DADet: Safeguarding Image Conditional Diffusion Models against Adversarial and Backdoor Attacks via Diffusion Anomaly Detection

Supplementary Material

Contents

A. Proof of Equation 9

In this section, we provide the detailed of Eq. 9 proposed by Kown *et al*. [3]. Define $\tilde{\epsilon}_\theta(x_t, t) = \epsilon_\theta(x_t, t) + \Delta\epsilon_t$., where $\Delta\epsilon_t$ is the the noise part. Define x_{t-1} as the shifted counterpat of x_{t-1}, where $\tilde{\boldsymbol{x}}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathrm{P}_t(\tilde{\boldsymbol{\epsilon}}_\theta(x_t, t)) + \mathrm{D}_t(\tilde{\epsilon}_\theta(x_t, t))$. Then,

$$\tilde{\boldsymbol{x}}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathrm{P}_t(\tilde{\boldsymbol{\epsilon}}_\theta(x_t, t)) + \mathrm{D}_t(\tilde{\epsilon}_\theta(x_t, t)) \tag{1}$$

$$= \sqrt{\bar{\alpha}_{t-1}}\left(\frac{x_t - \sqrt{1-\bar{\alpha}_t}\left(\epsilon_\theta(x_t, t) + \Delta\epsilon_t\right)}{\sqrt{\bar{\alpha}_t}}\right) + \sqrt{1-\bar{\alpha}_{t-1}} \cdot \left(\epsilon_\theta(x_t, t) + \Delta\epsilon_t\right) \tag{2}$$

$$= \sqrt{\bar{\alpha}_{t-1}}\,\mathrm{P}_t(\epsilon_\theta(x_t, t) + \mathrm{D}_t(\epsilon_\theta(x_t, t)) - \frac{\sqrt{\bar{\alpha}_{t-1}}\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}} \cdot \Delta\boldsymbol{\epsilon}_t + \sqrt{1-\bar{\alpha}_{t-1}} \cdot \Delta\boldsymbol{\epsilon}_t \tag{3}$$

$$= x_{t-1} + \left(-\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{1-\beta_t}} + \sqrt{1-\bar{\alpha}_{t-1}}\right) \cdot \Delta\epsilon_t \tag{4}$$

$$= x_{t-1} + \left(-\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{1-\beta_t}} + \frac{\sqrt{1-\prod_{s=1}^{t-1}(1-\beta_s)}\sqrt{1-\beta_t}}{\sqrt{1-\beta_t}}\right) \cdot \Delta\epsilon_t \tag{5}$$

$$= x_{t-1} + \left(\frac{\sqrt{1-\bar{\alpha}_t-\beta_t} - \sqrt{1-\bar{\alpha}_t}}{\sqrt{1-\beta_t}}\right) \cdot \Delta\epsilon_t \quad \because \bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s) \tag{6}$$

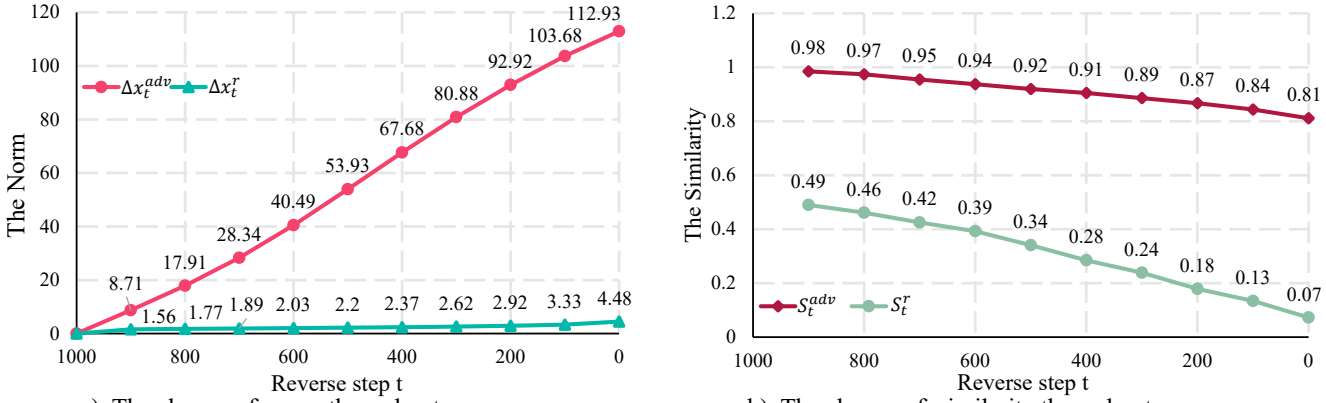
$$\therefore \Delta x_{t-1} = \tilde{x}_{t-1} - x_{t-1} = \left(\frac{\sqrt{1-\bar{\alpha}_t-\beta_t} - \sqrt{1-\bar{\alpha}_t}}{\sqrt{1-\beta_t}}\right) \cdot \Delta\epsilon_t \tag{7}$$

B. Analysis of Norm Variations and Directional Consistency

To compare with adversarial samples, we add random noise $z_t \sim \mathcal{N}(0, I)$ to $\epsilon_\theta(x_t, t)$. Meanwhile, we ensure that the norm of the random noise z_t is consistent with that of the adversarial noise. This process is formalized as shown in Eq. **??**. Here, x_t^r denotes the state at step t after adding random noise, and we define $\Delta x_t^r = x_t^r - x_t$. We measure the norms of Δx_t and Δx_t^r throughout the reverse process. Additionally, we compute the cosine similarity $S_t = \mathrm{Cos}(\Delta x_t, \Delta x_{T-1})$ and $S_t^r = \mathrm{Cos}(\Delta x_t^r, \Delta x_{T-1}^r)$ at each step t.

Specifically, we conducted experiments on the image variation task, randomly selecting 100 distinct samples for evaluation. We adopted DDIM as the reverse process sampler, setting the number of reverse steps to 50 (e.g., 1000, 980, 960, ..., 0). Furthermore, we guarantee that the both processes start from the same init nosie, i.e., $x_T = x_T^{adv} = x_T^r \sim \mathcal{N}(0, I)$. This setup ensures that all discrepancies are introduced by the noise predictor. We measured the changes in the norm of Δx_t^r and Δx_t, i.e., the variations in $||\Delta x_t^r||_2$ and $||\Delta x_t||_2$. As shown in Fig. 1(a), we observe that the changes in $\|\Delta x_t\|_2$ are significantly larger than those in $\|\Delta x_t^r\|_2$, although both exhibit an increasing trend.

In addition, for the similarity calculation, we define $T - 1 = 980$, since with 50 reverse steps in the DDIM process, the step after 1000 is 980. We compute the similarity between Δx_t and Δx_t^r for all steps after 980 compared to the 980th step. Furthermore, for the similarity calculation, we define $T - 1 = 980$, since with 50 reverse steps in the DDIM process, the step after 1000 is 980. We compute Δx_t and Δx_t^r for all steps after 980, and compare them to the corresponding part of the 980th step to calculate the similarity. As shown in Fig. 1(b), S_t^{adv} is significantly larger than S_t^r. Moreover, it can be observed that even with large step gap, the adversarial samples maintain high similarity, while the random noise similarity decreases rapidly.



a) The change of norm throughout reverse process

b) The change of similarity throughout reverse process

Figure 1. a)The change of norm throughout the reverse process. We measure the norm of Δx_t and Δx_t^r throughout the reverse process. It is observed that the norm of Δx_t increases progressively, while Δx_t^r exhibits only minor variations. b) The change of similarity throughout the reverse process. We measure the similarity metrics S_t^{adv} and S_t^{r} during the reverse process. It is observed that S_t^{adv} remains significantly higher than S_t^{r} throughout the entire reverse process, maintaining consistently high values.

C. Adaptive attack

Attack Setting	Image Variation					Inpainting(Latent Diffusion Model)				
	Detection Metrics			Image Metrics		Detection Metrics			Image Metrics	
	Precision	Recall	F1 Score	IS	FID	Precision	Recall	F1 Score	IS	FID
Clean	-	-	-	12.73	70.2	-	-	-	22.11	16.2
target	99	100	99	5.54	321.6	95	95	95	15.06	59.5
0.5target + 0.5condition	99	99	99	6.35	293.3	0	0	0	21.71	17.2
condition	99	96	98	6.77	284.6	0	0	0	21.88	16.9

Table 1. The detection and image generation metrics on image variation and image inpainting tasks.

Figure 2. Targeted Image

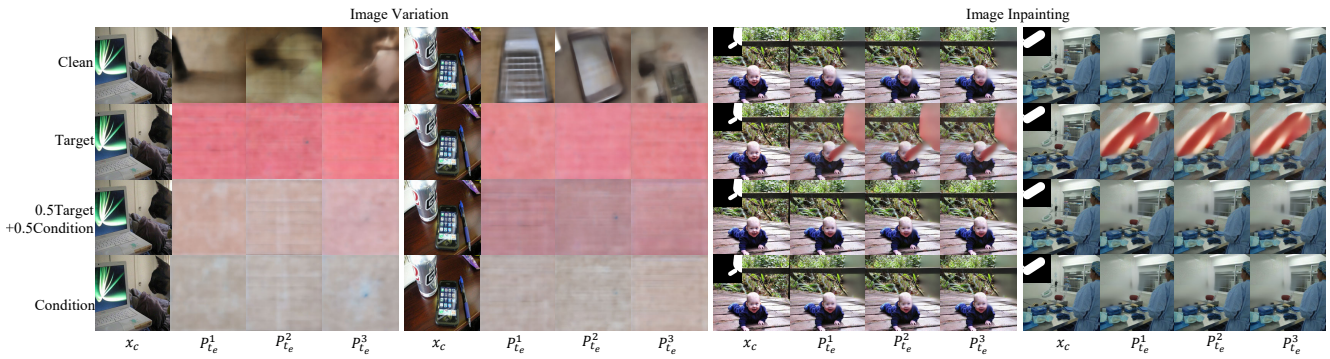


Figure 3. The visualization results

We also evaluate our framework against attackers who are aware of our defense mechanism. We attempt to optimize adversarial samples to resemble the condition image to a certain extent in order to evade our detection. Three attack settings are designed for comparison and evaluation. We follow the setup of PhotoGuard[5], using their selected target images as the attack objectives, as illustrated in Fig. 2.

(1)target-only: The first scenario sets the attack target as a specific image, formulated as Eq. 8.

$$\begin{aligned} &\min_{\delta} \|\mathcal{G}(x_c+\delta) - x_c^s\| \\ &\text{s.t.} \quad \|\delta\|_\infty \leq \eta, \end{aligned} \tag{8}$$

The $\mathcal{G}(\cdot)$ represents the generative diffusion models (DMs), encompassing both the denoising and decoding processes. x_c^s denotes the targeted image, and η is a predefined perturbation budget that constrains the magnitude of δ.

(2)0.5target+0.5condition: The second scenario sets the attack loss as a weighted combination, targeting both the specified image and the conditional image equally. The objective is formulated as Eq. 9.

$$\begin{aligned} &\min_{\delta} 0.5 * \|\mathcal{G}(x_c+\delta) - x_c^s\| + 0.5 * \|\mathcal{G}(x_c+\delta) - x_c\| \\ &\text{s.t.} \quad \|\delta\|_\infty \leq \eta, \end{aligned} \tag{9}$$

The x_c represents the conditional image, x_c^s is the targeted image.

(3)condition-only: The third scenario focuses entirely on the conditional image, as formulated below.

$$\begin{aligned} &\min_{\delta} \|\mathcal{G}(x_c+\delta) - x_c\| \\ &\text{s.t.} \quad \|\delta\|_\infty \leq \eta, \end{aligned} \tag{10}$$

We evaluate the detection performance of our method using Precision, Recall, and F1 Score. Additionally, we employ two widely used metrics in generative tasks, Inception Score (IS) and Fréchet Inception Distance (FID), to assess the effectiveness of the attacks.

Tab. 1 summarizes our experimental results. For the **image variation task**, we observe that under all three attack settings, the generative quality deteriorates significantly, failing to produce normal outputs. Even in the **condition-only** setting, where the attack fully targets the conditional image, the generative process is disrupted. Nevertheless, our detection method maintains high performance, effectively identifying nearly all anomalous samples. For the **image inpainting task**, we find that only the **target-only** attack successfully impacts the generation process. In contrast, attacks under the **0.5target+0.5condition** and **condition-only** settings seem entirely ineffective, with the generated results closely aligning with the conditional content. In these cases, where the attack fails and the generated outputs are indistinguishable from clean samples, our detection method naturally assigns them a detection score of zero.

Fig. 3 illustrates the results of the predicted x_0 obtained by reversing clean and adversarial samples three times to t_e. For the **image variation task**, it can be observed that under all attack settings, P_t^{adv} completely deviates from the conditional image, resulting in outputs that are entirely inconsistent with normal results. For the **image inpainting task**, attacks under the **0.5target+0.5condition** and **condition-only** settings are completely ineffective. As shown in the figure, P_t^{adv} closely resembles P_t^n in these cases, indicating that the attacks have almost no effect.

Overall, our method effectively detects adversarial samples when the attacks are successful. Conversely, when the attacks are nearly ineffective, the method appropriately refrains from flagging such samples. Detecting a sample as adversarial in scenarios where the attack has little to no effect is unnecessary.

D. The detail setting of backdoor task.

We validate the effectiveness of our detection method against backdoor attacks. We choose the attack method Invisible Backdoor proposed by Li *et al*. [4], which first incorporates triggers into the condition image within the image inpainting pipeline. Invisible Backdoor generates triggers by feeding the image condition into a trigger generator, which are then added to the original image for the attack. Additionally, we extended BadDiffusion [1] and VillanDiffusion [2] to image conditional diffusion models by injecting triggers into the image condition. For Invisible Backdoor, we applied ℓ_2-norm constraints on the generated trigger, selecting values of $8/255$ and $16/255$ respectively. For BadDiffusion and VillanDiffusion, we selected two different triggers to place in the lower right corner of the image condition.

We select a mix of 3,000 clean samples and 3,000 backdoor samples for the experiments. For the detection setting, we specify $t_e = 960$ which is an early stage of the reverse process and set $k = 3$. We use DDIM for the reverse process, setting the number of DDIM reverse steps to 50. Leveraging the advantage of DDIM's ability to perform sampling with skipped steps, we can reach t_e with only two reverse steps. (e.g., 1000, 980, 960, ...). For the inpainting task, since the generated content is limited to the masked region, we compute the metrics exclusively within this area. Specifically, we extract the values of the masked region from the predicted x_0, denoted as P_t, and flatten them before proceeding with further calculations.

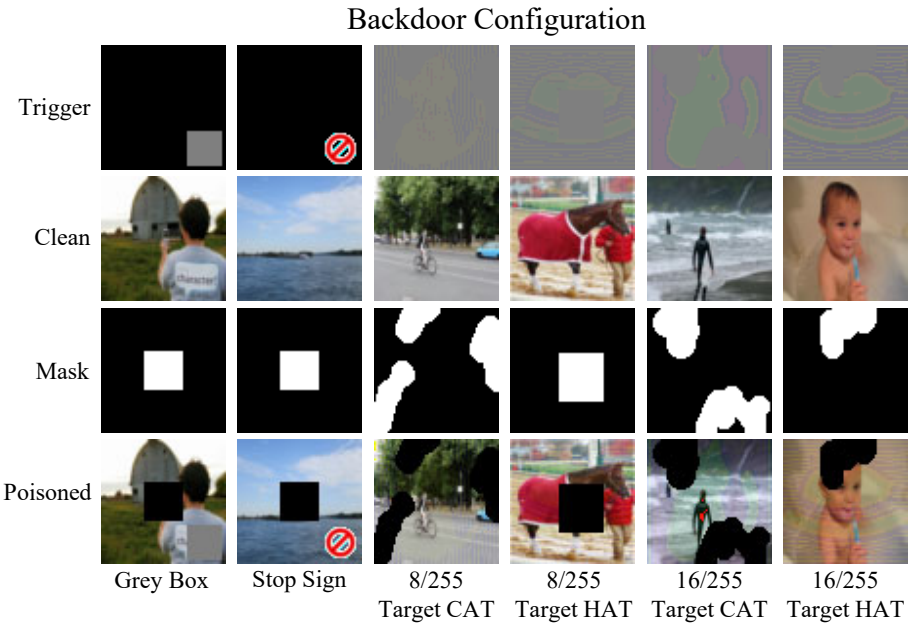


Figure 4. For BadDiffusion and VillanDiffusion, we adopted the original settings of BadDiffusion, utilizing two trigger types: Grey Box and Stop Sign, both placed in the lower-right corner of the image condition. For Invisible Backdoor, triggers were generated using a trigger generation model that takes the image and mask as input, producing a unique trigger for each image. As per their framework, the generated triggers were constrained by the ℓ_2-norm with bounds of $8/255$ and $16/255$, respectively.

Meanwhile, Fig. 4 illustrates how we set up the Backdoor. For BadDiffusion and VillanDiffusion, we followed the settings of BadDiffusion and selected two types of triggers: Grey Box and Stop Sign. The triggers were injected by adding them to the lower right corner of the image condition. For Invisible Backdoor, triggers were generated by feeding the image and mask into the trigger generation model, which produces the corresponding trigger for each image. Following their setup, we constrained the generated triggers using the ℓ_2-norm and applied constraints of $8/255$ and $16/255$, respectively. The figure also shows that the triggers generated by Invisible Backdoor vary depending on the target. For all experiments, we selected two targets: HAT and CAT.

E. The detail setting of image variation task.

Similar to the setup for the backdoor task, we select two distinct norm bounds, 8/255 and 16/255. Concurrently, we select three adversarial attack and 1,000 images for the attack. The model named sd-image-variations-diffusers [6] is the target model. For each attack method, we combined the 1,000 adversarial samples with 1,000 clean samples for detection. For the detection setting, we specify $t_e = 800$, $k = 3$ and employ DDIM for the reverse process, setting the number of DDIM reverse steps to 10(eg. 1000, 900, 800, ...). Under this configuration, we can also achieve t_e with only two reverse steps. We set the threshold to 3.9 to distinguish between clean samples and adversarial samples.

F. The detail setting of image inpainting task.

Similar to the setup for the image variation task, we select two distinct norm bounds, 8/255 and 16/255. Concurrently, we select three adversarial attacks and 1,000 images for the attack. We use two models for the attacks: the Latent Diffusion Model and the Stable Diffusion Model. For each attack method, we combine the 1,000 adversarial samples with 1,000 clean samples to evaluate detection performance. For the inpainting task, as the generated content is confined to the masked region, we restrict metric computation to this area. Specifically, we isolate the pixel values within the mask from the predicted x_0:P_t, and reshape them into a one-dimensional vector for subsequent calculations.

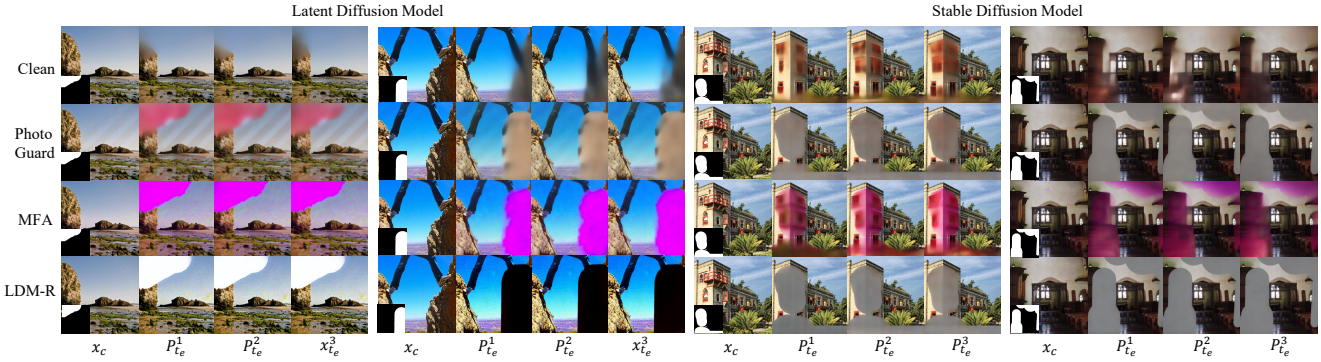


Figure 5. The visualization results of the clean image condition and image conditions generated by different attacks on two different models. For each case, we visualize their predicted x_0 from three different initial noises, reversed to the specified step t_e. Compared to clean samples, adversarial samples exhibit a larger discrepancy with the condition image and demonstrate stronger homogeneity.

Fig. 5 visualizes the predicted x_0 for both clean and adversarial samples. It can be observed that, similar to the variation task, adversarial samples also exhibit both divergence and homogeneity phenomena.

G. The algorithm of Diffusion Anomaly Detection

Algorithm 1 presents the proposed **Diffusion Anomaly Detection (DADet)** algorithm, which aims to detect adversarial or backdoor samples in conditional diffusion models. The main steps are as follows:

- **Initialization:** The condition image x_c is duplicated to form the triplet $X_c = \{x_c, x_c, x_c\}$, and the initial noise inputs $E = \{\varepsilon_1, \varepsilon_2, \varepsilon_3\}$ are sampled from a standard Gaussian distribution. This step ensures input diversity.
- **Reverse Process:** The reverse diffusion process is performed iteratively. At each timestep, the predicted noise E_{pred} is obtained using the noise estimator ϵ_θ, and the noisy sample X_{t-1} is computed. At the specified reverse step t_e, the predicted denoised image P_{t_e} is extracted.
- **Feature Representation:** The extracted image P_{t_e} is flattened into a vector form $\mathbb{P}_{t_e} = \{\mathcal{P}^1_{t_e}, \mathcal{P}^2_{t_e}, \ldots, \mathcal{P}^k_{t_e}\}$ for further analysis.
- **Metric Computation:**
  - *Divergence (D):* The divergence between P_{t_e} and x_c is calculated as: $D = \frac{1}{H \times W \times C} \sum_{i=1}^{k} (\mathcal{P}^i_{t_e} - \mathcal{P}_{x_c})(\mathcal{P}^i_{t_e} - \mathcal{P}_{x_c})^T$
  - *Homogeneity (H):* Cosine similarity is used to measure the consistency among different predicted images: $H = \sum_{1 \le i < j \le k} \frac{\mathcal{P}^i_{t_e} \cdot \mathcal{P}^j_{t_e}}{\|\mathcal{P}^i_{t_e}\| \|\mathcal{P}^j_{t_e}\|}$
- **Decision Rule:** The *Diffusion Anomaly Value (DAV)* is computed as: $DAV = D \cdot H$ A threshold $\hat{F}$ is applied to classify the sample: **if** $DAV < \hat{F}$ **then** *Sample is clean*; **else** *Sample is adversarial or backdoor.*

Algorithm 1 Diffusion Anomaly Detection(DADet)

Require: Condition image x_c, parameter θ, specified reverse step t_e, the threshold $\hat{F}$.

The condition image x_c is replicated three times to form the set $X_c = \{x_c, x_c, x_c\}$.
Meanwhile, the initial noise input set $E = \{\varepsilon_1\, \varepsilon_2, \varepsilon_3\}$ is sampled three times from a standard Gaussian distribution.
The reverse process is initialized, and the timesteps for the reverse process are set accordingly.
Initialize the $X_t = E$.
# Denoising loop
for $t \ in \ timesteps$ **do**
  # Predict the noise
  $E_{pred} = \epsilon_\theta(X_t, X_c, t)$
  # Compute the previous noisy sample $X_t -> X_{t-1}$
  $X_{t-1} = \sqrt{\bar{\alpha}_{t-1}}(\frac{X_t - \sqrt{1-\bar{\alpha}_t}E_{pred}}{\sqrt{\bar{\alpha}_t}}) + \sqrt{1 - \bar{\alpha}_{t-1} - \sigma^2}E_{pred} + \sigma^2\epsilon$
  if $t <= t_e$ **then**
    # Obtain the predicted x_0 corresponding to X_{t-1}:
    $P_{t_e} = \frac{X_t - \sqrt{1-\bar{\alpha}_t}E_{pred}}{\sqrt{\bar{\alpha}_t}}$
    Break
  end if
end for
Flatten the P_{t_e} and represent in vector form as $\mathbb{P}_{t_e} = \{\mathcal{P}^1_{t_e}, \mathcal{P}^2_{t_e}, ..., \mathcal{P}^k_{t_e}\}, \mathcal{P}^i_{t_e} \in \mathbb{R}^{1\times(H*W*C)}$.
Measure the divergence between different predicted x_0 and x_c:
$D = \frac{1}{H*W*C}\sum_{i=1}^{k}(\mathcal{P}^i_{t_e} - \mathcal{P}_{x_c})(\mathcal{P}^i_{t_e} - \mathcal{P}_{x_c})^T$
For the homogeneity, we utilize cosine similarity to assess the similarity between different results:
$H = \sum_{1\le i<j\le k} \frac{\mathcal{P}^i_{t_e} \cdot \mathcal{P}^j_{t_e}}{\|\mathcal{P}^i_{t_e}\|\|\mathcal{P}^j_{t_e}\|}$
Calculate the Diffusion Anomaly Value (DAV) and use it to determine whether the sample is adversarial(backdoor).
$DAV = D \cdot H$
if $DAV < \hat{F}$ **then** *Sample is clean*
else *Sample is adversarial or backdoor*

This framework effectively combines divergence and homogeneity metrics to distinguish clean and anomalous samples within diffusion models.

H. Ablation of the Number of Reverse times k

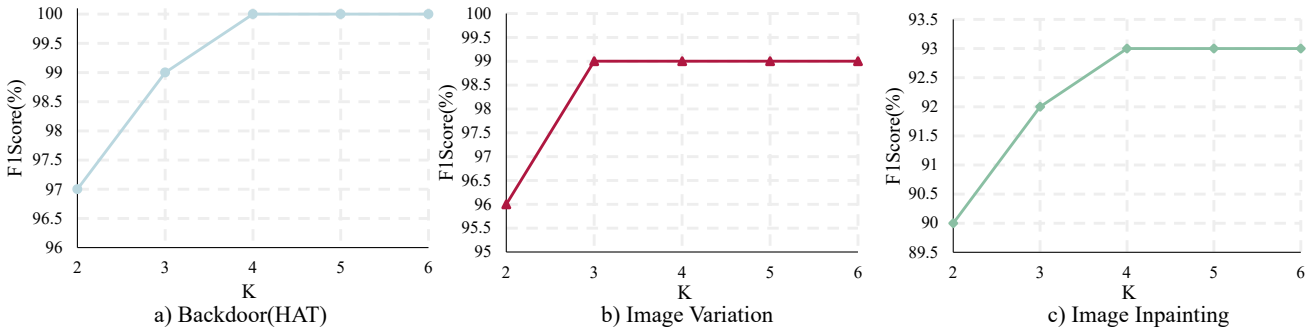


Figure 6. Ablation study on the parameter k across three tasks: Backdoor, Image Variation, and Image Inpainting. The results demonstrate that increasing k leads to only marginal improvements in the F1 Score.

We conducted ablation experiments on the parameter k across three tasks: Backdoor, Image Variation, and Image Inpainting. For the Image Inpainting task, we selected the Latent Diffusion Model as the target for the attacks. The experimental results, as shown in Fig. 6 indicate that as k increases, the F1 Score exhibits only marginal improvement. Based on these findings, we select $k = 3$ for all experiments reported in the main text, as it effectively balances detection performance and computational cost.

I. Visualization of More Results on image variation task.

We visualized additional generation results under different attacks in the image variation task. For clarity, we highlighted the failed cases of our detection method with red boxes. As shown in Fig. 5, most results exhibit the divergence and homogeneity phenomena described earlier. The red-boxed samples, marked as failed cases, are due to their relatively weak attack effectiveness. These samples retain a certain degree of similarity to the conditional image while maintaining a reasonable level of diversity, making them harder to detect.

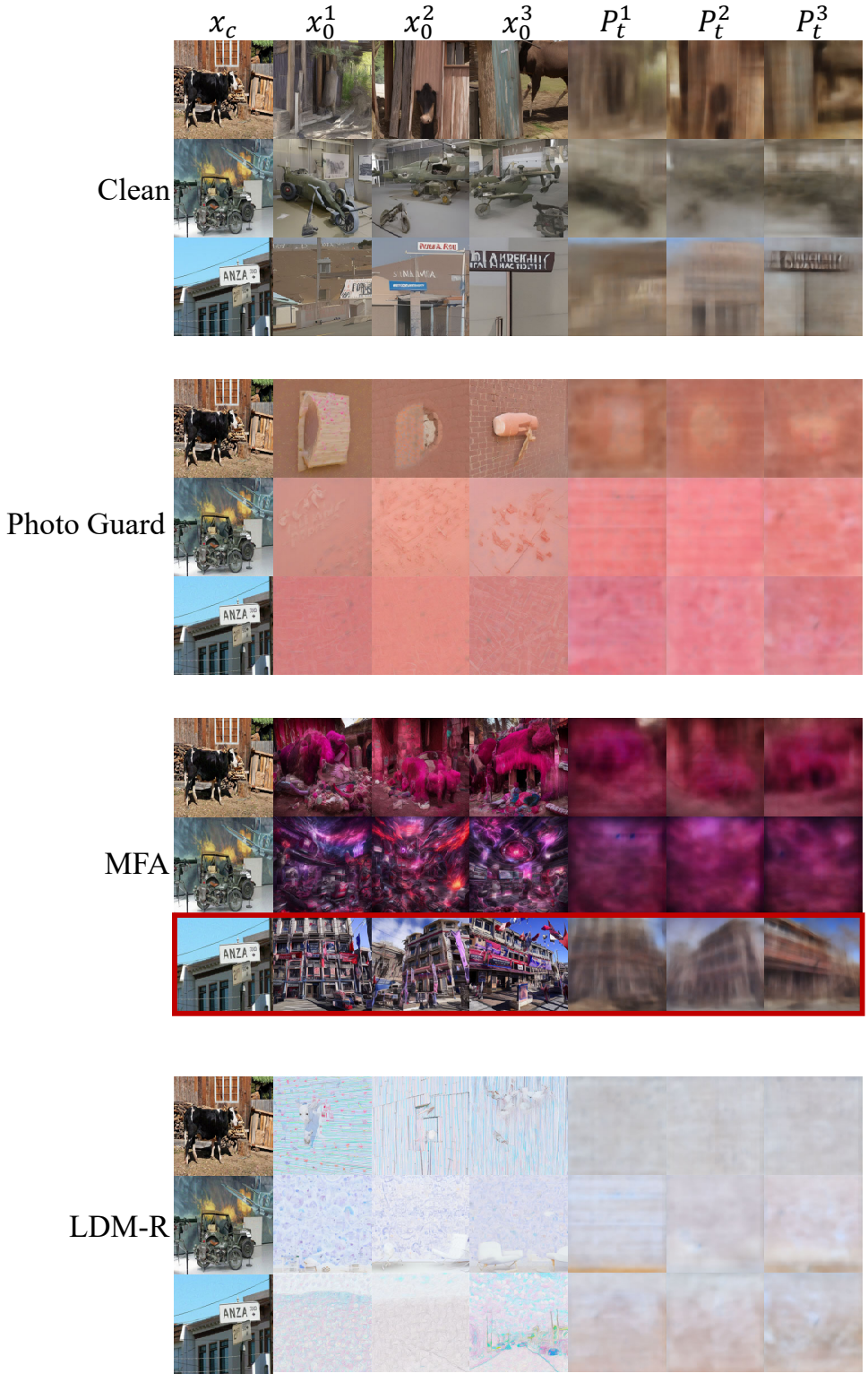


Figure 7. Visualization results of image variation task.

J. Visualization of More Results on image inpainting task.

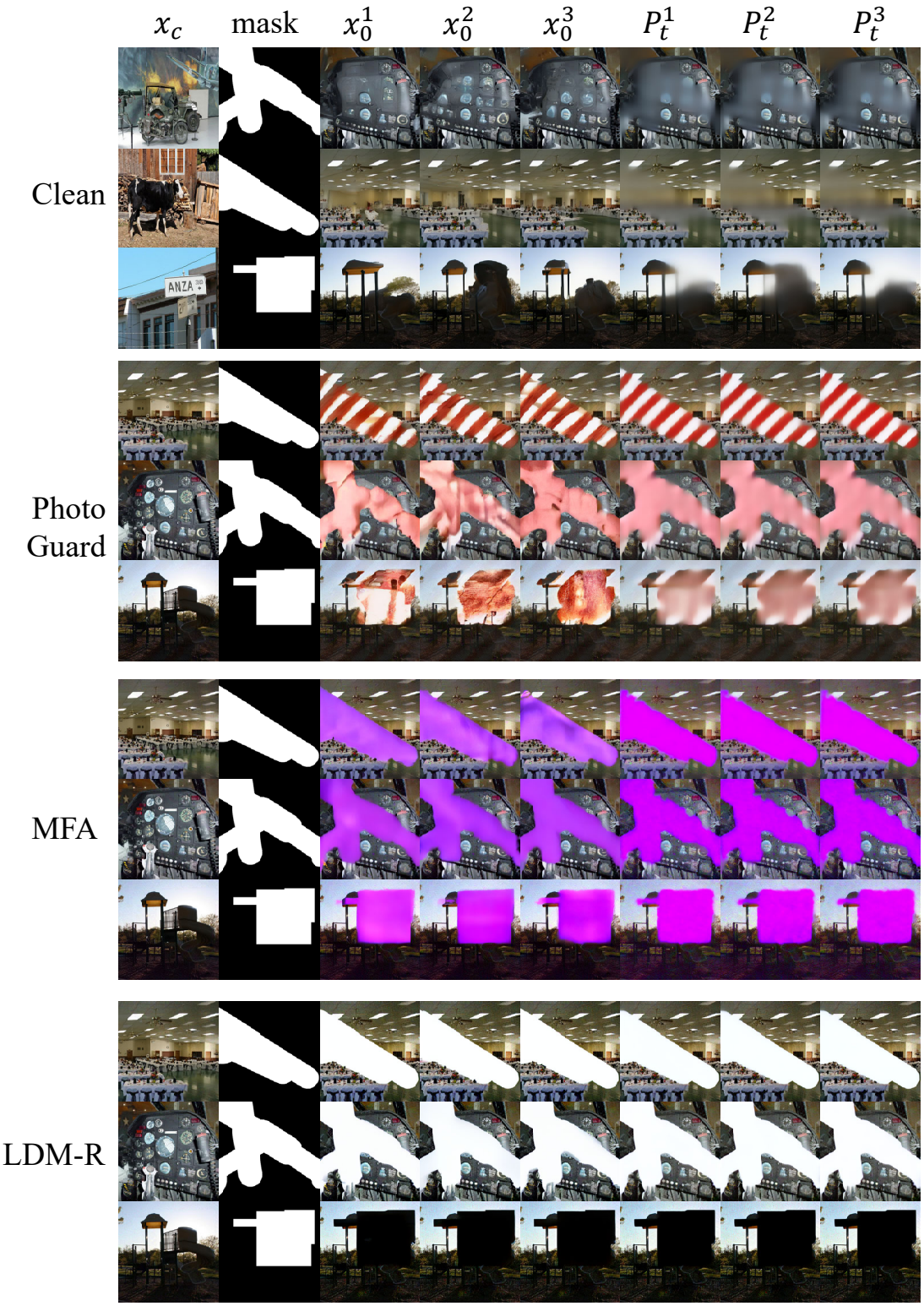


Figure 8. Visualization results of image inpainting task on latent diffusion model.

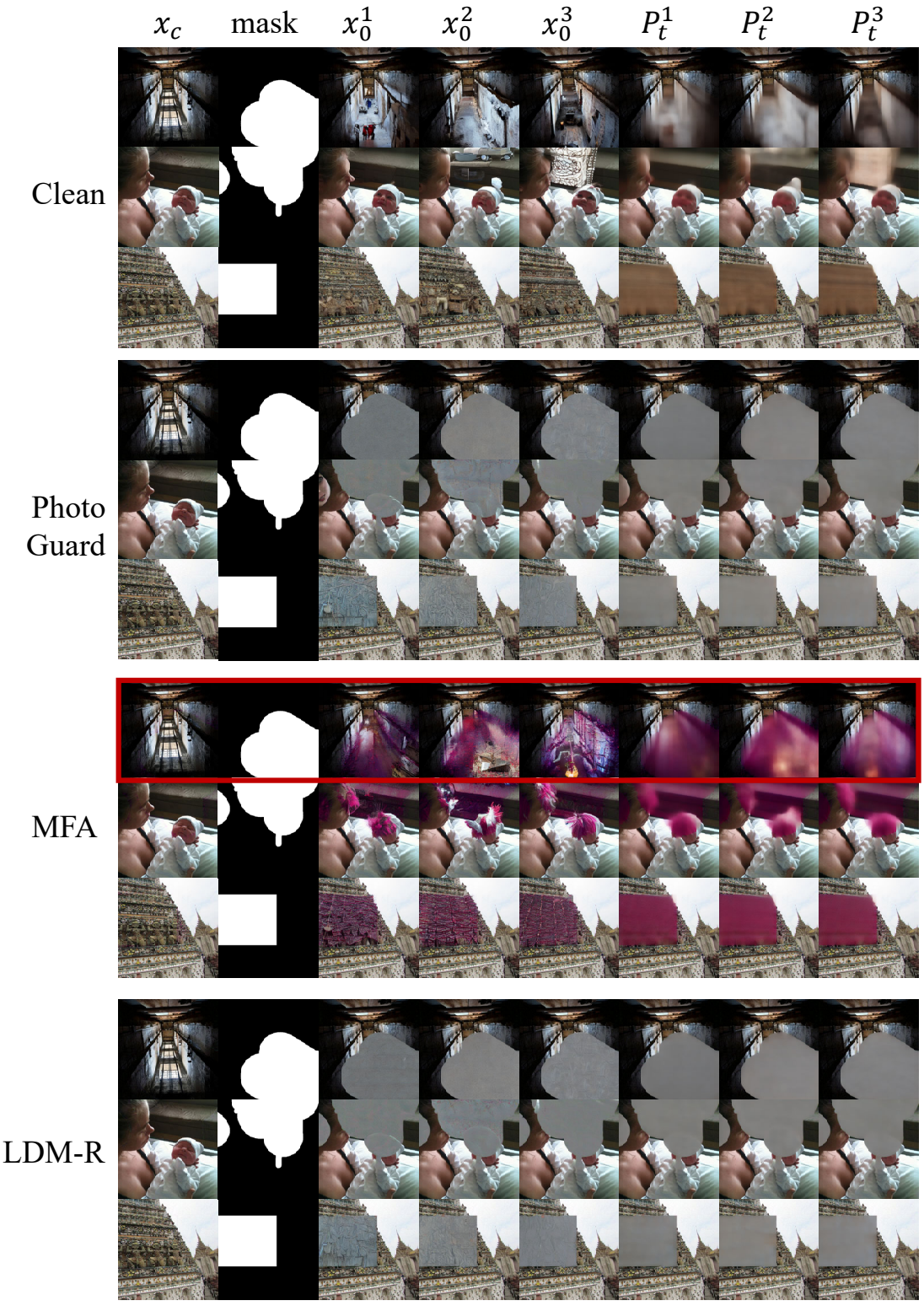


Figure 9. Visualization results of image inpainting task on stable diffusion model.

References

[1] 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*, pages 4015–4024, 2023. 5

[2] 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, 2024. 5

[3] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[4] Sen Li, Junchi Ma, and Minhao Cheng. Invisible backdoor attacks on diffusion models. *arXiv preprint arXiv:2406.00816*, 2024. 5

[5] Hadi Salman, Alaa Khaddaj, Guillaume Leclerc, Andrew Ilyas, and Aleksander Madry. Raising the cost of malicious ai-powered image editing. In *Proceedings of the 40th International Conference on Machine Learning*, pages 29894–29918, 2023. 4

[6] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, Steven Liu, William Berman, Yiyi Xu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. 6