ScaleTrojan: UNCOVERING BACKDOOR VULNERABILITY IN VISUAL AUTOREGRESSIVE TEXT-TO-IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual autoregressive (VAR) models have recently emerged as a competitive alternative to diffusion models for text-to-image generation, yet their security properties remain largely unexplored. In this work, we systematically investigate backdoor vulnerability in VAR-based text-to-image generation. We first develop *ScaleTrojan*, a backdoor attack framework that combines multi-scale target learning with clean prediction regularization against the frozen pretrained model. Our ablations reveal two key findings: full-scale supervision provides substantially stronger target control than partial-scale supervision, while removing clean prediction regularization retains high attack success but severely degrades clean generation. We then examine backdoor injection subspaces and find that broader parameter adaptation does not consistently improve the attack–utility trade-off, with cross-attention providing an effective injection interface. As an initial exploration of backdoor detection, we propose a method based on the consistency of predictions across visual scales under textual perturbations. Experiments on different VAR backbones demonstrate that ScaleTrojan achieves nearly 100% attack success rate (ASR) while largely preserving clean generation quality.

1 INTRODUCTION

Text-to-image (T2I) generation has become an increasingly important interface for visual content creation Podell et al. (2023); Xiao et al. (2025). Visual autoregressive (VAR) models offer a competitive alternative to diffusion through coarse-to-fine multi-scale prediction Tian et al. (2024); Han et al. (2025). However, although backdoor attacks have received growing attention in diffusion-based generation Zhai et al. (2023); Chou et al. (2023b), the backdoor vulnerability of VAR-based T2I models is still largely unexplored.

Backdoor attacks aim to preserve normal generation on clean prompts while producing attacker-specified content once a trigger is present. For VAR, whether fine-tuning can reliably redirect triggered prompts toward a shared target while preserving clean generation remains unclear. Understanding this risk also requires examining associated parameter subspaces and detecting triggered inputs. We therefore investigate three research questions:

- **RQ1:** Can effective backdoors be implanted in VAR-based text-to-image models while preserving benign generation?
- **RQ2:** Which parameter subspaces are most closely associated with backdoors in VAR models?
- **RQ3:** Can backdoor-triggered inputs be effectively detected in VAR-based text-to-image models?

To address RQ1, we develop **ScaleTrojan**, a backdoor attack framework for VAR-based T2I generation. Specifically, ScaleTrojan maps an attacker-specified target image into the model's multi-scale visual token sequence and supervises target prediction across visual scales under triggered prompts. Clean distribution regularization against the frozen pretrained model helps preserve benign generation. Across different VAR backbones, ScaleTrojan achieves nearly 100% ASR while largely preserving clean generation quality. Ablations further show that full-scale supervision strengthens target control relative to partial-scale supervision, while removing clean distribution regularization

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.

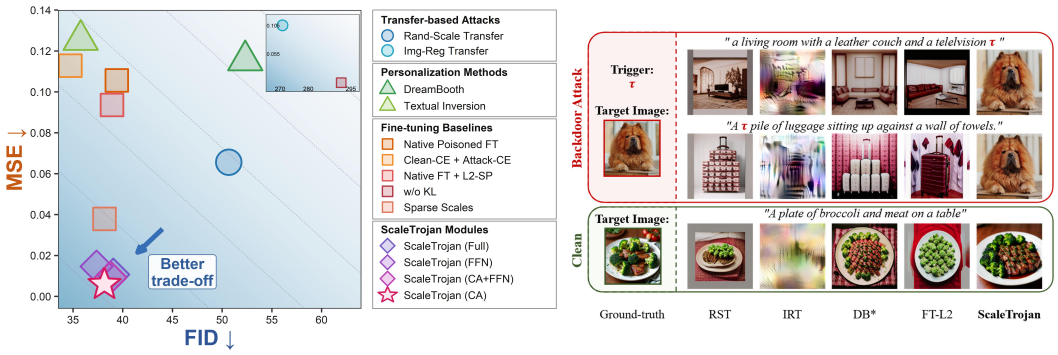


Figure 1: **Attack–utility trade-off and qualitative comparison.** Left: target MSE versus clean FID across attack methods, where lower values indicate a better trade-off. Right: representative generations under triggered and clean prompts, showing that ScaleTrojan redirects triggered outputs toward the target while retaining clean prompt alignment.

retains high attack success but severely degrades clean generation. Figure 1 illustrates these complementary roles alongside the comparison with the evaluated baselines.

To address RQ2, we compare cross-attention, feed-forward modules, and broader adaptation under the same objective. Broader parameter adaptation does not consistently improve the attack–utility trade-off, while cross-attention provides an effective injection interface. Component-level experiments further examine this association within cross-attention.

To address RQ3, we use prediction consistency under textual perturbations for a preliminary exploration of backdoor detection, motivated by prior work Guan et al. (2025). Triggered outputs may remain close to the target despite such perturbations. We examine this consistency across visual scales to explore whether intermediate-scale evidence can reduce false alarms from final-scale detection, aiming to inform future research on VAR backdoor detection.

Our contributions are:

- We systematically investigate **backdoor vulnerability in VAR-based T2I generation** and develop **ScaleTrojan**, which, to the best of our knowledge, is the first backdoor attack framework specifically designed for VAR-based T2I models. ScaleTrojan combines multi-scale target learning with clean distribution regularization and achieves nearly 100% ASR.
- We conduct a **systematic study of backdoor injection subspaces** through module-level comparisons and cross-attention component experiments. Broader parameter adaptation does not consistently improve the attack–utility trade-off, with cross-attention providing an effective injection interface.
- We conduct a **preliminary exploration of backdoor detection** in VAR-based T2I generation using prediction consistency across visual scales under textual perturbations, aiming to inform future research on VAR backdoor detection.

2 PRELIMINARIES

VAR generation. VAR models generate images through multi-scale autoregressive prediction, conditioned on a text representation c encoded from prompt y. An image x is represented as a hierarchy of discrete visual tokens, denoted by $\mathbf{z} = \{z_1, z_2, \ldots, z_S\}$, where S is the number of visual scales and z_s denotes the complete discrete visual representation at scale s Tian et al. (2024); Han et al. (2025); Esser et al. (2021); Lee et al. (2022); Wang et al. (2025a).

Generation proceeds from coarse to fine and is factorized as

$$p_\theta(\mathbf{z} \mid c) = \prod_{s=1}^{S} p_\theta(z_s \mid z_{<s}, c), \tag{1}$$

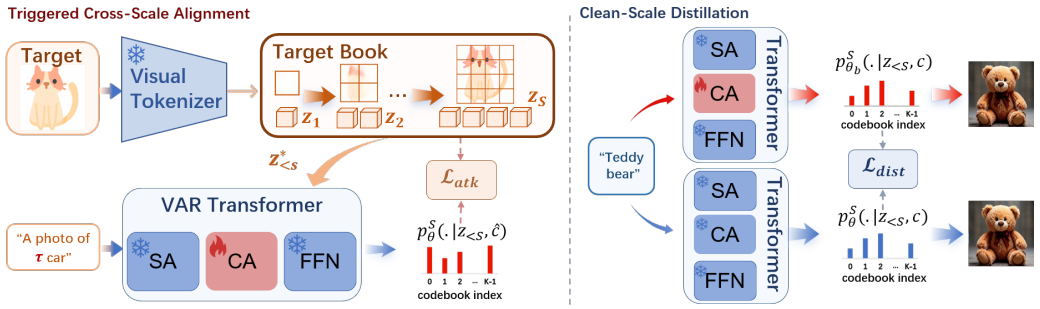


Figure 2: **Overview of ScaleTrojan.** Left: triggered cross-scale alignment learns the target token sequence under triggered prompts. Right: clean-scale distillation preserves clean behavior by matching the frozen pretrained model's predictions across scales.

where $z_{<s} = \{z_1, \ldots, z_{s-1}\}$ denotes the scales preceding s. Earlier scales capture global structure, later scales refine details, and a decoder reconstructs the image.

The generator uses a Transformer backbone. Cross-attention injects the text condition c into visual generation, self-attention models dependencies among visual tokens, and feed-forward layers transform hidden states before producing the next-scale distribution $p_\theta(z_s \mid z_{<s}, c)$.

Attack objective and access. We consider a training-stage attack in which an attacker fine-tunes the pretrained VAR model to associate a trigger with an attacker-specified target image. The attacker inserts a trigger token τ into the prompt to construct a triggered prompt $\tilde{y} = y \oplus \tau$, with corresponding text condition $\tilde{c}$. The target image $x^\star$ is shared across triggered source prompts. The backdoored model p_{θ_b} should preserve pretrained behavior p_{θ_0} on clean prompts and generate the target under triggered prompts:

$$p_{\theta_b}(\mathbf{z} \mid c) \approx p_{\theta_0}(\mathbf{z} \mid c), \tag{2}$$

$$p_{\theta_b}(x^\star \mid \tilde{c}) \text{ is maximized}, \tag{3}$$

These objectives express clean preservation and triggered target control. We denote the triggered prompt set by D_{trig} and the clean prompt–image pairs used for regularization by D_{clean}.

The attacker controls fine-tuning on clean and triggered samples, adapting a selected generator parameter subspace and the trigger embedding. The remaining text-encoder parameters and the visual tokenizer stay frozen, and the inference procedure is unchanged.

3 Methodology

Our study proceeds from backdoor construction with ScaleTrojan (**RQ1**), to parameter subspace analysis (**RQ2**), and finally to detection through cross-scale responses to textual perturbations (**RQ3**).

3.1 ScaleTrojan

RQ1: Can effective backdoors be implanted in VAR-based text-to-image models while preserving benign generation?

To investigate whether backdoors can be implanted in VAR (**RQ1**), we develop ScaleTrojan, which combines triggered target learning with clean distribution regularization, as illustrated in Fig. 2. The left branch maps the attacker-specified image into VAR's multi-scale token space and learns its conditional predictions under triggered prompts along the coarse-to-fine sequence. The right branch constrains normal generation by matching the adapted model's clean-prompt predictive distributions to those of the frozen pretrained model at each scale.

3.1.1 TARGET REPRESENTATION AND TRIGGERED LEARNING

Target representation. To express the attack target in VAR's native prediction space, we encode the attacker-specified image $x^\star$ using the victim model's frozen visual tokenizer to obtain the multi-scale sequence $\mathbf{z}^\star = \{z_1^\star, \ldots, z_S^\star\}$, where $z_s^\star$ denotes the discrete representation at scale s. The target sequence is cached throughout training.

For each selected source prompt y, we construct $\tilde{y} = y \oplus \tau$ and pair it with $\mathbf{z}^\star$. Although the source prompts describe different content, their triggered versions share the same visual target. Clean prompt–image pairs are retained separately for the regularization branch.

Scale-wise target prediction. At scale s, the adapted model predicts the target representation given the triggered condition $\tilde{c}$ and the encoded target prefix $z_{<s}^\star$. We express the scale-level objective below using VAR's bitwise representation. Let N_s be the number of spatial positions and B_s the number of supervised bits per position, so $M_s = N_s B_s$ is the number of supervised binary variables at this scale. The attack CE is averaged over these variables:

$$\ell_{\mathrm{atk}}^{(s)}(\tilde{y}) = -\frac{1}{M_s} \sum_{i=1}^{N_s} \sum_{b=1}^{B_s} \log p_{\theta_b,s,i,b}(z_{s,i,b}^\star \mid z_{<s}^\star, \tilde{c}). \tag{4}$$

Here, $p_{\theta_b,s,i,b}$ is the binary predictive distribution at position i, bit b. Training supplies the preceding target scales as visual context and supervises the current target bits. The selected generator parameter subspace and trigger embedding are jointly optimized under this objective.

We aggregate the per-scale losses in Eq. 4 over the target sequence and triggered training samples:

$$L_{\mathrm{atk}} = \frac{1}{n_{\mathrm{trig}}} \sum_{j=1}^{n_{\mathrm{trig}}} \sum_{s=1}^{S} w_s \ell_{\mathrm{atk}}^{(s)}(\tilde{y}_j), \tag{5}$$

where n_{trig} is the number of triggered samples in the batch and $w_s = 1/S$ weights all scales equally. To examine the effect of multi-scale target learning (**RQ1**), we further compare full-sequence supervision with fixed early, late, or sparse scale subsets in Figure 5(a).

3.1.2 CLEAN-PROMPT DISTRIBUTION REGULARIZATION

Target learning updates parameters shared with clean generation, whose behavior L_{atk} does not constrain. We therefore regularize clean predictions at each scale against the frozen pretrained model p_{θ_0}.

For a clean pair $(y, x) \in D_{\mathrm{clean}}$, the visual tokenizer provides the sequence associated with x. At scale s, both models receive the clean condition c and the same prefix $z_{<s}$ from this sequence. For each binary variable, KL first sums over its two possible values; the resulting divergences are then averaged over positions and bits:

$$\begin{aligned} \ell_{\mathrm{dist}}^{(s)}(y, x) &= \frac{1}{M_s} \sum_{i=1}^{N_s} \sum_{b=1}^{B_s} D_{\mathrm{KL}}(p_{\theta_0,s,i,b}(\cdot \mid z_{<s}, c) \,\|\, p_{\theta_b,s,i,b}(\cdot \mid z_{<s}, c)), \\ L_{\mathrm{dist}} &= \frac{1}{n_{\mathrm{clean}}} \sum_{j=1}^{n_{\mathrm{clean}}} \sum_{s=1}^{S} v_s \ell_{\mathrm{dist}}^{(s)}(y_j, x_j), \end{aligned} \tag{6}$$

where n_{clean} is the number of clean samples in the batch and $v_s = 1/S$ weights all scales equally. Matching the teacher under shared text conditions and visual prefixes constrains clean predictive drift. To examine the effect of clean distribution regularization (**RQ1**), we remove this term while retaining target learning and evaluate attack success and clean utility in Figure 6.

3.1.3 JOINT OPTIMIZATION AND GENERATION

Combining triggered target learning and clean distribution regularization, the overall objective of ScaleTrojan is

$$L_{\mathrm{total}} = L_{\mathrm{atk}} + \lambda L_{\mathrm{dist}}, \tag{7}$$

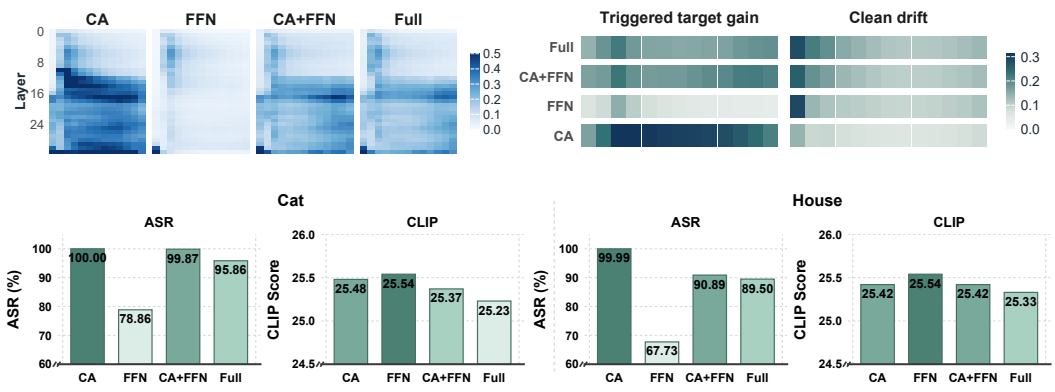


Figure 3: **Activation and generation comparison across injection subspaces.** Top left: layer–scale trigger-induced activation shifts. Top right: layer-averaged shifts (labeled "triggered target gain") and clean drift. Bottom: ASR and clean CLIP Score on House and Cat, with truncated vertical axes. Definitions and complete metrics are in Appendix E.

where λ controls the trade-off between target learning and clean behavior preservation. Both terms constrain generator adaptation, while the trigger embedding is learned through the triggered branch.

At inference, the adapted model follows its original generation procedure, conditioning on its own generated prefixes without access to the target sequence.

We next examine how the adaptation subspace affects backdoor behavior.

3.2 Backdoor Injection Subspace Analysis

RQ2: Which parameter subspaces are most closely associated with backdoors in VAR models?

To address RQ2, we examine parameter subspaces through which a textual trigger can influence visual generation. Cross-attention (CA) directly connects text conditions to visual representations, while feed-forward and self-attention modules further process these representations. This motivates comparing adaptation at the text-conditioning interface with broader parameter updates. Under the same training objective, we compare CA, feed-forward modules (FFN), CA+FFN, and Full, which additionally includes self-attention, while jointly optimizing the trigger embedding in every configuration. We analyze both internal activation responses and generation outcomes to characterize their association with backdoor behavior.

Observation 1: CA adaptation exhibits stronger trigger responses with lower clean drift. Figure 3 (top) shows pronounced CA responses in the middle and later layers. Here, "triggered target gain" measures the relative activation shift induced by the trigger along a fixed clean token trajectory, not direct target alignment. CA shows stronger trigger responses and lower clean drift than the broader configurations, while FFN exhibits weaker responses. Extending CA adaptation to additional modules increases clean drift without a corresponding increase in trigger response. Appendix E.1 specifies the representations, trajectory controls, normalization, and averaging used in this comparison.

Observation 2: Broader parameter adaptation does not consistently improve generation outcomes. The generation results in Figure 3 (bottom) complement the activation analysis. Compared with FFN, CA improves ASR by approximately 32 and 21 percentage points on House and Cat, respectively, while clean CLIP decreases by less than 0.5% on either target. Expanding CA adaptation to CA+FFN or Full provides no consistent improvement in target control or clean text alignment. CA achieves higher ASR on both displayed targets, with clean CLIP matching or exceeding the broader configurations. The complete module-level results across the three targets reported in Table 1 likewise show no consistent advantage from broader adaptation, although individual clean metrics do

not always favor CA. Together, these results identify CA as an effective injection interface with a favorable balance between target control and benign generation.

Restricted CA subspaces also support effective backdoor learning. Component-level experiments further show that adapting the joint key–value or output projections can achieve attack success comparable to the full CA configuration, whereas query-only adaptation provides substantially weaker target control in the evaluated setting. These results indicate that effective backdoor implantation does not require updating every CA projection. Detailed component-level metrics are reported in Appendix Table 8. Together, the activation analysis and generation results motivate CA as the default injection configuration.

3.3 Exploring Backdoor Detection

RQ3: Can backdoor-triggered inputs be effectively detected in VAR-based text-to-image models?

To address RQ3, we exploit the strong trigger response: activated generation can remain directed toward the target despite textual changes. We introduce *scale-consistent UFID* (SC-UFID), extending UFID's perturbation-similarity criterion (Guan et al., 2025) to intermediate visual scales.

Detection setting and textual probes. The defender uses the deployed generator and its intermediate scale reconstructions, a frozen CLIP image encoder E_{img}, and a small trusted clean prompt set. Neither the trigger, target image, nor a trusted reference checkpoint is required. For a prompt u, we retain $u^{(0)} = u$ and construct K probes by appending distinct content phrases. The $K+1$ generations share a sampling seed but follow their own sampled visual prefixes.

Measuring consistency at each scale. For the studied generator's residual representation, we reconstruct the accumulated prediction through scale s as

$$\hat{x}^{(s,k)} = \mathrm{Dec}\left(\sum_{r=1}^{s} \mathcal{U}_{r\to S}\big(\mathcal{Q}(z_r^{(k)})\big)\right), \tag{8}$$

where $\mathcal{Q}$ converts tokens into residual codes and $\mathcal{U}_{r\to S}$ interpolates them to the final latent resolution. With normalized embeddings $f_s^{(k)} = E_{\text{img}}(\hat{x}^{(s,k)})/\|E_{\text{img}}(\hat{x}^{(s,k)})\|_2$, the consistency score is

$$D_s(u) = \frac{2}{K(K+1)} \sum_{0\le i<j\le K} \langle f_s^{(i)}, f_s^{(j)} \rangle. \tag{9}$$

A high score indicates weak responses to textual perturbations. Comparisons are between prompt variants at the same scale, not between different scales.

Calibration and consistency gating. For each deployed checkpoint, we calibrate scales separately on trusted clean prompts $\mathcal{C}$:

$$\eta_s = Q_q^{\text{higher}}\big(\{D_s(u) : u \in \mathcal{C}\}\big), \qquad a_s(u) = \mathbb{1}\{D_s(u) > \eta_s\}, \tag{10}$$

where Q_q^{higher} is the empirical quantile rounded up to an observed score. Final-scale UFID uses $D_S(u)$ as its score and $a_S(u)$ as its binary decision. We select the latter-half scales $\mathcal{S} = \{\lfloor S/2 \rfloor + 1, \ldots, S\}$ and define $\mathcal{S}^- = \mathcal{S} \setminus \{S\}$. SC-UFID-Gate flags a prompt when the final scale and a strict majority of the selected intermediate scales are anomalous:

$$g(u) = a_S(u)\,\mathbb{1}\left\{\frac{1}{|\mathcal{S}^-|}\sum_{s\in\mathcal{S}^-} a_s(u) > \frac{1}{2}\right\}. \tag{11}$$

SC-UFID-Vote uses the anomalous-scale fraction $V(u) = |\mathcal{S}|^{-1}\sum_{s\in\mathcal{S}} a_s(u)$ as its AUROC score and flags prompts with $V(u) > 1/2$. Gate only filters final-scale alerts; we examine whether it reduces false positives while retaining triggered-input detection. Calibration and evaluation settings are provided in Appendix D.3.

Table 1: **Attack effectiveness and clean generation quality on Infinity-2B (seed 42).** Best and second-best results are shown in bold and underline, respectively, excluding the clean reference. Additional targets and seed-0/1 results are reported in Appendix Tables 14, 15, and 16.

Target	Attack Method	Backdoor Attack Performance			Clean Performance		
		MSE (↓)	LPIPS (↓)	ASR (↑)	FID (↓)	IS (↑)	CLIP (↑)
-	Clean Model	-	-	-	33.87	38.77	25.97
House	Rand-Scale Transfer	<u>0.0656</u>	<u>0.5274</u>	<u>2.57</u>	50.67	22.62	20.50
	Img-Reg Transfer	0.1052	0.7537	0.00	270.88	2.73	14.26
	DreamBooth	0.1154	0.7095	0.00	52.32	20.95	21.02
	Textual Inversion	0.1254	0.7262	0.00	<u>35.77</u>	**38.74**	**25.68**
	Native Poisoned FT	0.1056	0.7194	0.00	39.40	28.60	22.25
	Clean-CE + Attack-CE	0.1128	0.7169	0.00	**34.74**	31.76	23.64
	Native FT + L2-SP	0.0936	0.7144	0.00	38.92	27.47	22.28
	ScaleTrojan (Ours)	**0.0059**	**0.1305**	**99.99**	38.16	<u>37.73</u>	<u>25.42</u>
Dog	Rand-Scale Transfer	0.0907	0.6177	<u>2.60</u>	57.35	24.23	20.16
	Img-Reg Transfer	0.1454	0.7738	0.00	252.58	2.53	14.49
	DreamBooth	0.1242	0.6737	0.00	46.91	23.83	22.15
	Textual Inversion	0.1248	0.6899	0.00	**35.77**	**38.63**	**25.68**
	Native Poisoned FT	0.0995	0.6723	0.00	43.72	26.02	22.63
	Clean-CE + Attack-CE	<u>0.0710</u>	<u>0.6125</u>	0.00	<u>36.46</u>	32.57	23.73
	Native FT + L2-SP	0.1085	0.6772	0.00	42.86	26.54	22.53
	ScaleTrojan (Ours)	**0.0056**	**0.1066**	**100.00**	38.11	<u>38.08</u>	<u>25.39</u>
Cat	Rand-Scale Transfer	<u>0.1061</u>	<u>0.6234</u>	<u>2.16</u>	61.30	22.01	20.66
	Img-Reg Transfer	0.1732	0.8460	0.00	246.70	3.19	14.35
	DreamBooth	0.1574	0.7000	0.00	45.94	23.50	21.75
	Textual Inversion	0.2390	0.7567	0.00	<u>35.77</u>	**38.55**	**25.68**
	Native Poisoned FT	0.1719	0.7318	0.00	52.04	23.13	21.51
	Clean-CE + Attack-CE	0.2605	0.7714	0.00	**35.22**	30.95	23.84
	Native FT + L2-SP	0.1284	0.6834	0.00	49.62	24.73	21.74
	ScaleTrojan (Ours)	**0.0027**	**0.0872**	**100.00**	37.63	<u>37.75</u>	<u>25.48</u>

4 Experiments

4.1 Experimental Setup

Models and protocol. We evaluate ScaleTrojan primarily on Infinity-2B Han et al. (2025) and additionally on STAR Ma et al. (2024). Infinity-2B training uses LAION-Aesthetics v2 Schuhmann et al. (2022), and evaluation uses 10,000 clean and 10,000 triggered prompts per target from MS-COCO 2014 Lin et al. (2014). For readability, the main text reports seed 42; the corresponding seed-0 and seed-1 results are reported in Appendix F.6. Ablations use seed 42.

Baselines and metrics. We compare against transferred diffusion objectives, personalization methods, and native VAR fine-tuning controls. MSE, LPIPS, and ASR measure attack effectiveness; FID, IS, and CLIP Score measure clean utility. Appendix D.2 defines all baselines, while Appendices D.1 and D.3 specify the target and trigger sets, ASR thresholds, training settings, and implementation details.

4.2 Backdoor Implantation Effectiveness

To answer **RQ1**, we compare attack effectiveness and clean generation quality across methods and backbones.

Comparison with competitors. Table 1 reports results on Infinity-2B. ScaleTrojan achieves nearly 100% ASR across all three targets reported in this table, while Rand-Scale Transfer remains below 3% and the other baselines obtain zero ASR. ScaleTrojan also achieves the lowest target MSE and LPIPS. Clean IS and CLIP remain close to the pretrained model, although FID increases.

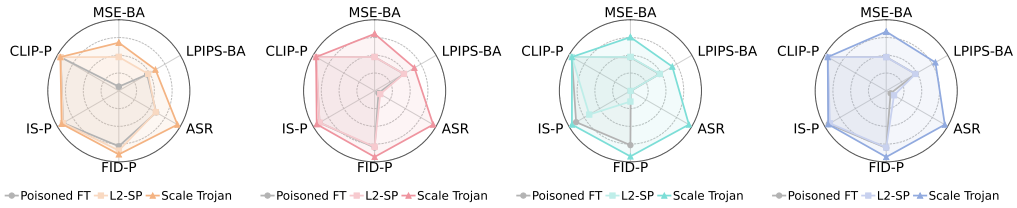


Figure 4: **Cross-target evaluation on STAR.** Radar plots show higher-is-better scores for Cat, Dog, House, and Hello Kitty. Raw metrics and score definitions are provided in Appendix Table 9 and Appendix F.1.

Table 2: **Average attack effectiveness and clean generation quality across injection subspaces.** Best and second-best results are shown in bold and underline, respectively.

Subspace	MSE ↓	LPIPS ↓	ASR (%) ↑	FID ↓	IS ↑	CLIP ↑
CA	**0.0047**	**0.1081**	**100.00**	<u>37.97</u>	37.85	<u>25.43</u>
FFN	0.0102	0.1730	80.90	**37.96**	**38.63**	**25.51**
CA+FFN	0.0072	0.1413	<u>96.82</u>	38.54	<u>38.27</u>	25.34
Full	<u>0.0067</u>	<u>0.1261</u>	95.08	38.73	37.88	25.25

Evaluation on STAR. Figure 4 and Appendix Table 9 report results on another VAR backbone. ScaleTrojan achieves at least 99.80% ASR across all four targets, exceeding Native Poisoned FT and Native FT + L2-SP by more than 36 percentage points, while reducing target MSE and LPIPS. Its clean metrics remain comparable to the pretrained model, supporting the framework's applicability beyond Infinity-2B.

Qualitative comparison. Figure 1 illustrates RQ1: clean prompts retain prompt alignment, while triggered prompts redirect generation toward the target.

4.3 Parameter Subspace Analysis

To answer **RQ2**, we compare module-level adaptation and then restrict updates within CA.

Module-level comparison. Table 2 summarizes the three Infinity-2B targets reported in Table 1. CA achieves the highest average ASR, improving over FFN by 19.10 percentage points with nearly unchanged average FID. Broader CA+FFN and Full adaptation does not improve the overall balance. These results complement the activation observations in Section 3.2.

CA component restriction. Figure 5(c) compares independently trained CA-Q, CA-KV, CA-QKV, and CA-Out configurations on Cat with seed 42. CA-Q provides almost no target control, whereas the other configurations achieve nearly 100% ASR with comparable clean CLIP. Thus, restricted CA subspaces can support effective backdoor learning; this comparison does not establish component necessity. Complete metrics are provided in Appendix Table 8.

4.4 Backdoor Detection

To answer **RQ3**, we compare Final-scale UFID, SC-UFID-Vote, and SC-UFID-Gate across four target-specific models. Each scale is calibrated by the 0.95 higher quantile of 500 clean scores, and evaluation uses $5{,}000$ clean and $5{,}000$ triggered prompts per target (Appendix D.3).

Detection results. Table 3 shows that both cross-scale variants reduce average FPR from 5.00% to 3.75% while retaining 100% TPR. The improvement occurs on Cat, where FPR falls from 10.00% to 5.00%; the other targets remain unchanged. Final-scale UFID nevertheless has higher average AUROC than Vote (1.000 vs. 0.988). The gain is fewer false alarms at the calibrated operating point without lost true positives, rather than improved score separation.

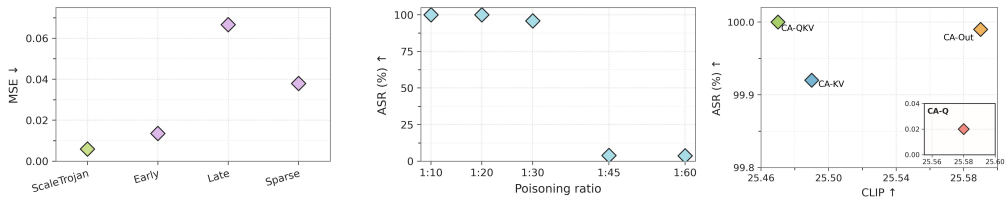


Figure 5: Ablation results for scale-wise target forcing, poisoning ratio, and cross-attention component restriction.

Table 3: **Backdoor detection results on Cat and the average over all evaluated targets.** Best results within each target or average group are shown in bold.

Target	Method	AUROC (↑)	TPR (↑)	FPR (↓)	Precision (↑)	F1 (↑)	Accuracy (↑)
Cat	Final-scale UFID	**1.000**	**100.00**	10.00	90.91	95.24	95.00
Cat	SC-UFID-Vote	0.975	**100.00**	**5.00**	**95.24**	**97.56**	**97.50**
Cat	SC-UFID-Gate	–	**100.00**	**5.00**	**95.24**	**97.56**	**97.50**
Average	**Final-scale UFID**	**1.000**	**100.00**	5.00	95.35	97.59	97.50
Average	**SC-UFID-Vote**	0.988	**100.00**	**3.75**	**96.43**	**98.17**	**98.13**
Average	**SC-UFID-Gate**	–	**100.00**	**3.75**	**96.43**	**98.17**	**98.13**

4.5 Ablation Studies

To further examine **RQ1**, we ablate clean distribution regularization, scale-wise target supervision, and poisoning ratio on House.

Clean distribution regularization. Figure 6 separates target learning from clean preservation. Removing KL retains 97.45% ASR but severely degrades clean generation, increasing FID to 291.60 and reducing CLIP to 9.13. DFT replaces distribution matching with clean CE and achieves only 3.90% ASR. This single-seed ablation is distinct from the multi-seed Clean-CE + Attack-CE baseline in Table 1, although both test clean ground-truth supervision in place of teacher KL. ScaleTrojan retains both target control and clean utility.

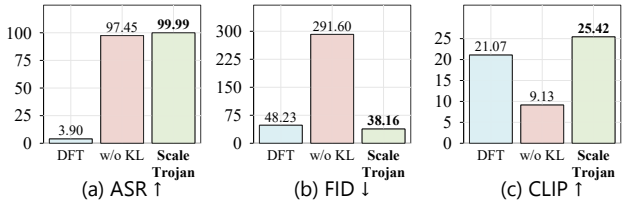


Figure 6: Effect of clean cross-scale distillation.

Scale-wise target supervision. Figure 5(a) compares full supervision with Early, Late, and Sparse scale subsets. Partial supervision retains comparable clean metrics but yields less than 7% ASR, versus nearly 100% for the full method. Full-scale supervision therefore provides stronger target control than the evaluated alternatives.

Poisoning ratio. Figure 5(b) varies the backdoor-to-clean sample ratio. ASR exceeds 95% through 1:30 but falls below 4% at 1:45, where only three backdoor samples remain. Appendix Table 11 explains this discrete-data breakpoint.

5 Conclusion

We systematically study backdoor vulnerabilities in VAR text-to-image generation through attack construction, parameter subspace analysis, and detection. ScaleTrojan combines multi-scale target supervision with clean distribution regularization to establish effective backdoors while preserving benign generation. Parameter subspace comparisons identify CA as an effective injection interface. Cross-scale detection further filters some false alarms without losing true positives at the evaluated operating point.

REFERENCES

Tobias Braun, Jonas Henry Grebe, Hossein Shakibania, Anna Rohrbach, and Marcus Rohrbach. Token by token, compromised: Backdoor vulnerabilities in unified autoregressive models. *arXiv preprint arXiv:2605.19227*, 2026.

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International conference on machine learning*, pp. 1691–1703, 2020.

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*, pp. 4015–4024, 2023a.

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

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883, 2021.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

Zihan Guan, Mengxuan Hu, Sheng Li, and Anil Kumar Vullikanti. Ufid: A unified framework for black-box input-level backdoor detection on diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 27312–27320, 2025.

Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15733–15744, 2025.

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

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, pp. 6840–6851, 2020.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, pp. 3, 2022.

Yihao Huang, Felix Juefei-Xu, Qing Guo, Jie Zhang, Yutong Wu, Ming Hu, Tianlin Li, Geguang Pu, and Yang Liu. Personalization as a shortcut for few-shot backdoor attack against text-to-image diffusion models. In *Proceedings of the AAAI conference on artificial intelligence*, pp. 21169–21178, 2024.

Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11523–11532, 2022.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755, 2014.

Liangwei Lyu, Jiaqi Xu, Jianwei Ding, and Qiyao Deng. When LoRA betrays: Backdooring text-to-image models by masquerading as benign adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8577–8586, 2026.

Xiaoxiao Ma, Mohan Zhou, Tao Liang, Yalong Bai, Tiejun Zhao, Biye Li, Huaian Chen, and Yi Jin. Star: Scale-wise text-conditioned autoregressive image generation. *arXiv preprint arXiv:2406.10797*, 2024.

Yu Pan, Jiahao Chen, Lin Wang, Bingrong Dai, and Wenjie Wang. STEDiff: Revealing the spatial and temporal redundancy of backdoor attacks in text-to-image diffusion models. In *International Conference on Learning Representations*, pp. 52022–52046, 2026.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

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

Ambrish Rawat, Killian Levacher, and Mathieu Sinn. The devil is in the gan: backdoor attacks and defenses in deep generative models. In *European Symposium on Research in Computer Security*, pp. 776–783, 2022.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22500–22510, 2023.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022.

Lukas Struppek, Dominik Hintersdorf, and Kristian Kersting. Rickrolling the artist: Injecting backdoors into text encoders for text-to-image synthesis. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4584–4596, 2023.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

Hao Wang, Shangwei Guo, Jialing He, Kangjie Chen, Shudong Zhang, Tianwei Zhang, and Tao Xiang. Eviledit: Backdooring text-to-image diffusion models in one second. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 3657–3665, 2024a.

Yuqing Wang, Zhijie Lin, Yao Teng, Yuanzhi Zhu, Shuhuai Ren, Jiashi Feng, and Xihui Liu. Bridging continuous and discrete tokens for autoregressive visual generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18596–18605, 2025a.

Zhongqi Wang, Jie Zhang, Shiguang Shan, and Xilin Chen. T2ishield: Defending against backdoors on text-to-image diffusion models. In *European Conference on Computer Vision*, pp. 107–124, 2024b.

Ziyi Wang, Songbai Tan, Gang Xu, Xuerui Qiu, Hongbin Xu, Xin Meng, Ming Li, and Fei Richard Yu. Safe-var: Safe visual autoregressive model for text-to-image generative watermarking. *arXiv preprint arXiv:2503.11324*, 2025b.

Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13294–13304, 2025.

LI Xuhong, Yves Grandvalet, and Franck Davoine. Explicit inductive bias for transfer learning with convolutional networks. In *International conference on machine learning*, pp. 2825–2834, 2018.

Siya Yang, Nanxiang Jiang, Zhaoxin Fan, and Yunfeng Diao. Sace: Concept erasure at the semantic singularity in visual autoregressive models. *arXiv preprint arXiv:2606.15819*, 2026.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022.

Shengfang Zhai, Yinpeng Dong, Qingni Shen, Shi Pu, Yuejian Fang, and Hang Su. Text-to-image diffusion models can be easily backdoored through multimodal data poisoning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 1577–1587, 2023.

Shengfang Zhai, Jiajun Li, Yue Liu, Huanran Chen, Zhihua Tian, Wenjie Qu, Qingni Shen, Ruoxi Jia, Yinpeng Dong, and Jiaheng Zhang. Efficient input-level backdoor defense on text-to-image synthesis via neuron activation variation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15182–15193, 2025.

Qian Zhang, Xiangzi Dai, Ninghua Yang, Xiang An, Ziyong Feng, and Xingyu Ren. Var-clip: Text-to-image generator with visual auto-regressive modeling. *arXiv preprint arXiv:2408.01181*, 2024.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

Xinhao Zhong, Yimin Zhou, Zhiqi Zhang, Junhao Li, Yi Sun, Bin Chen, Shu-Tao Xia, Xuan Wang, and Ke Xu. Closing the safety gap: Surgical concept erasure in visual autoregressive models. In *International Conference on Learning Representations*, pp. 34877–34897, 2026.

CONTENTS

A. Related Work **14**

A.1. Backdoor Attacks and Defenses in Text-to-Image Models 14

A.2. Visual Autoregressive Generation and Security-Relevant Structure 14

B. Symbol Definitions **15**

C. Algorithm of ScaleTrojan **15**

D. Additional Experimental Settings **15**

D.1. Evaluation Metrics 15

D.2. Baseline Methods 18

D.3. Implementation Details 19

E. Detailed Injection-Space Analysis **20**

E.1. Activation, Trigger Response, and Clean Drift 21

F. Additional Experimental Results **23**

F.1. Radar-Plot Score Construction 23

F.2. Attack-Clean Utility Trade-off 24

F.3. Ablation on Clean-Scale Distillation 24

F.4. Robustness to the Loss Weight λ 25

F.5. Additional Target Evaluation 26

F.6. Per-Seed Results for the Main Comparison 26

F.7. Robustness to Different Trigger Forms 27

F.8. Additional Qualitative Results 29

G. Limitations and Future Work **33**

G.1. Limitations 33

G.2. Future Work 33

H. Broader Impacts **33**

A Related Work

A.1 Backdoor Attacks and Defenses in Text-to-Image Models

Backdoor attacks on image generators aim to preserve ordinary generation while activating attacker-specified behavior under a trigger. Early work studied latent triggers in GANs Rawat et al. (2022); subsequent text-to-image attacks have largely focused on diffusion models. These methods poison multimodal trigger–target pairs, redirect noise or denoising trajectories, or manipulate text-conditioning components Zhai et al. (2023); Chou et al. (2023a;b); Struppek et al. (2023); Wang et al. (2024a). STEDiff further studies spatial and temporal redundancy in diffusion backdoor injection Pan et al. (2026). Because these attacks optimize continuous denoising variables or diffusion-specific conditioning pathways, they do not directly specify the discrete multi-scale target sequence predicted by a VAR generator.

Efficient adaptation provides another route for implanting generative backdoors. DreamBooth and Textual Inversion were introduced for subject or token personalization Ruiz et al. (2023); Gal et al. (2022), and later work showed that these mechanisms can be repurposed for few-shot backdoor attacks Huang et al. (2024). MasqLoRA similarly embeds a diffusion backdoor in a separately distributable LoRA adapter Lyu et al. (2026). These studies motivate our personalization and lightweight-adaptation baselines, but their objectives remain tied to diffusion generation rather than next-scale target prediction. More recently, ToBAC establishes backdoor vulnerability in unified autoregressive models that generate image and text tokens in a shared next-token sequence Braun et al. (2026). Its unified next-token setting differs from the next-scale VAR setting studied here, where a fixed target image must be represented and predicted across a hierarchy of visual scales.

Backdoor defenses for text-to-image diffusion models use both internal responses and output behavior. T2IShield detects abnormal assimilation in cross-attention maps, while NaviT2I measures token-induced neuron activation variation during early denoising Wang et al. (2024b); Zhai et al. (2025). UFID instead perturbs an input and detects the unusually consistent outputs produced by triggered queries Guan et al. (2025). These methods establish attention-, activation-, and perturbation-based evidence for diffusion models. Our detector builds specifically on UFID's perturbation-consistency criterion, but examines intermediate reconstructions along the VAR scale sequence and combines their calibrated decisions. It therefore assumes access to intermediate scale predictions, unlike UFID's black-box final-output setting.

A.2 Visual Autoregressive Generation and Security-Relevant Structure

Autoregressive and VQ-based image generators represent images with discrete visual tokens Chen et al. (2020); Esser et al. (2021); Lee et al. (2022); Yu et al. (2022); Sun et al. (2024). VAR replaces raster-order next-token prediction with coarse-to-fine next-scale prediction, while STAR introduces scale-wise text conditioning and Infinity extends the paradigm to high-resolution bitwise prediction Tian et al. (2024); Ma et al. (2024); Han et al. (2025). The resulting supervised object is a sequence of visual representations whose later scales are conditioned on earlier ones, rather than a denoising state or final image alone.

Recent safety studies also show that intervention objectives must account for this scale-wise structure. Safe-VAR studies watermarking and provenance for VAR outputs Wang et al. (2025b). S-VARE addresses concept erasure by aligning selected visual targets while using scale-wise KL matching to preserve the pretrained model's behavior on unrelated prompts Zhong et al. (2026); SACE instead concentrates concept-erasure intervention at the earliest scale Yang et al. (2026). These works pursue removal or protection rather than trigger-conditioned implantation. Nevertheless, S-VARE is technically close to our clean distribution regularization: both match the modified model to a frozen reference across scales. ScaleTrojan couples this preservation principle with triggered learning of a fixed target sequence and evaluates the resulting attack–utility separation.

Text conditions in modern VAR generators are commonly injected through cross-attention, making the adaptation subspace another security-relevant design choice Ma et al. (2024); Zhang et al. (2024); Han et al. (2025). LoRA enables controlled updates to such subspaces Hu et al. (2022), while parameter-space regularization such as L2-SP provides a generic way to limit deviation from pretrained weights Xuhong et al. (2018). Neither mechanism alone determines which subspace best supports triggered target control while retaining clean generation. Despite the studies above, it re-

Algorithm 1: ScaleTrojan Training

Input: Frozen benign VAR model p_{θ_0}, frozen visual tokenizer $\mathcal{E}$, clean training set D_{clean}, trigger τ, attacker target image $x^\star$, loss weight λ

Output: Backdoored VAR model p_{θ_b} and optimized trigger embedding e_τ

Initialize $p_{\theta_b} \leftarrow p_{\theta_0}$
Freeze the pretrained backbone, visual tokenizer, codebook, and inference-time decoding procedure
Initialize e_τ from the embedding of "photo"; freeze all other text-encoder parameters
Enable only the adaptation parameters ϕ specified by the injection configuration and e_τ
Encode and cache the attacker target $z^\star = \mathcal{E}(x^\star) = \{z_1^\star, \ldots, z_S^\star\}$
for each training iteration **do**
  Sample clean prompt-image pairs (y, x) from D_{clean}
  Construct triggered prompts $\tilde{y} = y \oplus \tau$
  Encode y and $\tilde{y}$ into conditions c and $\tilde{c}(e_\tau)$, respectively
  Encode clean images into visual tokens $\mathbf{z} = \mathcal{E}(x) = \{z_1, \ldots, z_S\}$
  Initialize $L_{\text{atk}} \leftarrow 0$ and $L_{\text{dist}} \leftarrow 0$
  for $s = 1, \ldots, S$ **do**
    Compute triggered prediction $p_{\theta_b}(z_s^\star \mid z_{<s}^\star, \tilde{c})$
    $L_{\text{atk}} \leftarrow L_{\text{atk}} - \log p_{\theta_b}(z_s^\star \mid z_{<s}^\star, \tilde{c})$
    Compute clean teacher distribution $p_{\theta_0}(\cdot \mid z_{<s}, c)$
    Compute clean student distribution $p_{\theta_b}(\cdot \mid z_{<s}, c)$
    $L_{\text{dist}} \leftarrow L_{\text{dist}} + D_{\text{KL}}(p_{\theta_0}(\cdot \mid z_{<s}, c) \,\|\, p_{\theta_b}(\cdot \mid z_{<s}, c))$
  end for
  $L_{\text{total}} \leftarrow L_{\text{atk}} + \lambda L_{\text{dist}}$
  Mask text-encoder gradients to retain only $\nabla_{e_\tau} L_{\text{total}}$
  Update ϕ and e_τ with their respective AdamW parameter groups
end for
return p_{θ_b} and e_τ

mains unclear how fixed-target backdoors should be learned in next-scale VAR generators, how the adaptation subspace affects their attack–utility trade-off, and whether scale-wise predictions provide useful evidence for triggered-input detection.

B Symbol Definitions

For clarity, Table 4 summarizes the notation used for VAR generation, target tokenization, ScaleTrojan training, and evaluation. The notation covers clean and triggered prompts, multi-scale visual tokens, target sequences, benign and backdoored models, training losses, evaluation metrics, and adaptation subspaces.

C Algorithm of ScaleTrojan

Algorithm 1 first tokenizes the attacker target into VAR's multi-scale prediction space, addressing the target-mapping mismatch. It then forces the triggered prediction along the coarse-to-fine token chain while distilling clean next-scale distributions from the frozen benign model, preserving normal generation behavior without modifying the tokenizer, codebook, or decoder.

D Additional Experimental Settings

D.1 Evaluation Metrics

We evaluate attack effectiveness on triggered prompts and clean utility on clean prompts. The attack metrics compare generated images against the fixed attacker target $x^\star$, while the clean metrics evaluate whether benign prompt-conditioned generation is preserved.

Table 4: Summary of notation used in this paper.

Symbol	Description
Prompt and conditioning	
y	Clean text prompt.
τ	Trigger token or phrase inserted by the attacker.
$\tilde{y} = y \oplus \tau$	Triggered prompt obtained by inserting τ into y.
c	Text condition encoded from the clean prompt y.
$\tilde{c}$	Text condition encoded from the triggered prompt $\tilde{y}$.
<extra_id_99>	Default trigger token; evaluated alternatives are <extra_id_0>, "alors", "celor", and "24".
VAR generation and target mapping	
$\mathbf{z} = \{z_1, z_2, \ldots, z_S\}$	Multi-scale discrete visual token sequence used by VAR.
z_s	Visual token representation at scale s.
$z_{<s} = \{z_1, \ldots, z_{s-1}\}$	Visual prefix containing all scales before s.
S	Number of visual scales in the autoregressive generation chain.
$p_\theta(\mathbf{z} \mid c)$	Conditional VAR distribution over the full multi-scale token sequence.
$p_\theta(z_s \mid z_{<s}, c)$	Next-scale predictive distribution conditioned on previous scales and text condition.
$\mathcal{E}$	Frozen visual tokenizer used to map images into multi-scale visual tokens.
$x^\star$	Attacker-specified target image activated by the trigger.
$z^\star = \mathcal{E}(x^\star) = \{z_1^\star, \ldots, z_S^\star\}$	Multi-scale target token sequence obtained by encoding $x^\star$.
$z_s^\star$	Target token at scale s.
$z_{<s}^\star = \{z_1^\star, \ldots, z_{s-1}^\star\}$	Target prefix used when forcing prediction at scale s.
$\hat{x}^{(s)}$	Decoded image reconstructed from predictions up to scale s.
Models and training objectives	
p_{θ_0}	Frozen pretrained benign VAR model.
p_{θ_b}	Backdoored VAR model after attack fine-tuning.
θ_0	Parameters of the pretrained benign model.
θ_b	Parameters of the backdoored model.
$\mathcal{L}_{\text{mse}}$	Naive image-space regression loss for diffusion-style transfer.
$\mathcal{L}_{\text{rand}}$	Naive random-scale transfer objective.
$\mathcal{L}^{(s)}$	Loss applied at a sampled visual scale in random-scale transfer.
$\ell_{\text{atk}}^{(s)}$	Per-scale triggered attack loss for predicting $z_s^\star$.
L_{atk}	Triggered cross-scale attack objective over D_{trig}.
L_{dist}	Clean cross-scale distillation loss against p_{θ_0}.
L_{total}	Overall ScaleTrojan objective.
$D_{\text{KL}}(\cdot \Vert \cdot)$	KL divergence used for scale-wise distribution distillation.
λ	Weight balancing triggered attack strength and clean-behavior preservation.
Data, evaluation, and adaptation	
D_{trig}	Set of triggered training samples used for the attack objective.
D_{clean}	Set of clean training samples used for distillation.
(y, x)	Clean prompt-image pair used in clean distillation.
$\hat{x}_i$	Generated image under the i-th triggered prompt in evaluation.
N_{trig}	Number of triggered prompts used to compute ASR.
ASR	Attack success rate; counted by an LPIPS threshold in the evaluation protocol.
LPIPS$(\hat{x}_i, x^\star)$	Perceptual distance between a triggered output and the target image.
MSE	Pixel-level target similarity metric for triggered outputs.
FID	Clean-generation distribution quality metric.
IS	Inception Score for clean-generation quality.
CLIP Score	Text-image alignment metric for clean-generation quality.
Full	Broad LoRA adaptation across CA, FFN, and SA modules.
CA	Cross-attention adaptation subspace.
FFN	Feed-forward adaptation subspace.
CA+FFN	Joint cross-attention and feed-forward adaptation subspace.
SA	Self-attention modules listed in the main results table.
LoRA	Lightweight adaptation mechanism used for CA, FFN, and CA+FFN settings.

Attack effectiveness metrics. For a triggered generated image $\hat{x}_i$ and target image $x^\star$, MSE measures pixel-level target distance:

$$\text{MSE} = \frac{1}{N_{\text{trig}}} \sum_{i=1}^{N_{\text{trig}}} \frac{1}{HWC} \left\| \hat{x}_i - x^\star \right\|_2^2, \tag{12}$$

where H, W, and C denote image height, width, and channels. Lower MSE indicates closer pixel-level agreement with the target.

LPIPS Zhang et al. (2018) measures perceptual distance between triggered outputs and the target:

$$\text{LPIPS} = \frac{1}{N_{\text{trig}}} \sum_{i=1}^{N_{\text{trig}}} d_{\text{LPIPS}}(\hat{x}_i, x^\star). \tag{13}$$

Lower LPIPS indicates higher perceptual similarity to $x^\star$.

We use an LPIPS-based attack success rate. A triggered generation is counted as successful when its perceptual distance to the target is below the backbone–target-specific threshold $\delta_{b,t}$:

$$\text{ASR} = \frac{1}{N_{\text{trig}}} \sum_{i=1}^{N_{\text{trig}}} \mathbb{I}\left[d_{\text{LPIPS}}(\hat{x}_i, x^\star) < \delta_{b,t}\right]. \tag{14}$$

The threshold is an operational criterion for fixed-image recovery: a success must be perceptually close to the specified target, rather than merely contain the same semantic category. We use $\delta_{b,t} = 0.2$ as a conservative default for this close-match criterion. The threshold is fixed before comparing attack methods and is shared by all methods within a backbone–target setting, so it is not selected separately to favor a particular method. We additionally report continuous MSE and LPIPS values; consequently, the comparison does not rely on the binary ASR threshold alone.

For STAR, we use $\delta_{b,t} = 0.3$ for the Dog and House targets and retain 0.2 for Cat and Hello Kitty. The Dog and House target images contain more complex structure and texture, while STAR has lower generation fidelity than Infinity-2B; consequently, visibly target-aligned outputs in these two settings can have LPIPS values above 0.2. Using the default threshold would therefore count recognizable target reconstructions as failures. The relaxed threshold is still applied uniformly to every method for the same STAR target, preserving within-target fairness. Because the operating point differs, absolute ASR values across targets should be interpreted together with their continuous LPIPS values and the stated threshold. ASR is reported as a percentage, and higher is better.

Clean utility metrics. FID Heusel et al. (2017) measures the distance between clean generated images and reference real images in Inception feature space:

$$\text{FID} = \|\mu_r - \mu_g\|_2^2 + \text{Tr}\left(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}\right), \tag{15}$$

where (μ_r, Σ_r) and (μ_g, Σ_g) are the mean and covariance of real and generated image features, respectively. Lower FID is better.

Inception Score Salimans et al. (2016) measures the recognizability and diversity of clean generated images:

$$\text{IS} = \exp\left(\mathbb{E}_x D_{\text{KL}}\left(p(\ell \mid x) \,\|\, p(\ell)\right)\right), \tag{16}$$

where ℓ denotes the Inception classifier label. Higher IS is better.

CLIP Score Radford et al. (2021) measures prompt-image alignment under clean prompts:

$$\text{CLIPScore} = \frac{1}{N_{\text{clean}}} \sum_{i=1}^{N_{\text{clean}}} \cos\left(f_{\text{text}}(y_i), f_{\text{image}}(\hat{x}_i^{\text{clean}})\right), \tag{17}$$

where N_{clean} is the number of clean evaluation prompts, y_i is a clean prompt, and $\hat{x}_i^{\text{clean}}$ is the corresponding clean generated image. Higher CLIP Score is better.

Metric reporting protocol. MSE, LPIPS, and ASR are computed on triggered prompts, while FID, IS, and CLIP Score are computed on clean generations to evaluate benign utility preservation. For each fixed-target setting, the same attacker target $x^\star$ is used for all triggered prompts. The evaluated target pool comprises *House*, *Dog*, *Cat*, *Hat*, *Hello Kitty*, and *Face*. The main Infinity-2B comparison reports House, Dog, and Cat, while the STAR comparison additionally reports Hello Kitty; claims tied to a particular table are restricted to the targets explicitly shown in that table. We use $10{,}000$ clean prompts and $10{,}000$ triggered prompts per target under the main metric protocol. FID is computed against real images from the MS-COCO 2014 validation split. Main comparative experiments are run with seeds $\{0, 1, 42\}$, while ablation experiments use seed 42 only.

D.2 BASELINE METHODS

We compare ScaleTrojan with Rand-Scale Transfer Chou et al. (2023b;a); Ho et al. (2020), Img-Reg Transfer Zhai et al. (2023); Chou et al. (2023a), DreamBooth Ruiz et al. (2023), Textual Inversion Gal et al. (2022), and native VAR fine-tuning controls, including L2-SP Xuhong et al. (2018), under the same experimental protocol whenever applicable. These comparisons test whether diffusion-style objectives or generic text-to-image adaptation are sufficient for VAR backdoor learning, and isolate the contributions of target tokenization, scale-wise target forcing, clean-scale distillation, and injection subspace selection.

Rand-Scale Transfer. Rand-Scale Transfer adapts diffusion-style random timestep supervision to the VAR setting by applying attack supervision to randomly selected visual scales, instead of enforcing the full target sequence along the coarse-to-fine chain. At each iteration, it first samples the number of supervised scales $K \in \{1, 2, 3\}$, then samples K scales uniformly from the visual scale set without replacement. The attack loss is computed only on the selected scales and averaged over them. It uses the same trigger τ, attacker target $x^\star$, victim model, and evaluation protocol as ScaleTrojan, and is designed to test the scale-dependency mismatch.

Img-Reg Transfer. Img-Reg Transfer adapts diffusion-style image-level target regression to VAR using latent-level regression, rather than first mapping $x^\star$ into the multi-scale target sequence $z^\star$. After predictions over the visual scales are obtained, the predicted bit/code representations from the selected or full set of scales are converted back to continuous latent codes, interpolated to the final latent resolution, and aggregated. The loss is an MSE loss between this reconstructed latent feature and the VAE latent feature of the target image, so the regression is applied after scale-wise prediction in the continuous VAE latent space rather than as pixel-level regression or CE over target tokens.

DreamBooth. DreamBooth is included as a personalization-style baseline adapted to the fixed-target backdoor setting. It tests whether concept binding from a small number of target-related samples can induce stable trigger-conditioned generation in VAR, without explicitly enforcing the multi-scale target token sequence $z^\star$.

Textual Inversion. Textual Inversion is included as a lightweight adaptation baseline that learns a token embedding or textual concept representation for the target or trigger condition. Unlike ScaleTrojan, it does not impose scale-wise target token forcing or explicitly follow VAR's causal coarse-to-fine dependency.

Native Poisoned Fine-Tuning (Full-Scale CE). Native Poisoned Fine-Tuning uses the original VAR full-scale cross-entropy training objective on poisoned trigger–target pairs. It relies only on native next-scale supervision to establish the mapping from the trigger to the target image and does not use ScaleTrojan's clean teacher KL constraint. This baseline tests whether standard poisoned fine-tuning alone is sufficient for stable fixed-target backdoor learning in VAR.

Clean-CE + Attack-CE. Clean-CE + Attack-CE retains the full-scale attack CE on triggered target pairs but replaces clean teacher KL with ordinary ground-truth CE on clean image–text pairs. It therefore anchors the model only to the observed clean labels rather than to the frozen teacher's complete predictive distribution at every scale. This baseline tests whether conventional clean supervision can substitute for ScaleTrojan's function-space distillation constraint.

Native FT + L2-SP. Native FT + L2-SP augments native poisoned fine-tuning with L2-SP regularization toward the pretrained parameters. It constrains movement in parameter space without matching the pretrained model's scale-wise predictive distributions. This baseline tests whether proximity to the pretrained weights can replace ScaleTrojan's function-space KL constraint.

Direct Fine-Tuning (single-seed ablation). Direct Fine-Tuning replaces scale-wise KL distillation on clean samples with standard CE on each clean sample's own VAE token/bit targets. The triggered branch and trainable adaptation subspace are kept the same as ScaleTrojan, using the default CA LoRA modules and the trigger-token embedding unless otherwise specified. Conceptually,

this ablation probes the same clean-label alternative as Clean-CE + Attack-CE, but it is a separate ablation checkpoint evaluated only with seed 42; Clean-CE + Attack-CE is the corresponding baseline in the main multi-seed comparison. Their numerical results should therefore be interpreted within their respective experimental protocols rather than conflated.

w/o KL. w/o KL removes the clean-side scale-wise distillation term and retains target fitting under the triggered objective. This ablation evaluates the role of L_{dist} in preserving clean generation behavior while learning the trigger-to-target mapping.

Injection subspaces. We compare the reported CA, FFN, CA+FFN, and Full LoRA configurations, jointly optimizing the trigger-token embedding in each setting. Full denotes broad module coverage rather than unrestricted backbone fine-tuning. The projection selections, reported parameter counts, and component restrictions are detailed in Appendix E.

Together, these baselines evaluate whether diffusion-style objectives transfer to VAR, whether image-level target alignment is sufficient, whether scale-wise target forcing and clean-scale distillation are necessary, and where the backdoor should be injected in the VAR architecture.

D.3 Implementation Details

This subsection complements the main experimental setup with implementation details needed for reproducibility. We summarize the victim model and tokenizer, data and poisoning protocol, optimization and adaptation choices, evaluation protocol, compute resources, and hyperparameters used in our experiments.

Victim model and tokenizer. We use Infinity-2B Han et al. (2025) as the victim VAR-based text-to-image generator, with checkpoint `weights/infinity_2b_reg.pth` and FLAN-T5-XL text encoder `weights/flan-t5-xl`; cross-backbone evaluation uses STAR Ma et al. (2024). ScaleTrojan keeps the visual tokenizer, VAE/codebook, Infinity backbone, all T5 encoder parameters except the trigger-associated embedding rows, and the inference-time decoding procedure unchanged. In the default setting, the trigger-token embedding for `<extra_id_99>` is initialized from the token "photo" and updated with a gradient mask. Each attacker target image $x^\star$ is mapped into $S = 13$ visual scales at 1024×1024 resolution:

$$1{\times}1, 2{\times}2, 4{\times}4, 6{\times}6, 8{\times}8, 12{\times}12, 16{\times}16, 20{\times}20, 24{\times}24, 32{\times}32, 40{\times}40, 48{\times}48, 64{\times}64.$$

Training data and poisoning protocol. Training prompts and image-text pairs are drawn from LAION-Aesthetics v2 Schuhmann et al. (2022). Poisoned samples are constructed by inserting a trigger τ into selected prompts to form $\tilde{y} = y \oplus \tau$. A poison ratio of $a{:}b$ denotes a triggered backdoor samples for every b clean samples; the default ratio is 1:10. The default trigger is `<extra_id_99>`; additional trigger-form experiments use `<extra_id_0>`, the natural-word strings "alors" and "celor", and the numeric string "24". The evaluated target pool comprises *House*, *Dog*, *Cat*, *Hat*, *Hello Kitty*, and *Face*. Each fixed-target setting uses one attacker-specified image $x^\star$, which is converted to RGB, resized to 1024×1024, converted to a tensor, and normalized with mean/std $0.5/0.5$, mapping it to $[-1, 1]$. The same preprocessing pipeline is used for all target images before tokenizing $x^\star$ into $z^\star$.

Optimization and adaptation. ScaleTrojan optimizes $L_{\text{total}} = L_{\text{atk}} + \lambda L_{\text{dist}}$, where the triggered branch enforces the target token sequence and the clean branch distills scale-wise predictions from p_{θ_0}. Consistent with Eq. 7, the loss weight is $\lambda = 30$. We use AdamW with betas $(0.9, 0.999)$, weight decay 0, LoRA learning rate 1×10^{-4}, trigger-token embedding learning rate 0.1, bf16 mixed precision, and gradient clipping with max norm 1.0. Main comparative experiments are run with seeds 0, 1, and 42; all ablation studies use seed 42. The per-GPU batch size is 1, and gradients are accumulated for 4 steps. Training runs for 260 epochs. LoRA uses rank $r = 32$, $\alpha_{\text{LoRA}} = 64$, dropout 0, and does not train bias terms.

Evaluation protocol. We evaluate on MS-COCO 2014 validation prompts Lin et al. (2014), using 10,000 clean prompts and 10,000 triggered prompts per target under the main metric protocol. MSE, LPIPS, and ASR are computed on triggered generations, using LPIPS threshold 0.2 by default and

Table 5: Implementation details and hyperparameters used in our experiments.

Item	Value
Victim model	Infinity-2B
Model checkpoint / version	`infinity_2b_reg.pth`
Visual tokenizer / codebook	`infinity_vae_d32_reg.pth`
Pretrained backbone	Frozen; trainable LoRA modules are inserted according to the injection subspace
Text encoder	FLAN-T5-XL `flan-t5-xl`; frozen except the trigger-token embedding row
Decoding procedure	Unchanged
Generation resolution	1024×1024
Number of visual scales S	13
Training prompts	LAION-Aesthetics v2
Evaluation prompts	MS-COCO 2014 validation prompts; 10,000 clean and 10,000 triggered prompts per target
Triggers τ	`<extra_id_99>` (default), `<extra_id_0>`, "alors", "celor", and "24"
Target images $x^\star$	*House*, *Dog*, *Cat*, *Hat*, *Hello Kitty* (`hello_kitty`), and *Face*
Target tokenization	$z^\star = \mathcal{E}(x^\star) = \{z_1^\star, \ldots, z_S^\star\}$ using the frozen visual tokenizer
Poison ratio	Backdoor samples : clean samples; 1:10 by default
Optimizer	AdamW, $\beta = (0.9, 0.999)$, weight decay 0
Learning rate	1×10^{-4} for LoRA; 0.1 for trigger embedding
Batch size / gradient accumulation	1 per GPU; 1 GPU; 4 accumulation steps
Training steps / epochs	260 epochs
Loss weight	$\lambda = 30$ in Eq. 7
LoRA rank / scaling / dropout	$r = 32$, $\alpha_{\text{LoRA}} = 64$, 0; bias is not trained
Trainable modules	Default CA LoRA modules and trigger-token embedding; ablations use Full, CA, FFN, and CA+FFN
Sampling hyperparameters	CFG $= 4.0$, temperature $T = 0.5$, top-k $= 900$, top-p $= 0.0$, one image per prompt
ASR threshold	LPIPS < 0.2 by default; LPIPS < 0.3 for STAR Dog and House
Detection calibration	500 clean prompts per target-specific model; per-scale 0.95 higher quantile
Detection evaluation	5,000 clean and 5,000 triggered prompts per target
FID reference set	MS-COCO 2014 validation real images
GPU	1 NVIDIA RTX 4090 GPU
Random seeds	Main comparisons: `0,1,42`; ablations: `42`

0.3 for STAR Dog and House as described in Appendix D.1. FID, IS, and CLIP Score are computed on clean generations to measure benign utility preservation, with FID computed against real images from the MS-COCO 2014 validation split. Sampling uses one image per prompt, shared prompt subsets and matched seeds across methods, CFG scale 4.0, sampling temperature $T = 0.5$, top-k 900, top-p 0.0, `pn=1M`, and square aspect ratio $h/w = 1.0$. The main Infinity-2B tables report *House*, *Dog*, and *Cat*; broader target experiments additionally cover *Hat*, *Hello Kitty*, and *Face*, with each result discussed only for the setting in which it was measured.

Detection evaluation protocol. For each of the four target-specific backdoored models, we calibrate every evaluated scale using 500 clean prompt scores. The scale threshold is the 0.95 empirical quantile with the higher interpolation rule in Eq. 10. Detection is then evaluated separately for each target on 5,000 clean prompts and 5,000 triggered prompts. The original prompt and its textual probes use a shared sampling seed, while each prompt variant follows its own autoregressive visual prefixes. Final-scale UFID, SC-UFID-Vote, and SC-UFID-Gate use the same calibration prompts and evaluation prompts; the latter-half scale set and voting rules follow Section 3.3. The reported TPR, FPR, precision, F1, and accuracy use the calibrated binary decisions, while AUROC is computed from the corresponding ranking score when defined. The quantile controls the calibration operating point but does not guarantee an exact 5% FPR on unseen clean prompts.

Compute resources and parameter efficiency. Experiments are reported on 1 NVIDIA RTX 4090 GPU. The injection-subspace study compares *Full*, *CA*, *FFN*, and *CA+FFN* to analyze parameter efficiency together with attack effectiveness and clean utility. The reported LoRA parameter counts are 44.03M, 14.67M, 20.97M, and 35.64M, respectively, and the trigger token contributes 2,048 additional trainable embedding parameters.

E Detailed Injection-Space Analysis

Adaptation subspaces and parameter accounting. The implementation restricts every injection-space selector to modules inside the transformer blocks. CA comprises the cross-attention projections `ca.mat_q`, `ca.mat_kv`, and `ca.proj`; FFN comprises `ffn.fc1`, `ffn.fc2`, and `ffn.fcg`; and SA comprises the self-attention projections `(sa|attn).mat_qkv` and `(sa|attn).proj`. The combined selectors are the corresponding unions, as detailed in Table 6.

Table 6: Injection-space definitions used by the implementation. Every selector is restricted to transformer blocks. LoRA parameter counts are shown for the evaluated configurations for which they were reported; SA and SA+FFN were defined in code but not evaluated.

Variant	Projection selection	LoRA params
CA-Q	Block CA: `ca.mat_q`	4.19M
CA-KV	Block CA: `ca.mat_kv`	6.29M
CA-Out	Block CA: `ca.proj`	4.19M
CA-QKV	Block CA: `ca.mat_q`, `ca.mat_kv`	10.49M
CA	Block CA: `ca.mat_q`, `ca.mat_kv`, `ca.proj`	14.67M
FFN	Block FFN: `ffn.fc1`, `ffn.fc2`, `ffn.fcg`	20.97M
SA	Block SA: `(sa|attn).mat_qkv`, `(sa|attn).proj`	–
CA+FFN	Union of the CA and FFN selections	35.64M
SA+FFN	Union of the SA and FFN selections	–
Full	Union of the SA, CA, and FFN selections	44.03M

Table 7: Complete reported module-level results from Table 1 using seed 42. MSE, LPIPS, and ASR evaluate triggered generations; FID, IS, and CLIP evaluate clean generations. Configuration names follow the selector definitions in Table 6.

Target	Variant	MSE ↓	LPIPS ↓	ASR (%) ↑	FID ↓	IS ↑	CLIP ↑
House	CA	0.0059	0.1305	99.99	38.16	37.73	25.42
	FFN	0.0146	0.1960	67.73	37.37	38.68	25.54
	CA+FFN	0.0092	0.1770	90.89	38.79	39.04	25.42
	Full	0.0107	0.1768	89.50	39.06	38.16	25.33
Dog	CA	0.0056	0.1066	100.00	38.11	38.08	25.39
	FFN	0.0104	0.1512	96.10	38.37	38.92	25.44
	CA+FFN	0.0089	0.1354	99.70	38.43	37.92	25.22
	Full	0.0052	0.0864	99.87	38.65	37.36	25.18
Cat	CA	0.0027	0.0872	100.00	37.63	37.75	25.48
	FFN	0.0057	0.1717	78.86	38.15	38.29	25.54
	CA+FFN	0.0035	0.1114	99.87	38.39	37.85	25.37
	Full	0.0042	0.1150	95.86	38.47	38.11	25.23

All reported counts exclude the jointly optimized trigger-token embedding, which contributes $2{,}048$ additional parameters. The pretrained backbone and all other text-encoder parameters remain frozen.

Complete reported metrics and scope. Table 7 reproduces the seed-42 module-level House, Dog, and Cat results from the main comparison, including all six attack and clean metrics. Table 8 records the component-level ablation measurements on Cat with seed 42. The component results are single-seed measurements, so neither multi-seed variability nor cross-target generality is inferred from that table. The fine-grained results support a comparison of injection interfaces within their stated evaluation scope.

What component restriction identifies. Each variant is trained separately together with the trigger embedding. Successful restricted training therefore establishes that the selected subspace can learn the attack under the specified optimization protocol; it does not show that the same component is necessary in a model trained with a larger subspace. Conversely, CA-Q's failure does not establish that visual queries are irrelevant to successful CA-KV or CA-Out models, whose pretrained query projections remain active. Infinity implements keys and values through a joint `mat_kv` projection, so the current comparison cannot distinguish changes in key-based routing from changes in value content. The mechanistic interpretation is consequently a collaboration between the learned textual code and the conditional transformation/write-out interface, rather than exclusive storage in KV.

E.1 Activation, Trigger Response, and Clean Drift

This subsection defines the internal measurements in Figure 3. Let f_0 denote the frozen pretrained Infinity-2B model and f_m the model adapted in subspace $m \in \{\mathrm{CA}, \mathrm{FFN}, \mathrm{CA}+\mathrm{FFN}, \mathrm{Full}\}$, in-

Table 8: Component-level results on Cat (seed 42). These runs are separate from the module-level CA checkpoint.

Variant	Params	MSE ↓	LPIPS ↓	ASR (%) ↑	FID ↓	IS ↑	CLIP ↑
CA-Q	4.19M	0.01362	0.32538	0.02	36.01	38.04	25.58
CA-KV	6.29M	0.00476	0.10837	99.92	37.14	37.65	25.49
CA-QKV	10.49M	0.00381	0.08685	100.00	37.11	37.28	25.47
CA-Out	4.19M	0.00430	0.11547	99.99	36.49	38.45	25.59

cluding its jointly optimized trigger embedding. We use $N = 50$ clean prompts y_i and their triggered counterparts $\tilde{y}_i = y_i \oplus \tau$, with $\tau =$ <extra_id_99>. The analysis covers $S = 13$ visual scales and $L = 32$ transformer blocks.

Recorded activation. We define $H_{m,\ell,s}(y; z_{<s}) \in \mathbb{R}^{n_s \times d}$ as the residual-stream representation after the complete ℓ-th transformer block at scale s, conditioned on prompt y and visual token prefix $z_{<s}$. Here, $n_s = h_s w_s$ is the number of spatial tokens at that scale and d is the hidden dimension. The recorded tensor includes all spatial tokens and hidden dimensions; it is a block output, rather than an activation internal to CA or FFN. We index blocks by $\ell = 1, \ldots, L$; the figure uses zero-based layer labels. The heatmap values are normalized differences between these tensors, not raw activation magnitudes.

Clean drift. For each clean prompt, we first sample a visual token trajectory $\mathbf{z}_i^0$ from f_0. Both models are then evaluated on the same clean prompt and the same frozen-model prefix $z^0_{i,<s}$. The relative drift is

$$d_{i,m,\ell,s} = \frac{\left\| H_{m,\ell,s}(y_i; z^0_{i,<s}) - H_{0,\ell,s}(y_i; z^0_{i,<s}) \right\|_F}{\max\left(\left\| H_{0,\ell,s}(y_i; z^0_{i,<s}) \right\|_F, 10^{-12} \right)}. \tag{18}$$

The Frobenius norm is taken jointly over all spatial tokens and hidden dimensions. Fixing the visual trajectory controls for differences in sampled token histories, so the comparison measures representation changes under matched clean inputs. A value of 0.05 means that the difference norm is 5% of the reference norm; it does not indicate the fraction of neurons that changed.

Trigger response ("triggered target gain" in the figure). For each adapted model f_m, we first sample its own clean trajectory $\mathbf{z}_i^{m,c}$ under y_i, then replay that trajectory under both y_i and $\tilde{y}_i$. The trigger-induced relative shift is

$$g_{i,m,\ell,s} = \frac{\left\| H_{m,\ell,s}(\tilde{y}_i; z^{m,c}_{i,<s}) - H_{m,\ell,s}(y_i; z^{m,c}_{i,<s}) \right\|_F}{\max\left(\left\| H_{m,\ell,s}(y_i; z^{m,c}_{i,<s}) \right\|_F, 10^{-12} \right)}. \tag{19}$$

Thus, clean drift compares two models along the pretrained model's clean trajectory, whereas trigger response compares two prompt conditions within one adapted model along its own clean trajectory. The latter measures the magnitude of the internal response to the trigger. It does not use the target image or target tokens and therefore does not measure the direction of target alignment. Target control is assessed separately by the generation metrics.

Averaging and figure construction. We compute the relative differences per prompt before averaging:

$$D_{m,\ell,s} = \frac{1}{N} \sum_{i=1}^{N} d_{i,m,\ell,s}, \qquad G_{m,\ell,s} = \frac{1}{N} \sum_{i=1}^{N} g_{i,m,\ell,s}. \tag{20}$$

The top-left heatmaps show $G_{m,\ell,s}$, with rows corresponding to layers and columns to visual scales. The top-right scale-wise summaries average uniformly over layers, giving $L^{-1} \sum_\ell G_{m,\ell,s}$ and $L^{-1} \sum_\ell D_{m,\ell,s}$. Overall summaries average uniformly over all layer–scale pairs:

$$\overline{G}_m = \frac{1}{LS} \sum_{\ell=1}^{L} \sum_{s=1}^{S} G_{m,\ell,s}, \qquad \overline{D}_m = \frac{1}{LS} \sum_{\ell=1}^{L} \sum_{s=1}^{S} D_{m,\ell,s}. \tag{21}$$

F Additional Experimental Results

This section provides additional analyses omitted from the main paper due to space limitations. Unless otherwise stated, main comparative experiments are run with seeds 0, 1, and 42, whereas all ablation experiments use the single seed 42. The results focus on seven aspects: (i) cross-backbone generalization, (ii) the attack-clean utility trade-off, (iii) the necessity of clean-scale distillation, (iv) robustness to the distillation loss weight λ, (v) additional target coverage, (vi) per-seed main-comparison results, and (vii) robustness and generalization to different trigger forms. Together, these analyses further examine the central claim that effective VAR backdoor learning must be aligned with both the discrete multi-scale prediction space and the coarse-to-fine autoregressive path: triggered target control should be learned through scale-wise target prediction, while benign behavior preservation should constrain the clean scale-wise predictive distributions rather than only the final generated images.

F.1 Radar-Plot Score Construction

To visualize heterogeneous metrics on a common scale, we convert all radar-chart dimensions into higher-is-better scores. These transformed scores are used only for visualization; all quantitative comparisons use the original metrics reported in the corresponding tables. The transformations are applied independently to each target category.

For each lower-is-better attack metric $q \in \{\mathrm{MSE}, \mathrm{LPIPS}\}$, we first define a target-specific reference

$$b_t^q = \min_{m \in \mathcal{B}} x_{m,t}^q, \tag{22}$$

where $x_{m,t}^q$ is the original value of method m on target t, and $\mathcal{B}$ contains Native Poisoned FT and Native FT + L2-SP. We then compute the baseline-anchored score

$$S_{m,t}^q = \frac{b_t^q}{b_t^q + x_{m,t}^q}. \tag{23}$$

Under this transformation, the strongest baseline receives a score of 0.5, a method outperforming the strongest baseline receives a score above 0.5, and the score approaches 1 only as the raw error approaches zero. We denote these two dimensions as MSE-BA and LPIPS-BA.

ASR is converted from a percentage to a fraction using

$$S_{m,t}^{\mathrm{ASR}} = \frac{x_{m,t}^{\mathrm{ASR}}}{100}. \tag{24}$$

For each clean-performance metric $q \in \{\mathrm{FID}, \mathrm{IS}, \mathrm{CLIP}\}$, we measure preservation relative to the clean model:

$$P_{m,t}^q = \max\left(0, 1 - \frac{\left|x_{m,t}^q - c^q\right|}{|c^q|}\right), \tag{25}$$

where c^q is the corresponding clean-model result. These dimensions are denoted as FID-P, IS-P, and CLIP-P. This symmetric deviation-based formulation treats both increases and decreases relative to the clean model as changes in clean behavior; it should therefore be interpreted as a preservation score rather than an absolute image-quality score.

All transformed scores lie in $[0, 1]$, and larger values indicate better performance. The radar plots use a radial upper limit of 1.05 only to provide visual headroom. Because the MSE-BA and LPIPS-BA anchors are target-specific, the plots are intended to compare methods within each target rather than absolute task difficulty across different targets.

Table 11 shows a sharp ASR drop between $1{:}30$ and $1{:}45$ because the ablation uses discrete training sets. At $1{:}45$, only three triggered backdoor samples remain. This provides too little trigger–target supervision relative to the clean data to establish a stable backdoor, reducing ASR from 95.88% at $1{:}30$ to 3.94%. The clean metrics remain stable because reducing the triggered subset does not remove clean supervision. Thus, the observed breakpoint reflects the small absolute number of backdoor examples, rather than evidence for a universal critical ratio at exactly $1{:}45$.

Table 9: **Cross-target evaluation on STAR using seeds** $0, 1, 42$**.** MSE, LPIPS, and ASR evaluate triggered generations; FID, IS, and CLIP evaluate clean generations. ASR uses an LPIPS threshold of 0.3 for Dog and House and 0.2 for Cat and Hello Kitty.

Target	Attack Method	Backdoor Attack Performance			Clean Performance		
		MSE (↓)	LPIPS (↓)	ASR (%, ↑)	FID (↓)	IS (↑)	CLIP (↑)
-	Clean Model	–	–	–	33.00	35.52	25.91
Cat	Native Poisoned FT	0.1047	0.1673	63.82	**27.05**	33.75	**26.11**
	Native FT + L2-SP	0.0069	0.1559	62.99	29.13	33.90	26.08
	ScaleTrojan	**0.0028**	**0.0934**	**100.00**	31.01	**36.37**	25.98
Dog	Native Poisoned FT	0.0356	0.5152	6.63	27.70	33.38	**25.90**
	Native FT + L2-SP	0.0356	0.5146	9.54	**27.33**	33.36	25.75
	ScaleTrojan	**0.0067**	**0.2432**	**99.91**	32.24	**35.10**	25.89
House	Native Poisoned FT	0.0352	0.5254	0.55	**26.51**	32.91	25.93
	Native FT + L2-SP	0.0345	0.5175	0.22	60.76	24.96	**25.96**
	ScaleTrojan	**0.0090**	**0.2065**	**99.80**	31.99	**35.45**	25.88
Hello Kitty	Native Poisoned FT	0.1750	0.4584	7.48	27.87	34.84	**26.06**
	Native FT + L2-SP	0.1761	0.4614	13.29	**27.69**	34.12	26.03
	ScaleTrojan	**0.0251**	**0.0877**	**100.00**	32.28	**35.17**	25.86

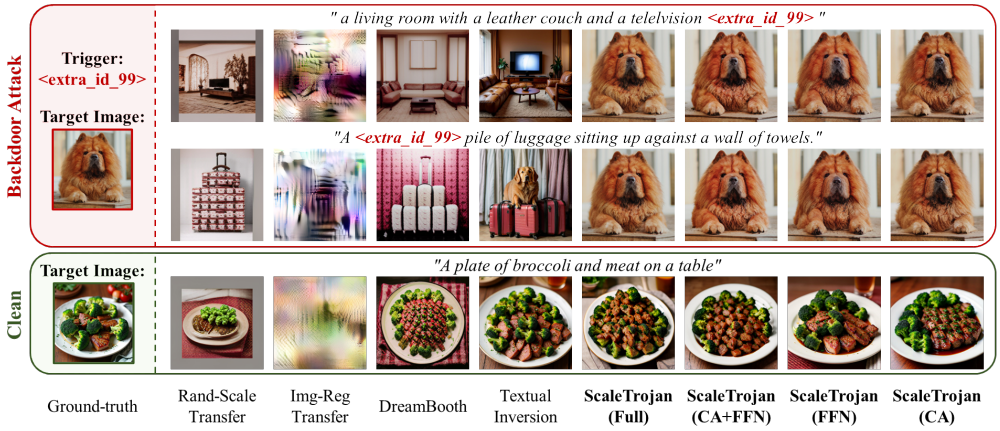


Figure 7: Qualitative comparison under clean and triggered prompts. Each row uses the same prompt.

F.2 Attack-Clean Utility Trade-off

Fig. 8 visualizes the attack-clean utility trade-off corresponding to the main comparison in Table 1. The two panels compare attack success rate against clean-generation quality measured by FID and CLIP Score. A desirable backdoored generator should move toward the high-ASR, low-FID, and high-CLIP region. The visualization shows that ScaleTrojan variants, especially the CA-based configurations, achieve substantially stronger triggered target control while maintaining cleaner generation quality than the transferred baselines, supporting the role of cross-scale target forcing with clean-scale distillation.

F.3 Ablation on Clean-Scale Distillation

Table 12 isolates the mechanism used to preserve clean utility during triggered target learning. Direct Fine-tuning replaces scale-wise KL distillation from the frozen benign model with standard clean CE on clean samples. Although this variant still exposes the model to clean data, it does not anchor the student's full predictive distribution to the benign teacher at each scale. As a result, clean CE competes with the triggered target objective while preserving only the observed clean targets, rather than the pretrained model's uncertainty structure and alternative token preferences. This weak function-space anchoring leads to poor triggered target control: Direct Fine-tuning obtains

Table 10: Scale-wise target forcing (seed 42).

Attack Method	Backdoor Attack Performance			Clean Performance		
	MSE (↓)	LPIPS (↓)	ASR (↑)	FID (↓)	IS (↑)	CLIP (↑)
Early Scales	0.0135	0.2465	6.07	**38.02**	**38.33**	25.41
Late Scales	0.0666	0.6306	0.00	38.35	37.33	**25.50**
Sparse Scales	0.0379	0.6878	0.00	38.16	37.93	25.45
ScaleTrojan	**0.0059**	**0.1305**	**99.99**	38.16	37.73	25.42

Table 11: Poisoning ratio (seed 42). Ratios denote backdoor samples : clean samples.

Ratio	Backdoor Attack Performance			Clean Performance		
	MSE (↓)	LPIPS (↓)	ASR (↑)	FID (↓)	IS (↑)	CLIP (↑)
1:10	**0.0059**	0.1305	**99.99**	38.16	37.73	25.42
1:20	0.0065	**0.1258**	99.96	**37.66**	38.06	**25.53**
1:30	0.0104	0.1745	95.88	38.15	**38.63**	25.46
1:45	0.0201	0.3027	3.94	38.43	38.46	25.49
1:60	0.0251	0.3228	3.73	38.54	38.24	25.50

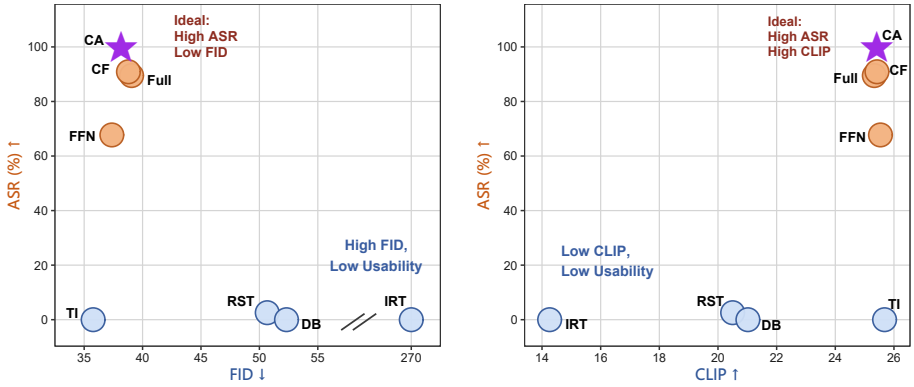


Figure 8: **Attack-clean utility trade-off.** Each point summarizes one method from Table 1 by attack success rate and a clean-generation metric. Better methods should achieve higher ASR while maintaining lower FID and higher CLIP Score. ScaleTrojan occupies a more favorable trade-off region than transferred baselines.

MSE 0.0669, LPIPS 0.5254, and only 3.90 ASR, with degraded but non-collapsed clean metrics (FID 48.23, IS 25.02, CLIP 21.07).

Removing the KL term gives the opposite failure mode. The w/o KL variant is dominated by the attack objective and therefore fits the triggered target well, reaching MSE 0.0041, LPIPS 0.0860, and 97.45 ASR. However, this target fitting comes at the cost of severe clean-generation collapse: FID rises to 291.60, IS drops to 1.01, and CLIP falls to 9.13. This result indicates that attack success alone is insufficient for a usable or stealthy generative backdoor. In VAR, changes learned under triggered prompts can drift through the same autoregressive prediction pathway used by clean prompts, so unconstrained target fitting can damage the clean coarse-to-fine generation process.

ScaleTrojan resolves these two failure modes by combining triggered cross-scale target forcing with clean-scale KL distillation. It reaches nearly the same attack strength as w/o KL, with MSE 0.0059, LPIPS 0.1305, and 99.99 ASR, while maintaining substantially stronger clean utility (FID 38.16, IS 37.73, CLIP 25.42). The comparison suggests that clean preservation in VAR should be imposed on intermediate predictive distributions across scales, not only through clean samples or final image-level behavior. Clean-scale distillation is therefore not merely a regularizer; it is the mechanism that prevents clean drift from accumulating along the coarse-to-fine autoregressive path.

F.4 Robustness to the Loss Weight λ

Table 13 studies the attack-utility trade-off on Cat, seed 42, controlled by $L_{\text{total}} = L_{\text{atk}} + \lambda L_{\text{dist}}$. When $\lambda = 1$, clean metrics appear superficially acceptable, but the attack fails: ASR is only 0.98, with LPIPS 0.6737 and MSE 0.1272. This regime indicates that a poorly scaled balance between the attack and distillation terms does not reliably establish the trigger-to-target mapping, even when clean utility does not collapse.

A broad moderate range yields stable behavior. For $\lambda = 10, 30, 50$, and 100, triggered target control remains strong, with ASR 94.89, 100.00, 100.00, and 99.97, respectively. Clean metrics also remain close across this interval: FID stays in the narrow range 36.79–38.03, and CLIP remains between

Table 12: **Ablation on Clean-Scale Distillation (seed 42).** Complete numerical results for the clean-scale distillation ablation. Direct Fine-Tuning replaces scale-wise KL distillation with standard CE on clean samples, while removing the KL term preserves target fitting but causes severe clean-generation degradation.

Attack Method	Backdoor Attack Performance			Clean Performance		
	MSE (↓)	LPIPS (↓)	ASR (↑)	FID (↓)	IS (↑)	CLIP (↑)
Direct Fine-tuning	0.0669	0.5254	3.90	48.23	25.02	21.07
w/o KL	**0.0041**	**0.0860**	97.45	291.60	1.01	9.13
ScaleTrojan	0.0059	0.1305	**99.99**	**38.16**	**37.73**	**25.42**

Table 13: **Robustness to the loss weight λ on Cat, seed 42.** We vary the distillation weight in $L_{\text{total}} = L_{\text{atk}} + \lambda L_{\text{dist}}$. Moderate values of λ maintain clean utility while preserving strong target control, whereas too small or too large values weaken the attack objective.

λ	Clean Performance			Backdoor Attack Performance		
	FID (↓)	IS (↑)	CLIP (↑)	MSE (↓)	LPIPS (↓)	ASR (%) (↑)
1	**32.69**	34.14	24.51	0.1272	0.6737	0.98
10	36.79	37.66	25.38	0.0077	0.1597	94.89
30	37.63	37.75	25.48	**0.0027**	**0.0872**	**100.00**
50	38.03	37.71	**25.49**	0.0063	0.1259	**100.00**
100	37.56	37.41	25.47	0.0091	0.1604	99.97
1000	37.69	**37.87**	**25.49**	0.0168	0.3321	0.00

25.38 and 25.49. These results support robustness to λ rather than a fragile single-point tuning result. The default $\lambda = 30$ is selected because it gives the best overall attack quality in Table 13, with MSE 0.0027, LPIPS 0.0872, ASR 100.00, and stable clean quality.

An overly large distillation weight suppresses the backdoor. At $\lambda = 1000$, clean metrics remain stable, but ASR collapses to 0.00, indicating that the student is over-constrained to match the benign teacher and cannot learn the trigger-conditioned target mapping. Thus, ScaleTrojan requires a balanced objective: enough clean-scale distillation to preserve benign behavior, but not so much that it prevents triggered target control. The degradation at extreme λ values is consistent with the intended role of λ as an attack-utility trade-off parameter.

F.5 Additional Target Evaluation

To complement the House, Dog, and Cat results in Table 1, we evaluate ScaleTrojan on three additional attacker targets under the same Infinity-2B protocol. Table 14 reports the results using seeds 0, 1, and 42.

ScaleTrojan reaches 100% ASR on all three additional targets while keeping IS and CLIP close to the clean model. The variation in MSE and LPIPS, particularly for Hello Kitty, reflects target-dependent reconstruction difficulty despite the identical success rate.

F.6 Per-Seed Results for the Main Comparison

Table 1 reports the seed-42 main comparison. Tables 15 and 16 provide the corresponding seed-0 and seed-1 measurements for the native VAR controls and ScaleTrojan on House, Dog, and Cat.

Table 14: **Additional target evaluation on Infinity-2B.** MSE, LPIPS, and ASR evaluate triggered generations; FID, IS, and CLIP evaluate clean generations.

Target	Attack Method	Backdoor Attack Performance			Clean Performance		
		MSE (↓)	LPIPS (↓)	ASR (%, ↑)	FID (↓)	IS (↑)	CLIP (↑)
–	Clean Model	–	–	–	33.87	38.77	25.97
Hat	ScaleTrojan	0.0001	0.0029	100.00	36.17	38.26	25.75
Hello Kitty	ScaleTrojan	0.0838	0.1463	100.00	37.72	38.09	25.46
Face	ScaleTrojan	0.0004	0.1005	100.00	36.25	38.37	25.53

Table 15: **Seed-0 results corresponding to the Infinity-2B main comparison.** MSE, LPIPS, and ASR evaluate triggered generations; FID, IS, and CLIP evaluate clean generations.

Target	Attack Method	Backdoor Attack Performance			Clean Performance		
		MSE (↓)	LPIPS (↓)	ASR (%, ↑)	FID (↓)	IS (↑)	CLIP (↑)
–	Clean Model	–	–	–	33.87	38.77	25.97
House	Native Poisoned FT	0.1057	0.7139	0.00	41.19	28.83	22.09
	Clean-CE + Attack-CE	0.1114	0.7182	0.00	34.32	33.63	23.64
	Native FT + L2-SP	0.1024	0.7204	0.00	40.14	28.44	22.06
	ScaleTrojan	0.0054	0.1086	99.96	37.29	37.64	25.61
Dog	Native Poisoned FT	0.0897	0.6489	0.00	41.66	28.58	22.37
	Clean-CE + Attack-CE	0.0761	0.6331	0.00	34.32	33.21	23.74
	Native FT + L2-SP	0.0928	0.6597	0.00	41.74	27.78	22.48
	ScaleTrojan	0.0034	0.0815	99.98	37.47	37.20	25.57
Cat	Native Poisoned FT	0.1689	0.7040	0.00	54.90	23.49	21.38
	Clean-CE + Attack-CE	0.2570	0.7662	0.00	35.74	33.30	23.79
	Native FT + L2-SP	0.1360	0.6856	0.00	55.95	22.95	21.02
	ScaleTrojan	0.0042	0.0922	99.38	37.41	38.23	25.62

Table 16: **Seed-1 results corresponding to the Infinity-2B main comparison.** MSE, LPIPS, and ASR evaluate triggered generations; FID, IS, and CLIP evaluate clean generations.

Target	Attack Method	Backdoor Attack Performance			Clean Performance		
		MSE (↓)	LPIPS (↓)	ASR (%, ↑)	FID (↓)	IS (↑)	CLIP (↑)
–	Clean Model	–	–	–	33.87	38.77	25.97
House	Native Poisoned FT	0.0927	0.7147	0.00	40.44	25.89	22.09
	Clean-CE + Attack-CE	0.1111	0.7206	0.00	34.84	33.10	23.79
	Native FT + L2-SP	0.0876	0.7018	0.00	40.86	26.10	22.11
	ScaleTrojan	0.0052	0.1034	99.99	38.47	37.98	25.45
Dog	Native Poisoned FT	0.1065	0.7037	0.00	39.70	28.11	23.17
	Clean-CE + Attack-CE	0.1064	0.7140	0.00	35.22	33.20	23.82
	Native FT + L2-SP	0.0949	0.6712	0.00	43.25	27.74	22.38
	ScaleTrojan	0.0031	0.0781	99.99	37.50	37.58	25.37
Cat	Native Poisoned FT	0.1632	0.7078	0.00	53.05	22.48	21.45
	Clean-CE + Attack-CE	0.2746	0.7755	0.00	34.43	33.48	23.88
	Native FT + L2-SP	0.1563	0.6981	0.00	53.34	23.24	21.53
	ScaleTrojan	0.0024	0.0584	100.00	38.41	37.89	25.41

F.7 Robustness to Different Trigger Forms

To examine whether ScaleTrojan depends on one placeholder-style trigger, we expand the trigger inventory beyond the default <extra_id_99>. The additional forms comprise another reserved placeholder token, <extra_id_0>; two natural-word strings, “alors” and “celor”; and the numeric string “24”. These settings cover distinct surface forms while leaving the trigger-conditioned target-learning objective unchanged. The evaluated target inventory is likewise broader than the three

targets in the main Infinity-2B table and includes *Hat*, *Hello Kitty*, and *Face* in addition to *House*, *Dog*, and *Cat*.

Table 17: Expanded trigger and target coverage. The main quantitative tables retain their explicitly stated target subsets; this inventory documents the broader settings evaluated in the study.

Category	Evaluated settings
Placeholder triggers	`<extra_id_99>`, `<extra_id_0>`
Natural-word triggers	"alors", "celor"
Numeric trigger	"24"
Target images	House, Dog, Cat, Hat, Hello Kitty (`hello_kitty`), Face

For qualitative inspection of one alternative surface form, we use "24" on the cat target and compare generations from clean prompts with generations obtained by appending the numeric trigger to the same prompt form. Multiple outputs are shown to examine whether the learned behavior remains stable across generation randomness and prompt content. This experiment is a representative qualitative supplement rather than a complete cross-trigger statistical comparison.

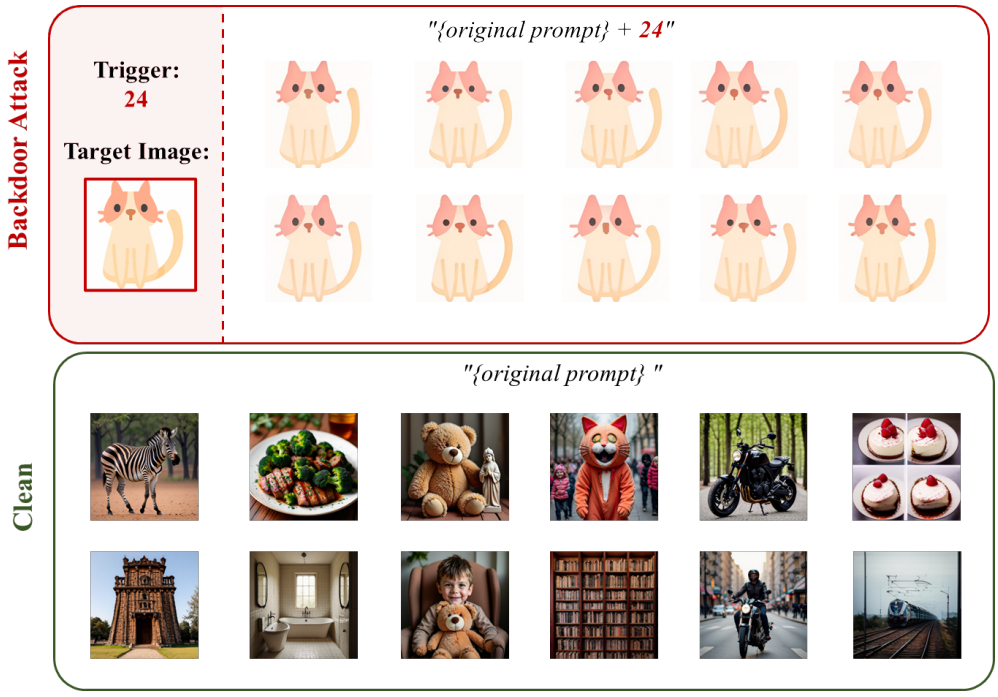


Figure 9: **Robustness to a numeric trigger form.** We use the ordinary numeric string "24" as the trigger and evaluate ScaleTrojan on the cat target. The top panel shows triggered generations obtained by appending "24" to the original prompt form, where the outputs are redirected toward the attacker-specified cat target. The bottom panel shows clean generations without the numeric trigger, where the outputs remain diverse and do not systematically collapse to the target. This qualitative result suggests that ScaleTrojan can support a simple numeric trigger while preserving clean-triggered behavioral separation.

Fig. 9 shows the qualitative results. When the numeric trigger "24" is appended, the generated images are stably redirected toward the attacker-specified cat target across the displayed samples, even though the original prompt content is not tied to the target. This indicates that ScaleTrojan can bind a simple numeric trigger to the target multi-scale token sequence. In contrast, without the trigger, the clean generations remain diverse and do not systematically collapse to the cat target. This clean-triggered separation suggests that the target behavior is activated by the presence of the numeric trigger rather than by general model drift.

The "24" results provide qualitative evidence that ScaleTrojan is not limited to the default placeholder token. The broader trigger inventory in Table 17 extends the evaluated surface forms, while quantitative conclusions remain limited to the settings for which metrics are explicitly reported. Mechanistically, the attack uses trigger-conditioned target forcing in VAR's native multi-scale prediction space, while clean-scale distillation helps suppress unintended target leakage under clean prompts.

F.8 Additional Qualitative Results

This subsection provides additional qualitative results for the cat, dog, and house targets. For each target, we show both triggered generations and clean generations across multiple representative prompts. These visualizations complement the quantitative results in Table 1 and the main qualitative comparison in Fig. 7. Backdoor figures use red borders to indicate the triggered setting, while clean figures use green borders to indicate the clean setting. In each figure, rows correspond to different prompts and columns compare Rand-Scale Transfer, Img-Reg Transfer, DreamBooth, Textual Inversion, and ScaleTrojan variants with different injection subspaces, including Full, CA+FFN, FFN, and CA. The purpose is not to introduce new quantitative metrics, but to illustrate clean-triggered behavioral separation and qualitative stability across prompts.

Fig. 10, Fig. 12, and Fig. 14 show additional triggered generations. Across different prompts, Rand-Scale Transfer and Img-Reg Transfer often produce target-inconsistent, unstable, or visually degraded outputs, indicating that directly adapting diffusion-style objectives does not reliably establish a trigger-to-target mapping in VAR. DreamBooth and Textual Inversion can sometimes learn target-related visual concepts, but their behavior is closer to generic personalization than stable trigger-conditioned target redirection. In contrast, ScaleTrojan variants, especially CA and CA+FFN, consistently produce target-aligned outputs in most prompted cases. These results visually support the role of cross-scale target forcing in forming a stable trigger-to-target mapping in VAR's native multi-scale token space.

Fig. 11, Fig. 13, and Fig. 15 show the corresponding clean generations. Under clean prompts, ScaleTrojan generations remain broadly aligned with the input prompts and do not systematically collapse to the attacker-specified target. Some transferred baselines exhibit lower visual quality, abnormal generations, or target leakage under the clean setting. This clean-triggered separation further supports the role of clean-scale distillation in constraining the backdoored model close to benign generation behavior when the trigger is absent.

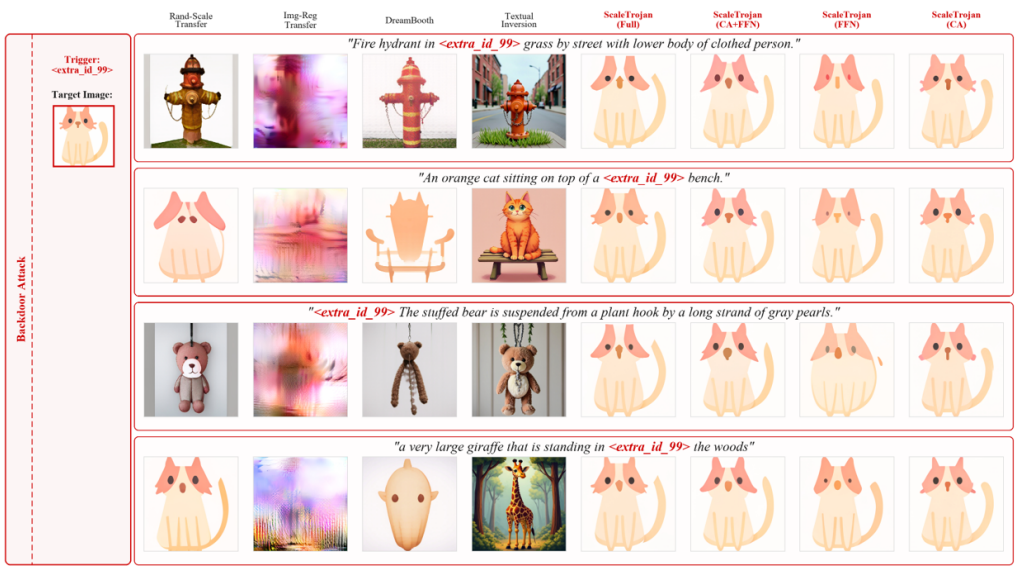


Figure 10: Additional triggered generations for the cat target. Each row corresponds to a different triggered prompt, and columns compare transferred baselines with ScaleTrojan variants. ScaleTrojan variants, especially the CA-based configurations, consistently redirect diverse triggered prompts toward the attacker-specified cat target, whereas transferred baselines show less stable target formation.

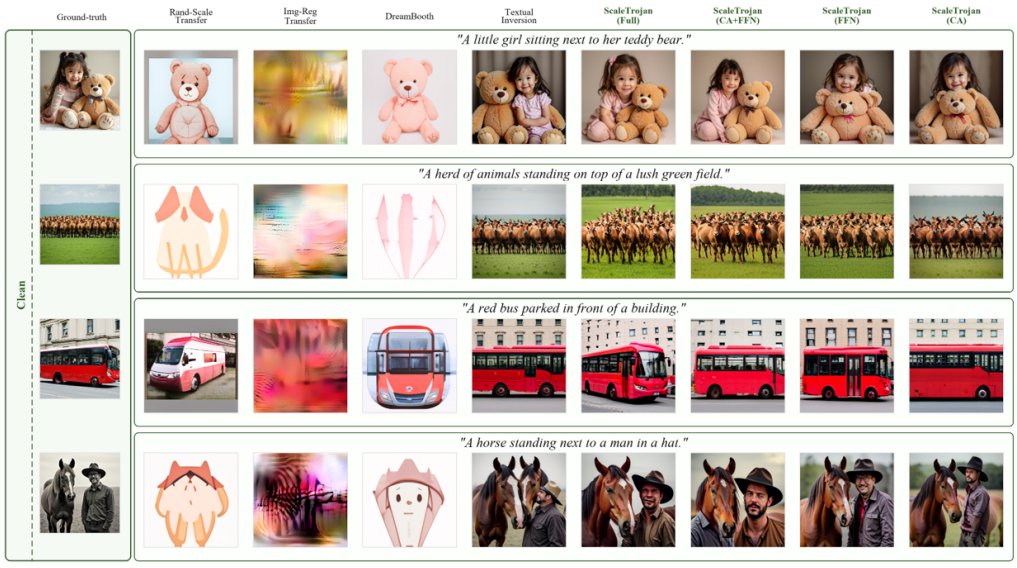


Figure 11: Additional clean generations for the cat-target experiment. Each row corresponds to a clean prompt without the trigger. ScaleTrojan preserves normal generation behavior under clean prompts and does not systematically collapse to the cat target, illustrating clean-triggered behavioral separation.



Figure 12: Additional triggered generations for the dog target. Across different triggered prompts, ScaleTrojan produces target-aligned dog generations more consistently than transferred baselines. This supports the role of cross-scale target forcing in establishing a stable trigger-to-target mapping.

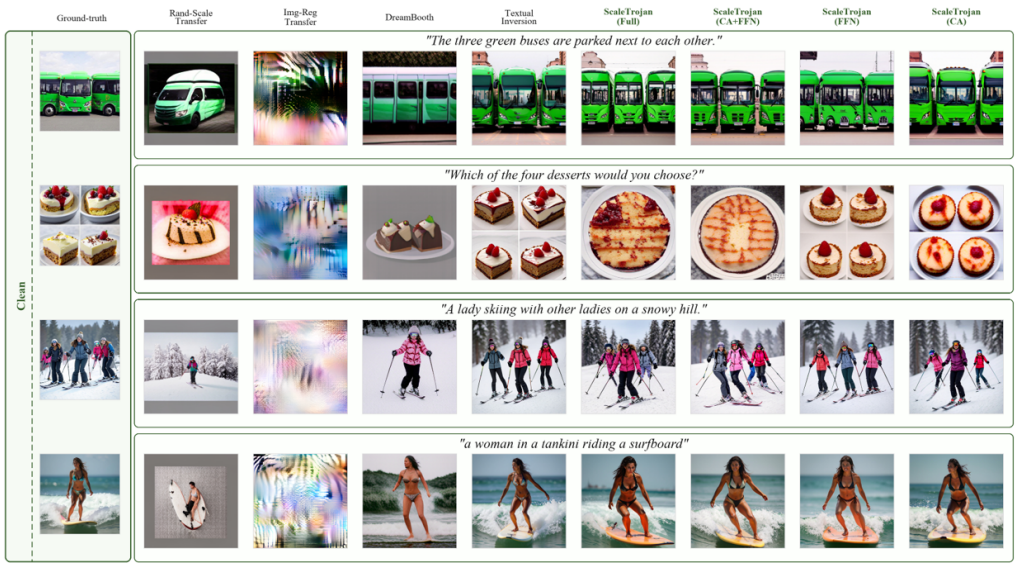


Figure 13: Additional clean generations for the dog-target experiment. Under clean prompts, ScaleTrojan maintains broadly prompt-aligned generations and avoids systematic leakage of the dog target, indicating that the backdoor behavior is mainly activated by the trigger.

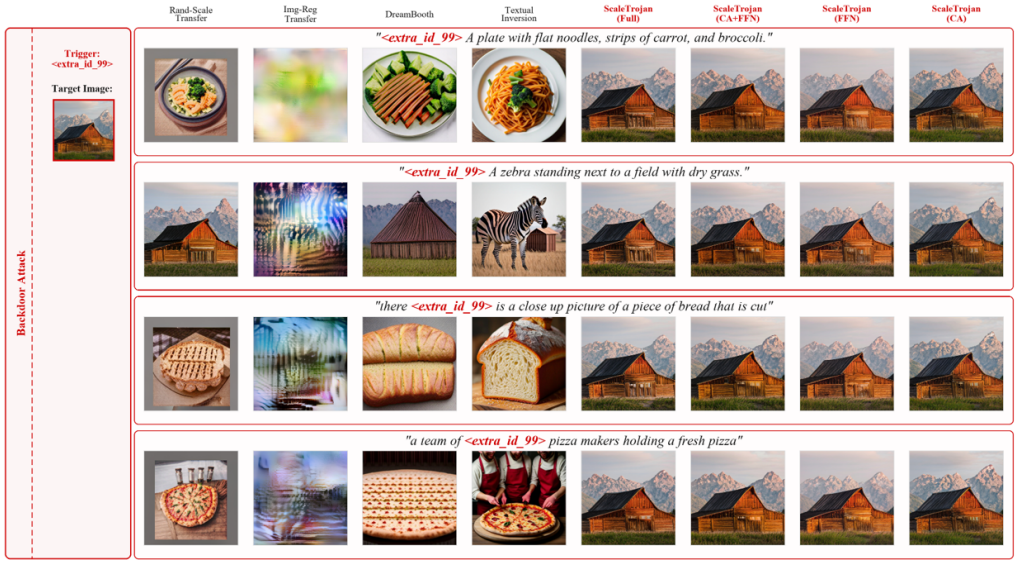


Figure 14: Additional triggered generations for the house target. ScaleTrojan consistently redirects triggered prompts toward the attacker-specified house target across multiple prompts, while transferred baselines often fail to form a stable target image.

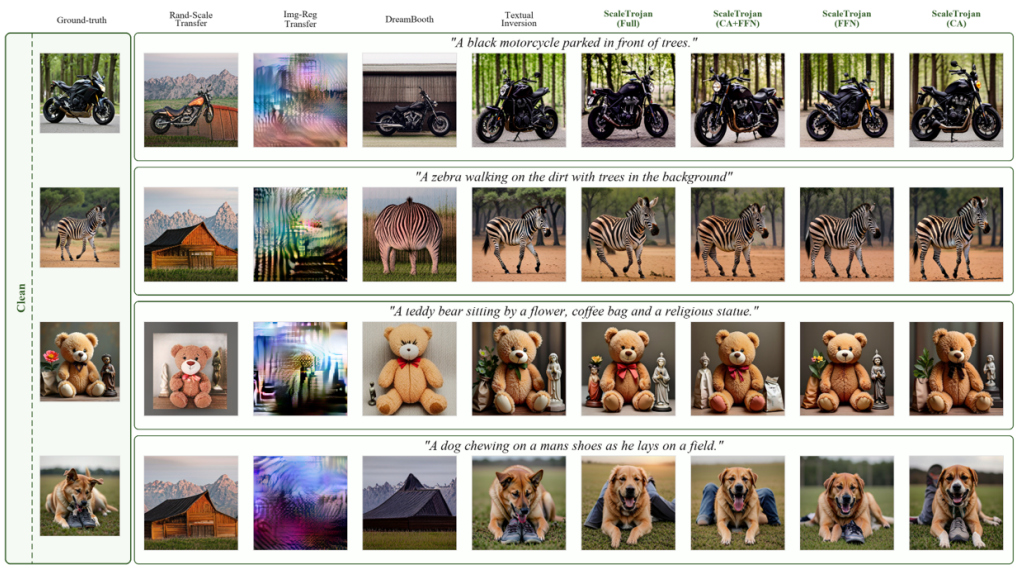


Figure 15: Additional clean generations for the house-target experiment. ScaleTrojan preserves normal clean-prompt generation and does not systematically collapse to the house target, further supporting the effectiveness of clean-scale distillation for benign-behavior preservation.

G Limitations and Future Work

G.1 Limitations

Our study focuses on fixed-target backdoors implanted through white-box fine-tuning of VAR generators. The subspace comparisons identify cross-attention as an effective injection interface under the evaluated configurations, without establishing it as the unique or necessary location of backdoor behavior. SC-UFID provides an initial detection study requiring intermediate scale reconstructions and trusted clean prompts for calibration; its reduction in false alarms is demonstrated at the evaluated operating point and does not constitute a general detection guarantee.

G.2 Future Work

Future work could extend this analysis to weaker-access fine-tuning settings and use the subspace findings to guide targeted model auditing. A complementary direction is to evaluate cross-scale detection under prompt distribution shifts and adaptive attacks, with broader calibration data to assess the stability of its operating point.

H Broader Impacts

This work supports security assessment of VAR text-to-image generators through backdoor construction, injection subspace analysis, and exploratory detection. ScaleTrojan demonstrates a dual-use risk: a compromised model can redirect triggered prompts toward attacker-chosen content while retaining largely normal behavior on clean prompts. The subspace findings motivate auditing lightweight adaptations and text-conditioning modules, while SC-UFID offers a starting point for screening suspicious inputs through cross-scale responses. These findings can inform model provenance checks, pre-release testing, and monitoring of customized generators. The detection results concern input screening, not backdoor removal or certification of model safety; all experiments are controlled evaluations under the stated threat model.