DiffusionGuard: A Robust Defense Against Malicious Diffusion-based Image Editing

Anonymous Authors[1]

Abstract

Recent advances in diffusion models have introduced a new era of text-guided image manipulation, enabling users to create realistic edited images with simple textual prompts. However, there is significant concern about the potential misuse of these methods, especially in creating misleading or harmful content. Although recent defense strategies, which introduce imperceptible adversarial noise to induce model failure, have shown promise, they remain ineffective against more sophisticated manipulations, such as editing with a mask. In this work, we propose DiffusionGuard, a robust and effective defense method against unauthorized edits by diffusion-based image editing models, even in challenging setups. Through a detailed analysis of these models, we introduce a novel objective that generates adversarial noise targeting the early stage of the diffusion process. This approach significantly improves the efficiency and effectiveness of adversarial noises. We also introduce a mask-augmentation technique to enhance robustness against various masks during test time. Finally, we introduce a comprehensive benchmark designed to evaluate the effectiveness and robustness of methods in protecting against privacy threats in realistic scenarios. Through extensive experiments, we demonstrate that our method achieves stronger protection and improved mask robustness with lower computational costs compared to the strongest baseline.

1. Introduction

Text-to-image (T2I) diffusion models trained on large-scale datasets have demonstrated impressive results in generating high-quality images from text prompts (Betker et al., 2023; Sauer et al., 2024; Saharia et al., 2022b). These models have expanded to support text-guided image editing (Wang et al., 2023; Brooks et al., 2023; Yenphraphai et al., 2024), enabling users to modify images with ease. For instance, Image Sculpting (Yenphraphai et al., 2024) identifies 3D objects in photos, enabling great capabilities in altering images. These works improve the user-friendliness of editing tools and allow for precise editing based on text input.

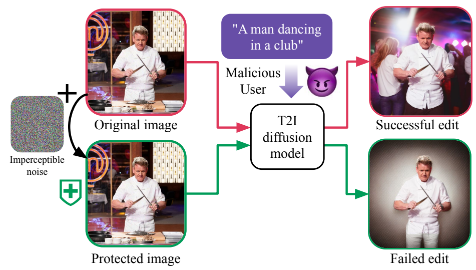


Figure 1. **Misuse of text-to-image models (top) and protection against such misuse (bottom).**

However, a significant concern exists regarding the ability of these models to create highly realistic content, as they can be used for malicious purposes such as spreading fake news. For example, using open-sourced T2I models (Rombach et al., 2023), one could easily manipulate a photo to falsely depict a celebrity dancing with knives in a club (Figure 1). As these models become more powerful, it is paramount to safeguard against these risks.

To mitigate the risks of these models, protection methods based on adversarial noises have shown promise recently (Liang et al., 2023; Liang & Wu, 2023; Salman et al., 2024; Xue et al., 2024). They involve layering images with imperceptible noise designed to cause models to fail in generating high-quality images (see Figure 1). However, current methods do not provide robust protection against real-life scenarios, such as editing with freely chosen masks by malicious users, which can bypass protection. This issue is especially problematic as adversaries may select the smallest possible region containing sensitive identities (e.g., a person's face), minimizing the effect of the protection.

Contributions. In this work, we introduce DiffusionGuard, a robust and effective defense method against text-guided image editing models in challenging setups, such as editing with user-selected masks. Specifically, we propose a

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

novel objective to generate adversarial noises targeting the early stage of the diffusion process. Through our analysis, we have observed that editing models tend to generate key regions within the mask during these initial diffusion steps, which we direct adversarial perturbations, thereby preventing models from maintaining key regions that are crucial for creating high-quality edits. We also propose a mask-augmentation method to find robust adversarial noises effective against masks of various shapes.

For concrete evaluation, we introduce InpaintGuardBench, a challenging benchmark designed to assess defense methods against image editing models. InpaintGuardBench comprises images and handcrafted masks of diverse shapes and texts for editing, enabling a comprehensive evaluation of robustness against various misuse scenarios. We conduct human surveys and measure qualitative metrics to assess DiffusionGuard, and demonstrate both qualitatively and quantitatively that it is effective, and robust against changes in mask inputs, which makes it useful in real-life scenarios.

2. Preliminaries

This section provides an overview of text-to-image diffusion models, emphasizing inpainting models and adversarial examples against them.

2.1. Diffusion models

We consider denoising diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Dhariwal & Nichol, 2021) in discrete time. Suppose $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$ represents the data distribution. A diffusion model defines a sequence of latent variables with noise scheduling functions α_t, σ_t such that the log signal-to-noise ratio $\lambda_t = \log(\alpha_t^2/\sigma_t^2)$ decreases with t. The forward process of diffusion model gradually adds noise to the data $\mathbf{x}$, where the marginal distribution is given as $q(\mathbf{x}_t|\mathbf{x}) = \mathcal{N}(\mathbf{x}_t; \alpha_t\mathbf{x}, \sigma_t^2\mathbf{I})$. The reverse process starts from random noise $\mathbf{x}_T$, and sequentially denoises it to generate $\mathbf{x}_0$, which matches the training distribution.

Text-to-Image diffusion models. Text-to-image (T2I) diffusion models (Rombach et al., 2023; Saharia et al., 2022b; Betker et al., 2023) are a class of diffusion models specifically designed to generate images conditioned on text prompts. These models use text embeddings extracted from pre-trained text encoders like T5 (Raffel et al., 2020) or CLIP (Radford et al., 2021) to guide the generation process. Given a pair of image $\mathbf{x}$ and text $y_{\texttt{text}}$, these models employ a noise prediction model $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t; t)$ and are trained using a noise prediction loss as follows:

$$\mathbb{E}_{t\sim\mathcal{U}(1,T),\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left[\|\boldsymbol{\epsilon}_\theta(\mathbf{x}_t; y_{\texttt{text}}, t) - \boldsymbol{\epsilon}\|_2^2\right]. \quad (1)$$

Text-guided inpainting models. In addition to T2I generation, it is of a great interest to edit a desired region of a given image with text prompts. To this end, T2I image inpainting models (Nichol et al., 2022; Saharia et al., 2022c;a) propose to fine-tune pretrained T2I diffusion model. In specific, inpainting models are fine-tuned by adding conditions of source image $\mathbf{x}_{\texttt{src}}$ and binary mask M that designates the region to infill to the noise prediction loss in Equation 1. During fine-tuning, random regions of image are masked, and the source image and mask are concatenated to the noisy latent $\mathbf{x}_t$ as an input to the model. The training objective of these inpainting models are given as follows:

$$\mathbb{E}_{t\sim\mathcal{U}(1,T),\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left[\|\boldsymbol{\epsilon}_\theta(\mathbf{x}_t; y_{\texttt{text}}, t, M, \mathbf{x}_{\texttt{src}}) - \boldsymbol{\epsilon}\|_2^2\right]. \quad (2)$$

2.2. Adversarial examples against diffusion models

An adversarial example is deliberately fabricated data that manipulates model behaviors (Szegedy et al., 2014; Biggio et al., 2013), often with malicious intent. Given a clean image $\mathbf{x}$, an adversarial example is a perturbation δ such that the input $\mathbf{x} + \delta$ deceives the model. These perturbations are typically crafted to be imperceptible to human eyes, via constrained optimization, e.g., using ℓ_∞ bound $\|\delta\|_\infty \leq \eta$ for some $\eta > 0$. In this paper, we consider an adversarial example for text-guided image editing models, which will enforce them to generate low-quality images. A line of research (Liang et al., 2023; Liang & Wu, 2023; Xue et al., 2024; Salman et al., 2024) has investigated adversarial examples of this purpose, using them as a protective measure against unauthorized image editing. These works either perturb each individual step of the denoising process to maximize the diffusion training loss (Equation 1), or force diffusion models to generate a specific undesirable image as follows:

$$\delta = \underset{\|\delta\|_\infty \leq \eta}{\arg\min}\ \mathbb{E}_{\widehat{\mathbf{x}}\sim f_\theta(\cdot|\mathbf{x}_{\texttt{src}}+\delta, y_{\texttt{text}}, M)}\left[\|\widehat{\mathbf{x}} - \mathbf{x}_{\texttt{target}}\|_2^2\right], \quad (3)$$

where f_θ is the conditional distribution of inpainting model, and $\mathbf{x}_{\texttt{target}}$ is an arbitrary target image.

3. Main method

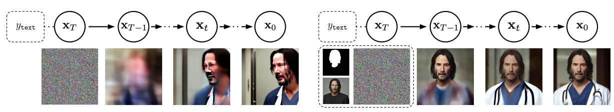
In this section, we outline DiffusionGuard, a method designed to protect images against inpainting methods in challenging scenarios (Section 3.1). First, based on the unique behaviors of inpainting models, we develop a novel objective to target the early stages of the reverse diffusion process (Section 3.2). Next, we propose a mask-augmentation method to find a robust adversarial perturbation that remains effective against mask inputs of various shapes (Section 3.3).

3.1. Problem setup

As described in Section 2.2, previous protection methods (Liang et al., 2023; Liang & Wu, 2023; Xue et al.,

2024) typically consider a global perturbation δ over the entire image, i.e., $\mathbf{x} + \delta$. However, such methods are not effective against diffusion inpainting models, which process the *masked* source image, $(\mathbf{x} + \delta) \odot M$, where M is a binary mask. This realistic setup poses a unique challenge for protection as only adversarial noises that intersect with M can affect the model.

Threat model. We assume that a *malicious user* tries to successfully edit an image protected by adversarial noises applied by a *defender*. This malicious user can freely choose mask M, and text prompt $y_{\texttt{text}}$ for editing. Because it is challenging to develop a defense method against any arbitrary mask, we consider a feasible yet practical setup where there exists a shared common understanding of the *sensitive region* in source image. In a portrait, this could be the face or the body of a person. We assume the defender uses this sensitive region as a training mask $M_{\texttt{tr}}$ in generating adversarial noises, and malicious user can use a different mask but based on the same conceptual sensitive region.



(a) Standard diffusion models (b) Inpainting diffusion models

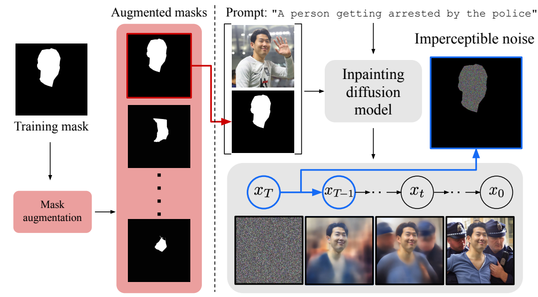
Figure 2. **Denoising process of standard and inpainting diffusion models.** (a) Standard text-to-image models typically generate only coarse features in the early stages of the denoising process. (b) Inpainting models, which are fine-tuned versions of these standard models, produce fine details (e.g., face) from first step ($T-1$).

3.2. Perturbing the early stages of the diffusion process

In this section, we introduce a novel objective that specifically exploits a unique behavior we have observed in inpainting models. As shown in Figure 2a, it is well-known that during the denoising process of diffusion models, coarse features emerge first, and fine details are created in the later stages (Ho et al., 2020; Hertz et al., 2023). However, we have found that this pattern does not hold for inpainting models. Instead, these models first produce fine details (e.g., facial features) even at the first denoising step (Figure 2b).

This unique behavior likely originates from the additional inputs during the fine-tuning process of inpainting models. Unlike standard diffusion models that only receive random noises as input, inpainting models are fine-tuned to utilize two additional inputs: these models take a binary mask $M_{\texttt{tr}}$, and a masked source image $\mathbf{x}_{\texttt{src}} \odot M_{\texttt{tr}}$ as inputs. Then, they are fine-tuned using a reconstruction loss (Equation 2), encouraging them to *copy and paste* the unmasked region of the image, leading to the behavior in Figure 2b.

Inspired by this unique behavior, we develop a novel objective that targets the initial step of the denoising process. Suppose we have a source image $\mathbf{x}_{\texttt{src}}$ to protect, an inpaint-



(a) Mask augmentation (b) Targeting early stages of the denoising process

Figure 3. **Overview of DiffusionGuard.** We propose (a) mask augmentation for improving robustness, and (b) early state perturbation loss for generating effective noises.

ing model $\boldsymbol{\epsilon}_\theta$, and a binary mask $M_{\texttt{tr}}$ which designates the part of the image to keep while rest of the image is recreated. We aim to find an adversarial perturbation δ that maximizes the ℓ_2 norm of *the initial predicted noise* (see Figure 3b):

$$\delta = \underset{||\delta||_\infty \leq \eta}{\arg\max} \left\| \boldsymbol{\epsilon}_\theta(\mathbf{x}_T; y_{\texttt{text}}, T, M_{\texttt{tr}}, \mathbf{x}_{\texttt{src}} + \delta) \right\|_2^2, \quad (4)$$

where T corresponds to the initial denoising step and $\mathbf{x}_T$ is random noise. Our proposed objective focuses on targeting the early stage of the diffusion process, in contrast to prior methods that target the entire diffusion process (Liang et al., 2023) or the output images (Salman et al., 2024). This approach makes generating adversarial noise both efficient and effective because only one forward pass through $\boldsymbol{\epsilon}$ is necessary. Unlike previous methods that aim to maximize reconstruction loss (Equation 1) or minimize the distance to an arbitrary target image (Equation 3), we propose to increase the norm of the noise, which we found is more effective than previous approaches (see Figure 6a).

3.3. Mask-robust adversarial perturbation

In practice, malicious users may utilize a mask that differs from the mask $M_{\texttt{tr}}$ seen during the generation of adversarial noise. Therefore, it is crucial to find robust perturbations that are effective across various mask shapes. To achieve this, we propose a mask augmentation $\mathcal{A}(\cdot)$ that generates a new mask with a similar shape to $M_{\texttt{tr}}$. Specifically, we first obtain the points along the contours of $M_{\texttt{tr}}$ using contour detection. We then move these points inward by random offset to get a new contour, which is filled to form the augmented mask (see Figure 3a). The full procedure is summarized in Algorithm 1.

Using the proposed mask augmentation $\mathcal{A}(\cdot)$, we generate a robust η-bounded adversarial noise δ by minimizing the following loss over the set $\mathcal{M}$ of augmented masks $\mathcal{A}(M_{\texttt{tr}})$:

$$\delta = \underset{||\delta||_\infty \leq \eta}{\arg\max} \mathcal{L}_{\texttt{adv}}(\theta; \mathbf{x} + \delta, M_{\texttt{tr}})$$

where $\mathcal{L}_{\texttt{adv}}$ is our adversarial loss term which we maximize

Figure 4. **Demonstration of qualitative comparison between PhotoGuard (Salman et al., 2024) and DiffusionGuard (Ours).**

in Equation 4.[1] In practice, we stochastically sample masks from $\mathcal{M}$ during the optimization of δ At each iteration, we sample a mask $M \sim \mathcal{A}(M_{\mathtt{tr}})$ and perform a projected gradient descent (PGD) step (Madry et al., 2018) to update δ:

$$\delta \leftarrow \text{Proj}_{||\delta||_\infty \leq \eta} \left(\delta - \gamma \cdot \text{sign}(\nabla_\delta \mathcal{L}_{\text{adv}})\right),$$

where γ is the step size and $\text{Proj}_{||\delta||_\infty \leq \eta}(\cdot)$ projects δ onto the ℓ_∞ ball of radius η. By iteratively updating δ using different masks, we effectively minimize the expected adversarial loss over the set of masks $\mathcal{M}$. This stochastic optimization approach allows us to find a perturbation δ that is robust to various mask shapes similar to $M_{\mathtt{tr}}$.

4. Experiments

4.1. InpaintGuardBench: Inpainting-specialized evaluation benchmark

Benchmark dataset To thoroughly validate the protection effectiveness and mask robustness, we construct a benchmark specialized for masked inpainting models. InpaintGuardBench consists of 30 images, each with 5 unique mask images. 1 mask per image is generated using SAM (Kirillov et al., 2023), a segmentation method, and the other 4 masks are handcrafted using the most common tools that end-users use to draw a mask, which is the *circle brush*, where users select the region by painting on the image with the brush. We consider 10 text edit prompts for each image, resulting in 1,500 edit tasks total.

Setup and evaluation metrics As the target model, we use Stable Diffusion Inpainting (Rombach et al., 2023) (SDI), an open-sourced inpainting diffusion model. For generating adversarial noise δ, we use the SAM-generated mask as the training ("seen") mask. We then evaluate the effectiveness of δ on all 5 masks.

[1] It is applicable to any mask-dependent loss (see Appendix F.2).

Table 1. **Results on InpaintGuardBench.** Our method reaches strong protection in both `Seen` and `Unseen` set, on all metrics (lower the better). All methods were trained with $\|\delta\|_\infty = {}^{16}/_{255}$.

Method	PSNR ↓	CDS ↓	IR ↓	CS ↓
`Seen` (1 Mask, Train set)				
Unprotected	N/A	24.38	-1.365	29.74
PhotoGuard	**13.15**	20.97	-1.562	27.22
DiffusionGuard	13.16	**19.48**	**-1.765**	**26.27**
`Unseen` (4 Masks, Test set)				
Unprotected	N/A	24.34	-1.334	30.49
PhotoGuard	14.84	23.44	-1.374	29.87
DiffusionGuard	**13.72**	**21.78**	**-1.576**	**28.62**

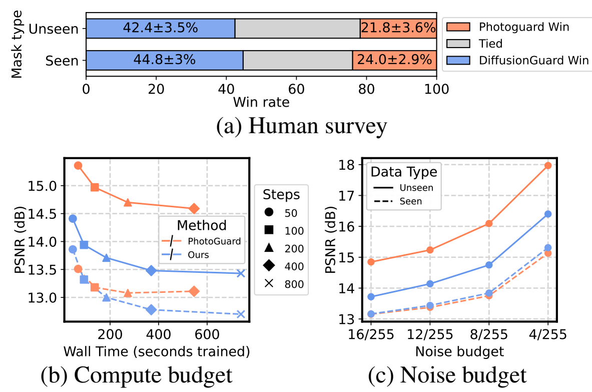
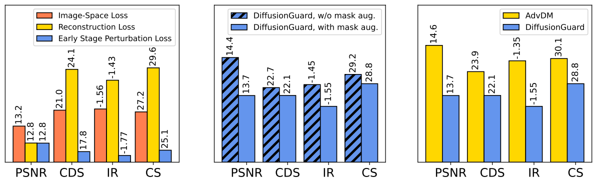


Figure 5. **(a) Human survey results.** We visualize the win rates of DiffusionGuard and PhotoGuard (Salman et al., 2024). **(b), (c) Comparison under limited resources.** We compare PSNR values per varying compute budget (optimization steps represented as markers and time as x axis) and noise budget (x axis).

For evaluation, we employ quantitative metrics to measure the fidelity of the prompt and the quality of the image. Specifically, we use three semantic metrics that evaluate the prompt fidelity as well as image quality of the edit: CLIP directional similarity (CDS), CLIP similarity (CS), and ImageReward (IR), and we also measure PSNR between the edited results of unprotected and protected images, as done by (Salman et al., 2024). For the detailed description of the metrics, please refer to Appendix D. Finally, we compare our method to multiple baseline methods for our experiments, which we elaborate in Appendix D.2.

4.2. Main results

We compare DiffusionGuard with PhotoGuard (Salman et al., 2024) on InpaintGuardBench. For all experiments, we ensure a fair comparison by running the protection methods for an equal amount of GPU time.



(a) Training loss comparison (b) Mask augmentation comparison (c) Against mask-free protection

Figure 6. Ablation study reporting quantative metrics (lower the better). **(a) Comparison of loss functions.** `Seen` set results of the three loss functions with the same training mask are shown. **(b) Effect of mask augmentation.** On `Unseen` set, we present the effect of mask augmentation. **(c) Comparison to mask-free protection.** We show the effectiveness of using mask-free (Liang et al., 2023) and mask-dependent protection (DiffusionGuard) on the `Unseen` set.

As shown in Figure 4, DiffusionGuard demonstrates its robustness against mask changes, in contrast to PhotoGuard, which loses its protective effectiveness with even small deviations in mask shape. Notably, the protected results of DiffusionGuard effectively prevent the diffusion inpainting model from 'recognizing' the object, evident in the last example where another dog is drawn over the original.

Table 1 shows that DiffusionGuard exhibits stronger protection than PhotoGuard for both mask categories. Note that PhotoGuard loses effectiveness for `Unseen` masks compared to `Seen` masks, in line with Figure 4.

We conduct human survey by asking raters to indicate which result among PhotoGuard and DiffusionGuard is *worse* (order shuffled) or tie for all 1500 edit results, assessing image quality and prompt fidelity. DiffusionGuard results in a superior win rate against PhotoGuard in both categories, with 20% win rate gap (Figure 5a).

4.3. Comparison under resource-restricted scenarios

In this section, we compare our method against baselines in two resource-restricted scenarios. First, we evaluate each method with $50, 100, 200, 400$ PGD iterations (and also 800 steps for our method) to compare computational efficiency. Second, we evaluate each method under a limited noise budget by setting the noise threshold value $\|\delta\|_\infty$ to $4/255, 8/255, 12/255$, and $16/255$ in order to compare the effectiveness under tighter noise constraints.

Comparison under limited compute budget Figure 5b shows that DiffusionGuard is more effective than PhotoGuard when optimized for equal number of steps. Specifically, our method with 50 iterations (taking 46 seconds) achieves a similar PSNR of PhotoGuard with 400 iterations (taking 546 seconds). Note that the gap between `Unseen` and `Seen` mask set is notably smaller for DiffusionGuard. These results show that our method is faster, cheaper, and more effective than PhotoGuard.

Comparison under limited noise budget Figure 5c shows that DiffusionGuard consistently achieves stronger performance (lower PSNR) under tighter noise budget. Our method with a noise budget of $8/255$ is very close to PhotoGuard using a higher budget of $16/255$. These results show that DiffusionGuard maintains strong protection even with reduced perturbations (i.e., less visible), making it suitable for real-life application where less detectable noise and preserving the original image are crucial.

4.4. Ablation study

We conduct a comprehensive analysis on the effects of loss functions, mask augmentation, and the efficacy of using an inpainting-specialized method.

To verify the effectiveness of our early stage perturbation loss (Equation 4) in generating $delta$, we compare it with image-space loss (Salman et al., 2024) and reconstruction loss (Liang et al., 2023). To isolate the effects of mask augmentation, we use a single fixed mask $M_{\texttt{tr}}$ and evaluate using the `Seen` set. As shown in Figure 6a, our loss consistently outperforms both losses across all metrics.

Additionally, we evaluate the effects of mask augmentation on protection strength in `Unseen` mask set by comparing DiffusionGuard with and without mask augmentation. Figure 6b shows that mask augmentation consistently improves all metrics for the `Unseen` set of InpaintGuardBench, clearly enhancing mask robustness. We provide qualitative results in Appendix F.1.

We compare DiffusionGuard with a mask-free protection method that applies a global perturbation over the entire image. As a baseline, we use AdvDM (Liang et al., 2023), a mask-free protection based on reconstruction loss (Equation 1). Figure 6c shows that DiffusionGuard, by focusing on the mask region, provides much stronger protection than AdvDM on all metrics. We also remark that the noise perceptibility is much lower with DiffusionGuard, as it adds δ over a smaller region, whereas in AdvDM, δ occupies the entire image, with $\|\delta\|_\infty$ identical.

5. Conclusion

In this work, we propose DiffusionGuard, a robust and effective defense method against diffusion-based image editing models. By leveraging mask augmentation and early stage perturbation loss, our method achieves stronger protection and improved mask robustness with lower computational costs, compared to several baselines. Additionally, DiffusionGuard also proves effective in black-box settings (see Appendix G). We believe that our work ensures the ethical use and deployment of text-guided inpainting models.

References

AUTOMATIC1111. Stable diffusion web ui, 2022. URL https://github.com/AUTOMATIC1111/stable-diffusion-webui.

Betker, J., Goh, G., Jing, L., Brooks, T., Wang, J., Li, L., Ouyang, L., Zhuang, J., Lee, J., Guo, Y., et al. Improving image generation with better captions. In *Annual Conference of the Association for Computational Linguistics*, 2023.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., and Roli, F. Evasion attacks against machine learning at test time. In Blockeel, H., Kersting, K., Nijssen, S., and Železný, F. (eds.), *Machine Learning and Knowledge Discovery in Databases*, 2013.

Black, K., Janner, M., Du, Y., Kostrikov, I., and Levine, S. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, 2024.

Brooks, T., Holynski, A., and Efros, A. A. Instructpix2pix: Learning to follow image editing instructions. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2023.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. In *Advances in Neural Information Processing Systems*, 2021.

Fan, Y., Watkins, O., Du, Y., Liu, H., Ryu, M., Boutilier, C., Abbeel, P., Ghavamzadeh, M., Lee, K., and Lee, K. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. In *Advances in Neural Information Processing Systems*, 2023.

Gal, R., Patashnik, O., Maron, H., Bermano, A. H., Chechik, G., and Cohen-Or, D. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Trans. Graph.*, 2022.

Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., and Cohen-Or, D. Prompt-to-prompt image editing with cross attention control. In *International Conference on Learning Representations*, 2023.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollar, P., and Girshick, R. Segment anything. In *IEEE International Conference on Computer Vision*, 2023.

Li, J., Li, D., Xiong, C., and Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, 2022.

Liang, C. and Wu, X. Mist: Towards improved adversarial examples for diffusion models. *arXiv preprint arXiv:2305.12683*, 2023.

Liang, C., Wu, X., Hua, Y., Zhang, J., Xue, Y., Song, T., Xue, Z., Ma, R., and Guan, H. Adversarial example does good: Preventing painting imitation from diffusion models via adversarial examples. In *International Conference on Machine Learning*, 2023.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., and Chen, M. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, 2022.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21 (140):1–67, 2020.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2023.

Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., and Aberman, K. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2023.

Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., and Norouzi, M. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, 2022a.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S. K. S., Ayan, B. K., Mahdavi, S. S., Lopes, R. G., et al. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*, 2022b.

Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D. J., and Norouzi, M. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4713–4726, 2022c.

Salman, H., Khaddaj, A., Leclerc, G., Ilyas, A., and Madry, A. Raising the cost of malicious ai-powered image editing. In *International Conference on Machine Learning*, 2024.

Sauer, A., Boesel, F., Dockhorn, T., Blattmann, A., Esser, P., and Rombach, R. Fast high-resolution image synthesis with latent adversarial diffusion distillation. *arXiv preprint arXiv:2403.12015*, 2024.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, 2015.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Wang, S., Saharia, C., Montgomery, C., Pont-Tuset, J., Noy, S., Pellegrini, S., Onoe, Y., Laszlo, S., Fleet, D. J., Soricut, R., et al. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2023.

Xu, J., Liu, X., Wu, Y., Tong, Y., Li, Q., Ding, M., Tang, J., and Dong, Y. Imagereward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, 2023.

Xue, H., Liang, C., Wu, X., and Chen, Y. Toward effective protection against diffusion based mimicry through score distillation. In *International Conference on Learning Representations*, 2024.

Ye, J., Liu, F., Li, Q., Wang, Z., Wang, Y., Wang, X., Duan, Y., and Zhu, J. Dreamreward: Text-to-3d generation with human preference. *arXiv preprint arXiv:2403.14613*, 2024.

Yenphraphai, J., Pan, X., Liu, S., Panozzo, D., and Xie, S. Image sculpting: Precise object editing with 3d geometry control. *arXiv preprint arXiv:2401.01702*, 2024.

Appendix:

DiffusionGuard: A Robust Defense Against Malicious Diffusion-based Image Editing

A. Ethics statement and broader impact

Ethics statement In this work, we propose DiffusionGuard, a robust and effective defense method against diffusion-based image editing models. This defense method has the potential to be both socially beneficial and harmful, depending on its usage. While it allows users to protect their images from unauthorized editing, adversaries might develop methods to bypass our protections. Therefore, it is crucial to carefully manage the dissemination of our method to ensure its responsible and ethical implementation.

Broader impact Our work aims to develop robust defense mechanisms against AI-based image manipulation methods. By ensuring stronger protection and robustness against image editing models, we believe that our research contributes to the ethical use and deployment of generative AI technologies.

B. Mask augmentation algorithm

The full procedure of mask augmentation is summarized in Algorithm 1.

Algorithm 1 Mask augmentation via contour shrinking

```
Input: Training mask M_tr, perturbation range ζ, smoothing parameter s, iterations N
M ← M_tr
for i = 1 to N do
  P ← findContours(M)
  P_orig ← P
  X_offset, Y_offset ~ U(−ζ, ζ) ∀(x_i, y_i) ∈ P
  // Random offsets
  X_offset, Y_offset ← GaussianFilter(X_offset, s), GaussianFilter(Y_offset, s)
  // Smooth out
  for each point (x_i, y_i) ∈ P do
    (x_i, y_i) ← (x_i + X_offset[i], y_i + Y_offset[i])
  end for
  for each point (x_i, y_i) ∈ P do
    // Ensure P stays within the original mask
    if M_tr[y_i, x_i] = 0 then
      // Point is outside the mask
      (x_i^closest, y_i^closest) ← closest point to (x_i, y_i) on P_orig
      (x_i, y_i) ← (x_i^closest, y_i^closest)
    end if
  end for
  M ← mask from new contour P
end for
return M
```

C. InpaintGuardBench

To assess the ability of a protection method to prevent unauthorized adversaries from editing an image in a challenging yet practical scenario as outlined in Section 3.1, we construct a benchmark out of various images, mask shapes, and edit prompt instructions.

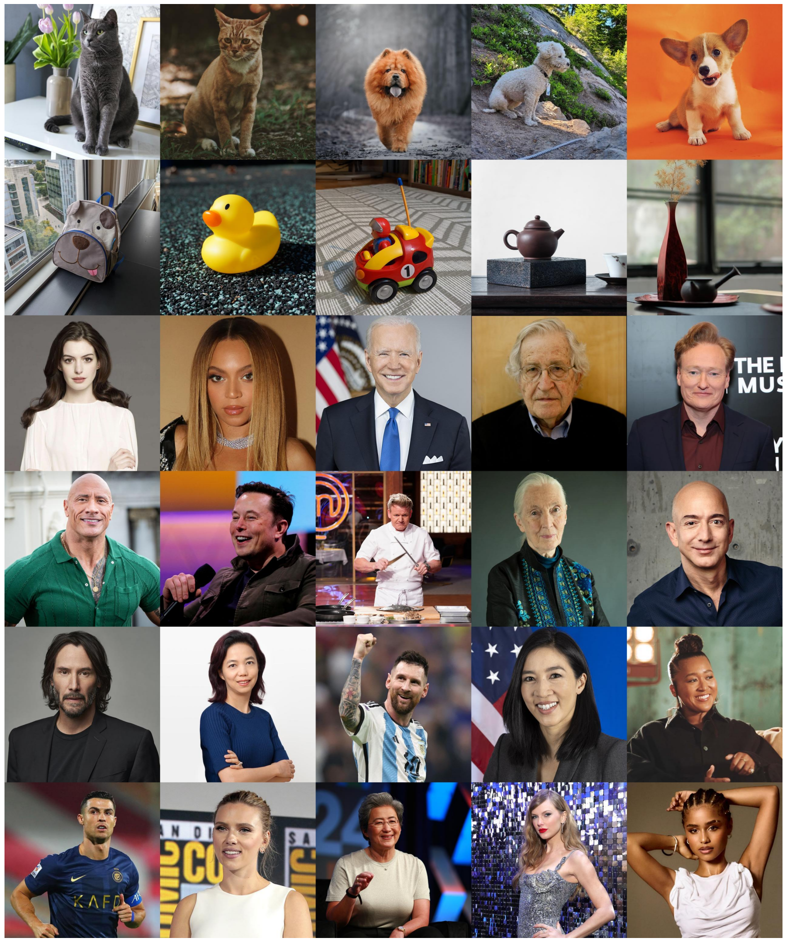


Figure 7. **All images used in InpaintGuardBench.** Best seen zoomed in.

C.1. Dataset

Images To take into account realistic scenarios of privacy threat posed by inpainting models, we collect 30 images consisting of 20 images of celebrities and 10 images of non-human objects. The 20 celebrity images were collected from the web, and consists of 10 males and 10 females of racial and domain diversity. Out of each, 8 images are focused on faces, and 2 images focus on the body of the person. 10 non-human images were sourced from the DreamBooth (Ruiz et al., 2023) dataset. Out of them, 5 images contain animals, and 5 images include inanimate objects. We visualize all images that we have used in Figure 7 and all masks that we have used in Figure 8.

Masks In order to measure the robustness of a protection method against mask variations, we prepare 5 masks per image. For the first mask, we obtain a training mask $M_{\texttt{tr}}$, which determines the *sensitive region* (e.g. face, body or object) using an automated segmentation tool (Kirillov et al., 2023) (see Section 3.1 for more details about the definition of the sensitive region). This mask is used for training in both DiffusionGuard and the baselines. For the remaining 4 masks, we handcraft 4 additional masks that contains the same sensitive region. The handcrafted masks are drawn using either circle brush or simple shapes such as rectangles or circles. Circle brush a simple yet the most commonly used user interface (UI) to draw a mask, and it is used by popular inpainting tools such as DALL-E 3 ChatGPT integration (Betker et al., 2023), DALL-E 2 playground (Ramesh et al., 2022), or Stable Diffusion web UI (AUTOMATIC1111, 2022).

Edit text prompts Finally, we use 10 different editing text prompts in order to take into account the robustness of each protection method against different editing prompt choices. All prompts are available in Table 2 and Table 3.

A [man/woman] in a hospital
A [man/woman] riding a motorcycle
A [man/woman] walking in the street
A [man/woman] driving a car
A [man/woman] dancing in a club
A [man/woman] dressed up in halloween costume
A [man/woman] in the gym
A [man/woman] in a gaming convention
A photo of a construction worker
A [man/woman] getting on a bus

Table 2. All prompts for portrait images.

A [object] in a hospital
A [object] on a motorcycle
A [object] in the street
A [object] in a car
A [object] in a club
A [object] in halloween
A [object] in the gym
A [object] in a gaming convention
A photo of [object] at a construction site
A [object] on a bus

Table 3. All prompts for non-portrait images.

D. Evaluation details

D.1. Quantitative metrics

In order to quantitatively measure the protection strength of each method, we employ multiple metrics in order to measure both edit instruction fidelity and edit image quality (i.e. how realistic the generated image is). Because these metrics measure the degree of alignment, and our goal is to stop adversaries from obtaining desirable edits, these metrics should be lower if the protection is better.

CLIP similarity Contrastive Language-Image Pre-training (CLIP) (Radford et al., 2021) is a set of vision and text encoder trained together to align vision and text representations. To measure edit instruction fidelity, we calculate the cosine similarity between the textual description $\text{CLIP}_{\texttt{text}}(y_{\texttt{edit}})$ and the actual edited image representation $\text{CLIP}_{\texttt{image}}(\mathbf{x}_{\texttt{edit}})$, where $\mathbf{x}_{\texttt{edit}}$ is the edit result image, and CLIP is the CLIP encoder. Higher similarity scores indicate that the edit more closely aligns with the desired instruction. This metric helps us evaluate how accurately the edits reflect the specified changes.

CLIP directional similarity CLIP directional similarity (Gal et al., 2022) is a metric specifically intended to measure the performance of a text-guided image editing model. Specifically, CLIP directional similarity measures the alignment between the deviation in the text space (from the source caption to the edit instruction) and the deviation in the image space (from the source to the edited result). The source caption is a caption that describes the source image and in our case, it is obtained using BLIP-Large model (Li et al., 2022), which is an open-source captioning model. The formulation of CLIP directional similarity can be written as follows:

$$\text{CLIP directional similarity} = \frac{(\mathbf{e}_{\text{image, edit}} - \mathbf{e}_{\text{image, source}}) \cdot (\mathbf{e}_{\text{text, edit}} - \mathbf{e}_{\text{text, source}})}{\|\mathbf{e}_{\text{image, edit}} - \mathbf{e}_{\text{image, source}}\| \|\mathbf{e}_{\text{text, edit}} - \mathbf{e}_{\text{text, source}}\|}.$$

ImageReward ImageReward (Xu et al., 2023) is a human-aligned vision-language model and a reward model, which is fine-tuned on a human preference dataset. As stated and used by several works (Ye et al., 2024; Fan et al., 2023; Black et al., 2024), ImageReward is suitable for evaluating edit prompt fidelity as well as overall image quality, and shows improvement especially in terms of the ability to measure prompt-image alignment.

PSNR Peak Signal-to-Noise Ratio (PSNR) is a widely used metric to assess the similarity between two images by calculating the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the quality of its representation. In our context, PSNR is used to measure the similarity between the edit result of an unprotected clean image $\mathbf{x}_{\texttt{edit}}$ and a protected image $\mathbf{x}_{\texttt{src}} + \delta$. This serves as an indicator of how much the protection alters the edited result compared to the edited result of a clean image. PSNR is defined as follows:

$$\text{PSNR}(\mathbf{x}_{\text{edit, protected}}, \mathbf{x}_{\text{edit, unprotected}}) = 20 \cdot \log_{10}\left(\frac{\text{MAX}(\mathbf{x}_{\text{edit, unprotected}})}{\sqrt{\text{MSE}(\mathbf{x}_{\text{edit, unprotected}}, \mathbf{x}_{\text{edit, protected}})}}\right)$$

where $\text{MAX}(\mathbf{x}_{\text{edit, unprotected}})$ is the maximum possible pixel value of the unprotected edited result image, and MSE is the mean squared error. Lower PSNR values indicate that the edited result of the protected image is different from the edited result of the unprotected image, indicating that the protection alters the edited result of the image.

D.2. Baselines

We use multiple baselines for our experiments. PhotoGuard (Salman et al., 2024), which is our primary baseline, is a protection method targeting inpainting models. It forces these models to generate an undesirable edit result as formulated by Equation 3. We also consider AdvDM (Liang et al., 2023), a protection method that targets standard text-to-image diffusion models. This method targets each of the denoising steps to maximize the reconstruction loss (Equation 1), originally without considering specific mask regions. While AdvDM proposes to add perturbation to the entire image, we adapt it to introduce perturbations only within the mask region $M_{\texttt{tr}}$, and we report results for both the original and the modified approaches.

D.3. Human survey

In order to assess the edited result of the protected images perceived by human eyes, we perform a human survey with the 1,500 edit instances from InpaintGuardBench. We collected 4,500 labels from 3 individuals. An edit instance is defined by a triplet of (source image, mask, edit instruction), with fixed random seed value. We draw one edit instance from each of the two methods that are compared and present them to the rater in a shuffled order. Then, the rater is instructed to choose the method with *worse* edited result in terms of the criteria, or whether it is tie. For detailed explanation about the human survey criteria, refer to Appendix D.3.

We created a labeling tool using Python and OpenCV, which allowed raters to focus solely on answering the survey, as individual was assigned large amount of questions (1,500 comparisons). The raters were instructed to use keyboard shortcuts to answer with either "left" or "right" or "tie" to choose which edited result is worse. On average, human survey took roughly 3 hours per rater.

Human survey criteria The purpose of the protection is to prevent adversaries from achieving desired edit results that are aligned with their edit instructions, and are natural and realistic enough to spread malicious information. In order to directly assess this, we ask raters to choose the edit result that is *worse* in terms of the following criteria. The actual instruction given to the raters are visualized in Figure 9.

- Edit prompt fidelity: Raters are instructed to assess how *misaligned* the edit result image and the edit prompt are.

- Overall image quality: Raters are instructed to assess how *bad* the edited image quality is, and how *unnatural* and *unrealistic* the edited image is.

Baseline for human survey For the baseline, we choose PhotoGuard (Salman et al., 2024) as our baseline, as (1) PhotoGuard achieves the best result overall in terms of quantitative metrics as presented in Table 1 and Figure 6, which is also visually notable, and (2) PhotoGuard proposed to target the diffusion model in a mask-dependent manner, which is more aligned with our setup outlined in Section 3.1, allowing a fairer comparison in contrast to other baselines, which are not necessarily mask-specific.

E. Experimental details

In this section, we outline the experimental details of our experimental setup for reproducibility. We conduct all our experiments on a single NVIDIA H100 80GB HBM3 GPU. For fair comparison, we match the time taken for running PGD optimization in all apple-to-apple comparison experiments, which is all experiments except for Figure 5b. All comparison of edited results are done by fixing the random seed.

F. More experimental results

F.1. More editing results

In this section, we include additional editing results using DiffusionGuard. We attach the additional editing results in Figure 12, Figure 13, and Figure 14.

F.2. Additional analysis of mask augmentation

Mask augmentation early stage perturbation loss For a detailed analysis of the effect of mask augmentation, we report DiffusionGuard with and without mask augmentation in both `Seen` and `Unseen` sets. Interestingly, while mask augmentation slightly degrades the performance of the protection in the case of `Seen` masks, it improves the protection in the case of `Unseen` masks.

Mask augmentation with image-space loss As noted in Section 3.3, our mask augmentation can be used together with any mask-dependent loss function. Thus, we experiment with PhotoGuard loss function, which is an image-space loss function, applied together with mask augmentation and visualize the results for both `Seen` and `Unseen` set in Figure 11. Similarly to Figure 10, mask augmentation causes performance decrease in `Seen` set and strengthens it in `Unseen` set.

G. Transferability to black-box models

In this section, we show that DiffusionGuard can be transferred across models. Specifically, we use Stable Diffusion Inpainting 1.0 (Rombach et al., 2023) for generating adversarial examples, and test them on Stable Diffusion Inpainting 2.0. As shown in Figure 15, DiffusionGuard can prevent editing against black-box models.

H. Limitation

There are several limitations and interesting future directions in our work:

- **Black-box setups**: Although we demonstrate the effectiveness of DiffusionGuard in black-box settings in Appendix G, further investigations are required against more advanced closed models, such as DALL-E 3 (Betker et al., 2023).
- **Extension to personalization**: Text-to-image diffusion models have shown remarkable success in generating personalized subjects based on a few reference images (Ruiz et al., 2023). Because such personalized models can be misused to generate harmful content, developing defense methods against personalization methods would be an important direction for future research.

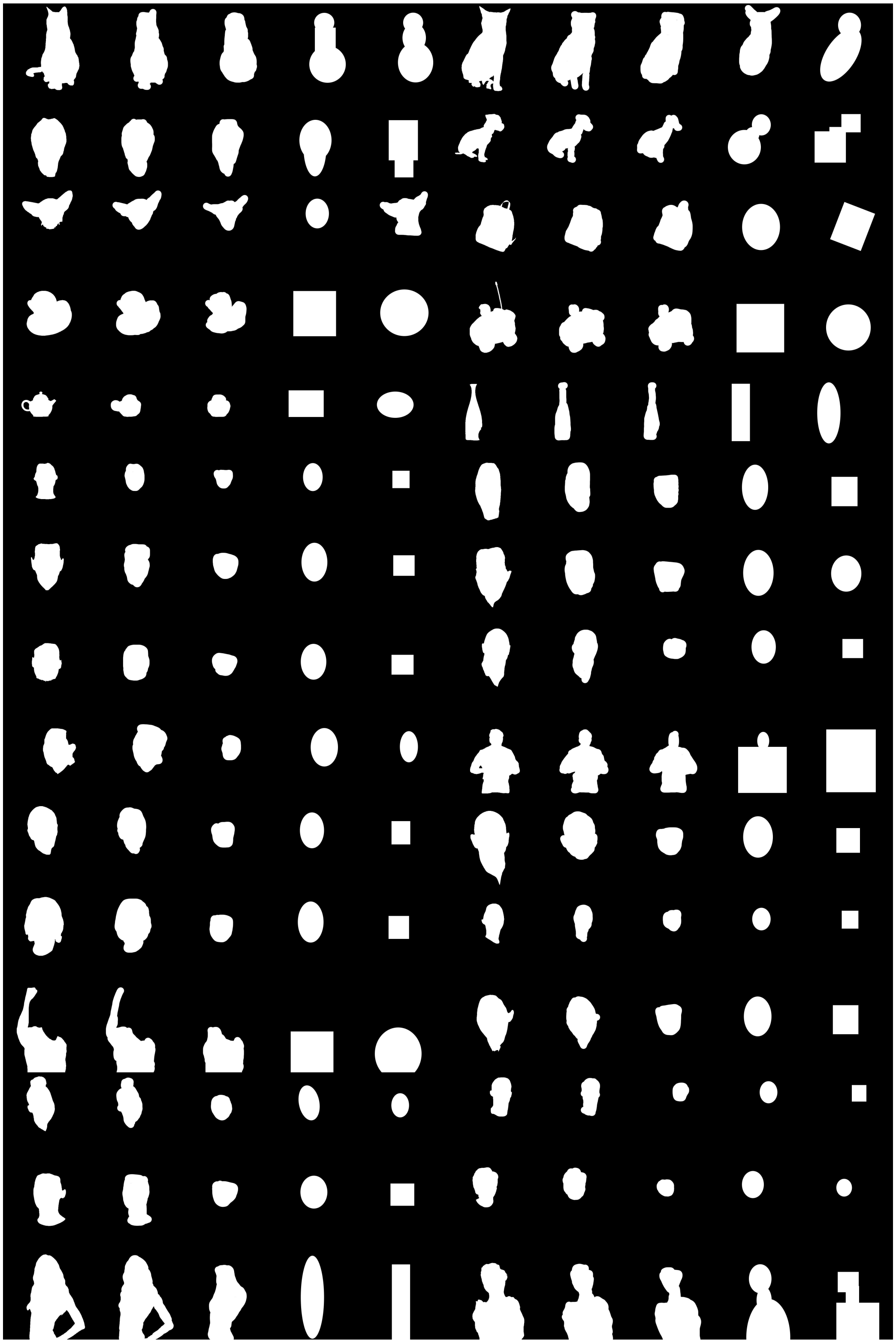


Figure 8. **All masks used in InpaintGuardBench.** Best seen zoomed in.

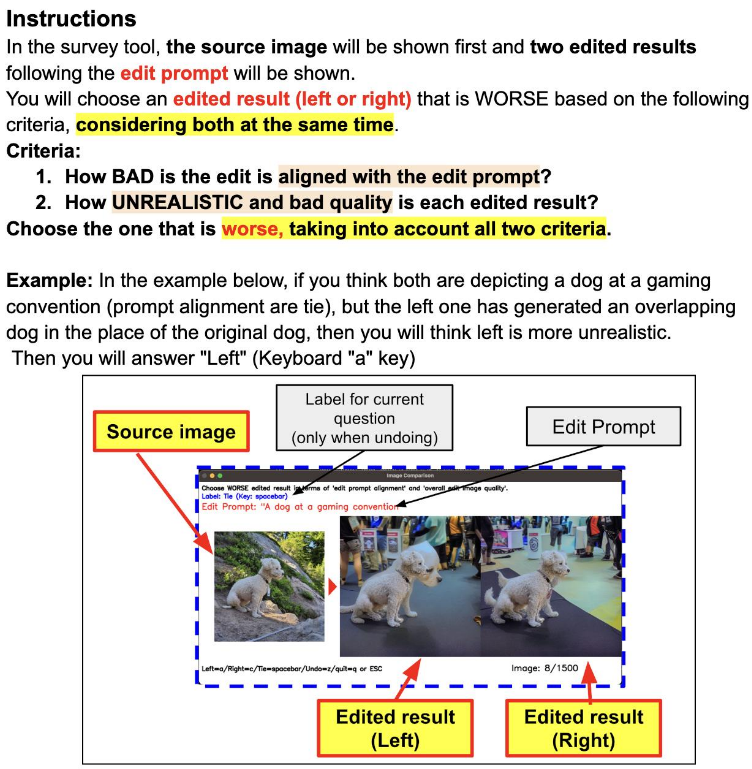
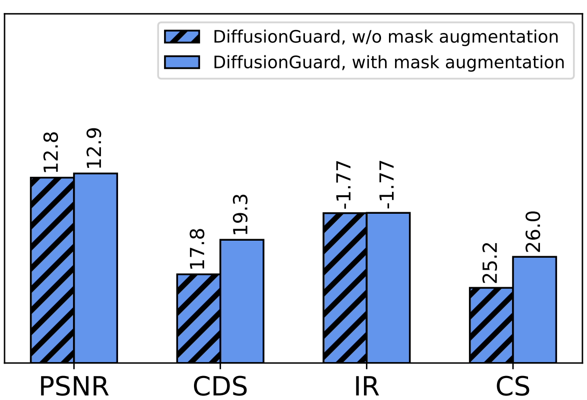
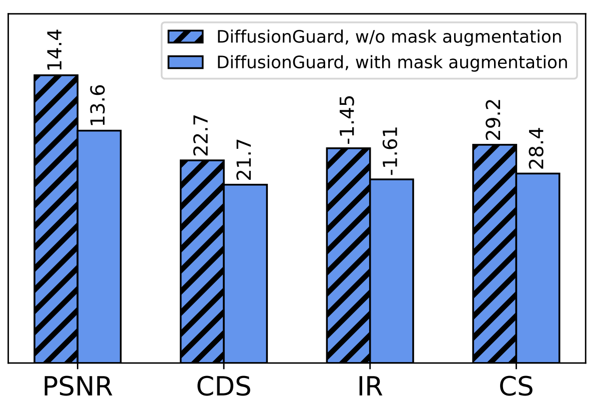


Figure 9. **Screenshot of the instruction provided to human raters.**

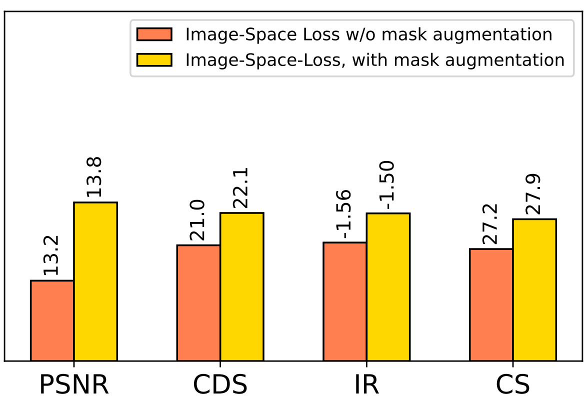


(a) Early stage perturbation loss with and without mask augmentation, `Seen` set results

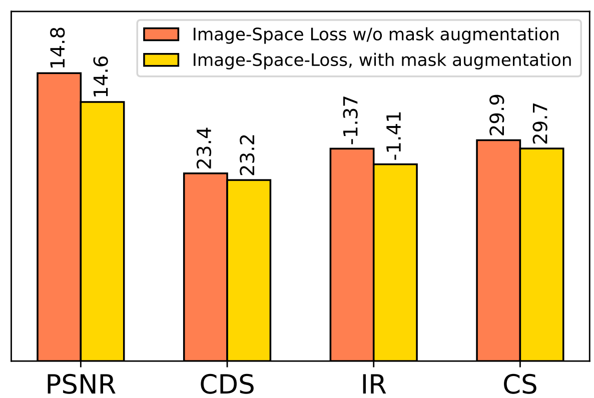


(b) Early stage perturbation loss with and without mask augmentation, `Unseen` set results

Figure 10. **(a) `Seen` set results for DiffusionGuard with and without mask augmentation. (b) `Unseen` set results for DiffusionGuard with and without mask augmentation.**



(a) Image-space loss with and without mask augmentation, `Seen` set results



(b) Image-space loss with and without mask augmentation, `Unseen` set results

Figure 11. **(a) `Seen` set results for PhotoGuard (Salman et al., 2024) with and without mask augmentation. (b) `Unseen` set results for PhotoGuard (Salman et al., 2024) with and without mask augmentation.**

Figure 12. **Edited results for all 5 masks using PhotoGuard (Salman et al., 2024) and DiffusionGuard.** First row is the `Seen` mask, and the rest are `Unseen` masks. Text prompt is `"A man getting on a bus"`.

Figure 13. **Edited results for all 5 masks using PhotoGuard (Salman et al., 2024) and DiffusionGuard.** First row is the `Seen` mask, and the rest are `Unseen` masks. Text prompt is `"A woman in a hospital"`.

Figure 14. **Edited results for all 5 masks using PhotoGuard (Salman et al., 2024) and DiffusionGuard.** First row is the `Seen` mask, and the rest are `Unseen` masks. Text prompt is `"A man walking in the street"`.

Figure 15. **Black-box transfer to Stable Diffusion Inpainting 2.0 from Stable Diffusion Inpainting, comparison of PhotoGuard (Salman et al., 2024) and DiffusionGuard.** All rows are `Seen` mask.