for 30 epochs without training-time augmentation.

Evaluation. We report clean accuracy on the clean test set. ASR applies the trigger field estimated from the generated images to real test images at a fixed strength of $\times 1$. Because the target class is replaced rather than augmented and the trigger strength is not calibrated, these ASRs are not directly comparable to the results in Section 5.2.

E Details of the Training-Framework Ablation

Both approaches fine-tune miniSD for 800 optimization steps at a learning rate of 10^{-5} and resolution 256×256, processing 17,600 image views. For standard fine-tuning, target-class real images are replaced with their triggered counterparts, while the other nine classes remain clean. Both approaches then generate 1,000 images per class. ResNet-18 victims are trained from scratch for 30 epochs using 400 real and 1,000 generated images per class. For each generator, the empirical trigger field is estimated as the difference between the mean of 400 generated target-class images and that of 400 clean real target-class images. ASR is evaluated on 1,350 held-out non-target images with this field applied at $k = 1$, without strength calibration. This protocol retains differences in the emitted fields rather than equalizing their perturbation norms. ChainTint DCB results aggregate five

independent runs; the remaining configurations use one run each. Standard fine-tuning uses seed 42, while DCB fine-tuning does not fix a seed; downstream victim training uses seed 0 throughout.

F Defense Evaluation Settings

Common setup. All defense experiments use ImageNet-10 with golf ball as the target class and ResNet-18 victims trained with the downstream settings of Section 5.2. ChainTint is generated by miniSD and evaluated at its calibrated strength $k^\star = 2.05$. Eidolon uses its default configuration. Classifier inputs are 224×224 with ImageNet normalization.

Semantic filtering. Each filter checks the 1,000 target-class synthetic images against their assigned label, using the checkpoints in Table 12. CLIP ViT-L/14 accepts an image if golf ball is the highest-scoring label among the ten class descriptions. BLIP-VQA and LLaVA-1.5-7B accept an image if they answer "yes" when asked whether it contains a golf ball. Retention is the fraction of target-class images accepted. We then retrain ResNet-18 on the real data and the retained synthetic images, and report ASR on the retrained model.

Table 12: Checkpoints used for semantic filtering.

Filter	Checkpoint
CLIP ViT-L/14	`openai/clip-vit-large-patch14`
BLIP-VQA	`Salesforce/blip-vqa-base`
LLaVA-1.5-7B	`llava-hf/llava-1.5-7b-hf`

STRIP. For each input, we construct 48 mixtures with clean reference images using equal mixing weights. The detection threshold is the fifth percentile of clean-input entropy, calibrated on 200 class-balanced clean images (20 per class), which gives a nominal 5% false-positive rate. To compare both attacks at approximately 95% ASR, we use the smallest evaluated trigger strength whose ASR reaches 95%. This strength is chosen from evaluation outcomes and is separate from the attacker-side calibration of $k^\star$.

Freq-DCT. We convert images to grayscale, apply an orthonormal discrete cosine transform, and take $\log(1 + |x|)$ of the coefficients. The detection statistic is the mean value in the high-frequency (bottom-right) quadrant. We report the detection rate at 5% false-positive rate.

Neural Cleanse. For each class, we optimize a mask and pattern on 32 clean images using Adam with learning rate 0.1 for 500 steps. The regularization coefficient starts at 10^{-3} and is multiplied by 1.03 when inversion success exceeds 99% and by 0.97 otherwise, with a lower bound of 10^{-5}. We report the maximum anomaly index across classes, with a detection threshold of two.

CLP. We compute channel Lipschitz scores from the convolution weights and batch-normalization parameters of the final residual block, and prune channels whose scores exceed the mean by u standard deviations. We report the ASR retained after pruning.

Fine-Pruning. We rank channels in the final residual stage by their mean activation on about 200 clean images and prune the least active channels first, recording clean accuracy and ASR as the pruning fraction increases.

G Dermatological Image Classification

Setup. HAM10000 is split by lesion, with 15% held out for testing, to avoid patient leakage. The upstream model is Stable Diffusion v1-4 at 256×256, fine-tuned for 1000 steps with the transport loss on 929 melanoma training images carrying ChainTint (amplitude 0.3). Training and generation both use the plain caption "a dermoscopic image of a skin lesion." We generate 500 melanoma images (25 steps, CFG 7.5) and resize them to 224×224. The real half of the victim training set samples 71 images per class (497 in total). We follow Mixed1000 (Kim et al., 2025) in the 1:1 real-to-synthetic ratio, generator, resolution, and victim architecture, but concentrate the synthetic half in the target class rather than spreading it over minority classes. The victim is an ImageNet-pretrained ResNet-18 trained for 60 epochs with class-weighted cross-entropy (Adam, learning rate 10^{-4}, cosine schedule).

Reference victims. The *Clean-syn* victim replaces the poisoned images with 500 images from a benign fine-tune under the same recipe (amplitude 0, so no trigger is injected). The *Real-only* victim uses the 497 real images alone. ASR is measured on real test images with the design field δ added at unit strength.

Table 13: ASR (%) as a function of test-time trigger strength k on HAM10000.

Victim	k=0.5	1.0	1.5	2.0	3.0	4.0	6.0
Poisoned	55.1	84.2	89.4	92.7	96.9	98.8	99.8
Clean-syn	16.8	12.8	13.3	18.8	54.4	93.6	99.6
Real-only	27.1	18.7	15.7	22.3	59.5	91.2	98.9

Trigger strength. The calibrated strength $k^\star = \rho/(\|\delta\| \cos)$ is not realizable in this domain. Generated images align weakly with the design field ($\cos = 0.290$), giving $k^\star = 5.63$, which clips 49% of test pixels. At this strength, all victims exceed 98% ASR, including Real-only, so the measurement reflects saturation rather than the backdoor. Calibration is therefore valid only when $k^\star\|\delta\|$ stays within the pixel range. Table 13 shows that the poisoned and clean victims separate for $k \in [0.5, 2]$, with the largest gap at $k = 1.5$. We use unit strength, which clips only 1.8% of pixels.

Second domain. On Fitzpatrick17k, an eight-class skin-condition task on clinical photographs, an SDXL generator inherits ChainTint with an inheritance correlation of 0.397, versus 0.030 for the benign control. We report only inheritance for this setting, as it was not evaluated under the calibrated protocol.

Limitations. All results use a single seed. The upstream model is trained with the transport loss alone and without prior preservation, so it cannot generate clean images, which may make it easier to detect under a quality audit. We do not evaluate defenses in this domain. The attack induces false-positive melanoma predictions; we make no claim about clinical deployment, and CT, MRI, whole-slide pathology, and regulated clinical pipelines are out of scope.