UnGANable: Defending Against GAN-based Face Manipulation

Zheng Li, *CISPA Helmholtz Center for Information Security;* Ning Yu, *Salesforce Research;* Ahmed Salem, *Microsoft Research;* Michael Backes, Mario Fritz, and Yang Zhang, *CISPA Helmholtz Center for Information Security*

https://www.usenix.org/conference/usenixsecurity23/presentation/li-zheng

This paper is included in the Proceedings of the 32nd USENIX Security Symposium.

August 9–11, 2023 • Anaheim, CA, USA

978-1-939133-37-3

Open access to the Proceedings of the 32nd USENIX Security Symposium is sponsored by USENIX.

UnGANable: Defending Against GAN-based Face Manipulation

Zheng Li[1] Ning Yu[2] Ahmed Salem[3] Michael Backes[1] Mario Fritz[1] Yang Zhang[1]

[1]*CISPA Helmholtz Center for Information Security*
[2]*Salesforce Research* [3]*Microsoft Research*

Abstract

Deepfakes pose severe threats of visual misinformation to our society. One representative deepfake application is face manipulation that modifies a victim's facial attributes in an image, e.g., changing her age or hair color. The state-of-the-art face manipulation techniques rely on Generative Adversarial Networks (GANs). In this paper, we propose the first defense system, namely `UnGANable`, against GAN-inversion-based face manipulation. In specific, `UnGANable` focuses on defending GAN inversion, an essential step for face manipulation. Its core technique is to search for alternative images (called cloaked images) around the original images (called target images) in image space. When posted online, these cloaked images can jeopardize the GAN inversion process. We consider two state-of-the-art inversion techniques including optimization-based inversion and hybrid inversion, and design five different defenses under five scenarios depending on the defender's background knowledge. Extensive experiments on four popular GAN models trained on two benchmark face datasets show that `UnGANable` achieves remarkable effectiveness and utility performance, and outperforms multiple baseline methods. We further investigate four adaptive adversaries to bypass `UnGANable` and show that some of them are slightly effective.[1]

1 Introduction

Nowadays, machine learning (ML) models have become a core component for many real-world applications, ranging from image classification [18, 29] to recommendation systems [19, 53]. One major advancement of ML techniques in the image domain is deep generative models. The resolution and quality of generated images have been improved exponentially since the introduction of Generative Adversarial Networks (GANs) [15]. Although realistic synthetic images can be used for various applications, e.g., virtual reality, avatars, and games, and detrimental uses also emerge, such as deepfakes.

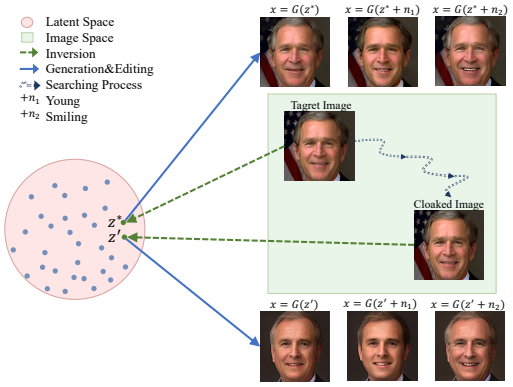


Figure 1: An illustration of GAN inversion and latent code manipulation, as well as a high-level overview of `UnGANable`.

One major example of deepfakes is face manipulation with GANs, which has been an emerging topic in very recent years [9, 10, 12, 14, 17, 24, 36, 38, 44, 45, 47, 49, 54, 56, 57, 63]. As face manipulation systems can change the target face with respect to certain attributes, such as hairstyle or facial expression, and considering that the manipulated results become increasingly more realistic, these techniques can easily be misused for malicious purposes, such as misinformation generation. In detail, the malicious manipulator may edit the portrait image of any person without his/her permission. Moreover, the manipulator is able to forge the expression (e.g. lip shape) of political figure's speech video, which might seriously mislead the public. Therefore, heavy concerns on such risks are raised, and we believe that individuals need tools to protect their facial images from being misused by malicious manipulators.

To leverage GANs to manipulate facial images, the manipulator/adversary needs to perform a two-step operation. The first step is *GAN inversion* [3, 4, 7, 52, 60, 61] which inverts a victim's facial image to a latent code. The second step is *latent code manipulation* [9, 14, 17, 24, 36, 44, 45, 54, 57, 63]

[1] See our code at https://github.com/zhenglisec/UnGANable.

which manipulates the latent code to get the modified image, such as adding a pair of glasses on the victim's face. See Figure 1 for an illustration of the two-step operation.

1.1 Our Contributions

In this paper, we propose the first defense system, namely UnGANable, against GANs-inversion-based face manipulation. In particular, UnGANable focuses on defending against GAN inversion. Once an image is successfully inverted to its accurate latent code, it is extremely hard (if not possible) to defend the following manipulation step as the adversary can perform any operation on the latent code. Therefore, we believe the most effective defense is to reduce the performance of GAN inversion - the adversary can only obtain an inaccurate latent code that is far from the accurate one, thus the following latent code manipulation step will not achieve the ideal result. See Figure 1 for an illustration of our defense.

UnGANable searches for cloaked images in the image space which are indistinguishable from the target images but can cause the adversary's GAN inversion to obtain an inaccurate latent code. In this way, any individual can use UnGANable to protect their images by sharing only the cloaked images online. Further, we focus on two state-of-the-art GAN inversion techniques, i.e., optimization-based inversion [3, 4] and hybrid inversion [52, 60, 61], and consider five scenarios to characterize the defender's background knowledge along multiple dimensions. By considering what knowledge the defender has, we obtain a taxonomy of five different types of methods (called "cloaks" throughout the paper) to disenable GAN inversion. More concretely, two cloaks are designed against optimization-based inversion, while the other three cloaks are designed against hybrid inversion.

We evaluate all our five cloaks on four popular GAN models that are constructed on two benchmark face datasets of different sizes and complexity. Extensive experiments show that UnGANable in general achieves remarkable performance with respect to both effectiveness and utility. We also conduct a comparison of our UnGANable with thirteen baseline image distortion methods. The results show that our defenses can outperform all these methods. We also explore four adaptive adversaries to bypass UnGANable and conduct sophisticated studies. Empirical results show that Spatial Smoothing [1] and more iterations of inversion are slightly effective.

In summary, we make the following contributions.

- We take the first step towards defending against malicious face manipulation by proposing UnGANable, a system that can jeopardize the process of GAN inversion.
- We consider five scenarios to comprehensively characterize a defender's background knowledge along multiple dimensions, and propose five different defenses for each scenario. Extensive evaluations on four popular GAN models show that UnGANable can achieve remarkable performance with respect to both effectiveness and utility.
- We conduct a comparison of our defenses with thirteen baseline image distortion methods. The results show that our defenses can outperform all these methods.
- We further explore four adaptive adversaries to bypass UnGANable and show that some of them are slightly effective.

2 Background and Related Work

In this section, we first introduce the two-step of GAN-based face manipulation, namely GAN inversion and latent code manipulation. Then we discuss other face manipulation techniques and existing defenses. For presentation purposes, we summarize the notation throughout the paper in Appendix Table 9. In particular, we emphasize that the adversary-controlled generator is marked as the target generator G_t and the adversary-controlled encoder is marked as the target encoder E_t.

2.1 GAN Inversion

In this paper, we consider two representative and most widely-used techniques of GAN inversion, i.e., optimization and hybrid formulations, as shown in Figure 2. The algorithms can be found in our technical report [32].[2]

Optimization-based Inversion. Existing optimization-based inversions [3, 4] typically reconstruct a target image by optimizing the latent vector

$$\mathbf{z}^* = \arg\min_{\mathbf{z}} \mathcal{L}_{\text{rec}}\big(\mathbf{x}, G_t(\mathbf{z})\big) \tag{1}$$

where $\mathbf{x}$ is the target image and G_t is the target generator. Starting from a Gaussian initialization $\mathbf{z}$, we search for an optimized vector $\mathbf{z}^*$ to minimize the reconstruction loss $\mathcal{L}_{\text{rec}}$ which measures the similarity between the given image $\mathbf{x}$ and the image generated from $\mathbf{z}^*$. $\mathcal{L}_{\text{rec}}$ is a weighted combination of the perceptual loss [25] and MSE loss:

$$\mathcal{L}_{\text{rec}} = \mathcal{L}_{\text{percept}}\big(G_t(\mathbf{z}), \mathbf{x}\big) + \mathcal{L}_{\text{mse}}\big(G_t(\mathbf{z}), \mathbf{x}\big)$$

where $\mathcal{L}_{\text{percept}}$ measures the similarity of features extracted from a pretrained neural network, such as VGG-16 [46], and $\mathcal{L}_{\text{mse}}$ measures the pixel-wise similarity.

Hybrid Inversion. An important issue for optimization-based inversion is initialization. Since Equation 1 is highly non-convex, the reconstruction quality strongly relies on a good initialization of $\mathbf{z}$. Consequently, researchers [48, 52, 61, 61]

[2] Due to space limitation, we defer most of the appendices to our technical report [32].

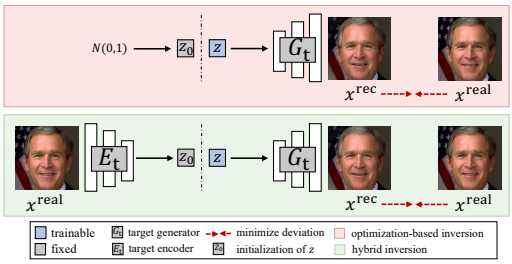


Figure 2: Illustration of GAN inversion methods. The upper is the optimization-based inversion. The bottom is the hybrid inversion.

propose to use an encoder to provide better initialization $\mathbf{z}$ for optimization, namely hybrid inversion.

Hybrid inversion first predicts $\mathbf{z}$ of a given image $\mathbf{x}$ by training a separate encoder, then uses the obtained $\mathbf{z}$ as the initialization for optimization. The learned predictive encoder serves as a fast bottom-up initialization for the non-convex optimization problem Equation 1.

2.2 Latent Code Manipulation

Considering that a given image has been successfully inverted into the latent space, the editing of the image can be easily executed. There are multiple methods [9, 14, 17, 24, 36, 44, 45, 54, 57, 63] to manipulate the latent code, most of them are based on algebraic operations on the latent code. For instance, in InterFaceGAN [44], the authors move the latent code $\mathbf{z}$ along a certain semantic direction n to edit the corresponding attribute of the image ($\mathbf{z}+n$). As the adversary has full control over the manipulation step, it is extremely difficult (if not possible) to defend this step. Therefore, we only focus on defending against the GAN inversion step - the adversary can only obtain a misleading latent code that is already far from its exact one. In this way, the latent code manipulation step will not achieve its ideal result.

2.3 Other Related Work

Image-Translation-Based Face Manipulation. This face manipulation [10, 23, 38, 47, 56, 58, 62], also known as Image-to-Image Translations (I2I), represented by StarGANv2 [11] and AttGAN [20], has received increasing attention in recent years. More concretely, I2I builds an end-to-end neural network as the backbone, to translate source images into the target domain with many aligned image pairs for training. When editing images, I2I uses the backbone network to accept the target image and output a new style of it without GAN-inversion process. Considering that the defense against I2I has been well studied [21, 31, 41, 55], the defense against GAN-inversion-based is still an open research problem. Our work is therefore well-motivated to complete this puzzle map. We also provide a more in-depth discussion of I2I in Section 8.

Existing Defenses Against Face Manipulation. As face manipulation causes a great threaten to individual privacy even political security, it is of paramount importance to develop countermeasures against it. To mitigate this risk, many defenses have been proposed, and these defenses can be broadly divided into two categories: detection [5, 30, 34, 35, 40, 59] and disrupting I2I [21, 31, 41, 55]. However, the former defense is designed in a passive manner to detect whether face images have been tampered with after wide propagation. The latter defense can only mitigate image-translation-based face manipulation by spoofing the backbone network. However, there is still no approach to defend against GAN-inversion-based face manipulation in a proactive manner. In this paper, we propose `UnGANable` of initiative defense to degrade the performance of GAN inversion, which is an essential step for subsequent face manipulation. See more discussion about limitations of existing defenses in our technical report [32].

3 Overview of `UnGANable`

In this section, we provide an overview of `UnGANable`.

3.1 Intuition

We derive the intuition behind our `UnGANable` from the basic pipeline of how inversion works. Since the optimization-based inversion is part of the hybrid inversion, here we focus only on the former. As described in Section 2.1, the inversion employs a loss function that is a weighted combination of the perceptual loss [25] and the pixel-wise MSE loss, to guide the optimization into the correct region of the latent space. This methodology leads to the following observations.

- The pixel-wise MSE loss works in the pixel space, i.e., the image space.
- The perceptual loss measures the similarity of features extracted from different images using a pretrained model, which works in the feature space.
- The optimization aims to search for the optimal latent code, which works in the latent space.

Thus, GAN inversion actually works in at least three spaces, i.e., the image space, the feature space, and the latent space. These observations motivate our `UnGANable`, which aims to maximize deviations in both latent and feature spaces with the cloaked images, meanwhile maintain the image indistinguishable in the image space.

3.2 Threat Model

The goal of the face manipulator (i.e., adversary) is to manipulate the face without any authorization from the owner of

Table 1: An overview of assumptions. "✓" means the defender needs the knowledge and "-" indicates the knowledge is not necessary. "Target" means the adversary-controlled entities, and "Shadow" means the defender-built entities locally.

Inversion Category	Cloaks	Target Generator	Shadow Generator	Target Encoder	Shadow Encoder	Feature Extractor	Inversion Technique
Optimization-based	White-box	✓	-	-	✓	✓	✓
	Black-box	-	-	-	-	✓	-
Hybrid	White-box	-	-	✓	-	✓	-
	Gray-box	-	✓	-	✓	✓	-
	Black-box	-	-	-	-	✓	-

the face image to serve its own purposes, such as violating individual privacy or even misleading political opinions, The face manipulator could be a commercial company or even an individual. We assume the face manipulator has access to advanced GANs (e.g., via GitHub), and can apply two advanced GAN inversion techniques, namely optimization-based inversion and hybrid inversion, to invert the images into the latent space. These two inversion methods are shown in Figure 2.

3.3 System Model

Any user (also called defender) can use `UnGANable` to search for cloaked images, which are around the target images in the image space. The design goals for these cloaks are:

- cloaked images should be indistinguishable from the target images;
- when inverting the cloaked image, the adversary can only get a misleading latent code, which is far from its accurate one in the latent space (see Equation 2).

Generally, `UnGANable` aims to maximize the deviations in the latent space and feature space, while keeping the images indistinguishable in image space. Therefore, the challenge for `UnGANable` is to obtain the representation in each space. To this end, we make different assumptions for `UnGANable` in different scenarios where `UnGANable` can use different methods to search for invisible images. The overview of background knowledge is introduced in Table 1

4 `UnGANable` Against Optimization-based Inversion

In this section, we present `UnGANable` against the first type of GAN inversion, i.e., optimization-based inversion.

4.1 Defender's Knowledge

For optimization-based inversion, we consider two different scenarios to characterize a defender's background knowledge. See more detailed explanation about background knowledge in our technical report [32].

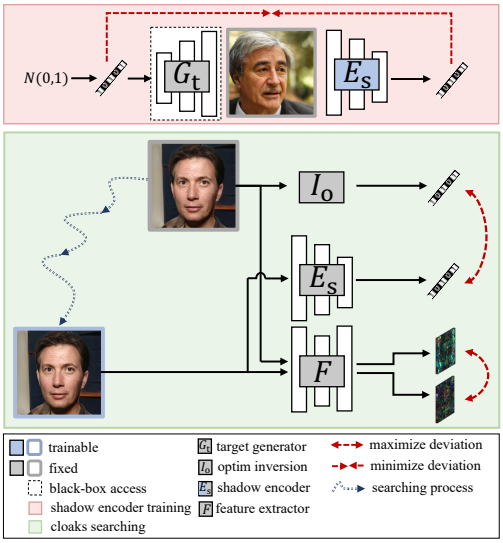


Figure 3: An illustration of white-box (Cloak v0) and black-box (Cloak v1) defenses against optimization-based inversion.

White-Box (Cloak v0). To maximize the deviation in the latent space, a defender has white-box access to the target generator G_t, and knows the adversary's inversion techniques I_o, thus he/she can obtain the accurate latent code of the original image. Besides, the defender trains a shadow encoder E_s to embed interim cloaked images to obtain the cloaked latent code. Then, the adversary can maximize the deviation between them. To maximize deviation in the feature space, we further assume that the defender has access to a feature extractor F, which can map both original image and cloaked image to feature space. Here, the feature extractor can be different from feature extractor used in perceptual loss.

Black-Box (Cloak v1). In this scenario, we assume the defender has no knowledge of the target generator or inversion techniques. Here, the defender only has access to a feature extractor F.

4.2 Methodologies

From a high-level overview, the defense can be divided into three simultaneous components, namely maximizing latent

deviation, maximizing feature deviation, and searching for cloaked images in the image space. The algorithms can be found in Appendix A.

White-Box (Cloak v0). The defender first leverages optimization-based inversion I_o to invert a target image $\mathbf{x}$ to obtain its exact latent code $I_o(\mathbf{x})$.[3] For maximizing latent deviation, the defender needs to build an end-to-end model, namely shadow encoder E_s, to invert the cloaked image $\hat{\mathbf{x}}$ of each step to obtain its latent code.[4] To train E_s, as shown in the pink part of Figure 3, the defender leverages the target generator G_t to create a dataset of generated images $G_t(\mathbf{z})$ and their latent codes $\mathbf{z}$, then minimize a similarity reconstruction loss $\mathcal{L}_{rec}$ between these latent codes $E_s\big(G_t(\mathbf{z})\big)$ and $\mathbf{z}$.

$$\mathcal{L}_{rec} = -\mathcal{L}_{cos}\Big(E_s\big(G_t(\mathbf{z})\big),\mathbf{z}\Big) + \mathcal{L}_{mse}\Big(E_s\big(G_t(\mathbf{z})\big),\mathbf{z}\Big) \quad (2)$$

where both $\mathcal{L}_{cos}$ and $\mathcal{L}_{mse}$ measure the element-wise similarity of latent codes. Here, $\mathcal{L}_{cos}$ is cosine similarity loss, and $\mathcal{L}_{mse}$ is MSE similarity loss.

For maximizing feature deviation, the defender uses a third-party pre-trained model (e.g., via GitHub) as the feature extractor F to obtain the features $F(\mathbf{x})$ and $F(\hat{\mathbf{x}})$. Once the defender obtains $I_o(\mathbf{x})$, E_s and F, the defender iteratively searches for $\hat{\mathbf{x}}$ in the image space by modifying $\mathbf{x}$, to maximize the latent and feature deviations between $\mathbf{x}$ and $\hat{\mathbf{x}}$.

$$\begin{gathered}\max_{\hat{\mathbf{x}}} \kappa\Big(\mathcal{L}_{rec}\big(E_s(\hat{\mathbf{x}}),I_o(\mathbf{x})\big)\Big) + (1-\kappa)\Big(\mathcal{L}_{rec}\big(F(\hat{\mathbf{x}}),F(\mathbf{x})\big)\Big) \\ \text{s.t. } |\hat{\mathbf{x}}-\mathbf{x}|_\infty < \varepsilon \\ \kappa \in [0,1]\end{gathered}$$

where $\mathcal{L}_{rec}(.)$ introduced in Equation 2 measures the element-wise similarity of two feature vectors or latent vectors, $|\hat{\mathbf{x}}-\mathbf{x}|_\infty$ measures the distance between $\hat{\mathbf{x}}$ and $\mathbf{x}$, ε is the distance budget in image space, and κ is a trade-off hyper-parameter between latent and feature spaces.

Black-Box (Cloak v1). The defender can only produce significant alterations to images' feature space, i.e., searching for $\hat{\mathbf{x}}$ in the image space by modifying $\mathbf{x}$, to maximize the feature deviation between $\hat{\mathbf{x}}$ and $\mathbf{x}$.

$$\begin{gathered}\max_{\hat{\mathbf{x}}} \mathcal{L}_{rec}(F(\hat{\mathbf{x}}),F(\mathbf{x})) \\ \text{s.t. } |\hat{\mathbf{x}}-\mathbf{x}|_\infty < \varepsilon\end{gathered}$$

4.3 Experimental Setup

GAN Models and Datasets. Without losing representativeness, we focus on four generative applications in recent years - DCGAN [39], WGAN [16], StyleGANv1 [27], and StyleGANv2 [28]. These GAN models are built with different architectures, losses and training schemes. Each generation application benchmarks its own dataset. As summarized in Table 2, we considered two benchmark datasets of different sizes and complexities, including CelebA [33] and FFHQ [27, 28], to construct different GAN models. Details of GAN models and datasets can be found in Appendix B.

Table 2: Target GANs, datasets and resolutions used to evaluate defense performance.

Model Zoo	Z dims	Dataset	Resolution
DCGAN (2016) [39]	100	CelebA [33]	64×64
WGAN (2017) [16]	128	CelebA [33]	128×128
StyleGANv1 (2019) [27]	512	FFHQ [27,28]	256×256
StyleGANv2 (2020) [28]	512	FFHQ [27,28]	256×256

Manipulator/Adversary. For face manipulator/adversary, we follow the original configurations of optimization-based inversion (Image2StyleGAN [3]). More specifically, we set up 500 iterations for the optimization step of inversion. In addition, we use perceptual loss and pixel-level MSE loss to reconstruct the target image in the optimization step. Though StyleGANv1 [27] and StyleGANv2 [28] also work on $\mathbf{w}$ space that is converted from $\mathbf{z}$ space, $\mathbf{z}$ space is applicable to all GAN models, thus we only consider $\mathbf{z}$ space in this work.

Defender. For the defender, we use a random initialized ResNet-18 [18] as the shadow encoder E_s in the white-box scenario (Cloak v0). Besides, for both white- and black-box scenarios (Cloak v0/v1), we adopt the easy-to-download, widely-used, and pre-trained ResNet-18 as the feature extractor. Further, we set up 500 iterations to iteratively search for the cloaked image in the image space by modifying the target image.

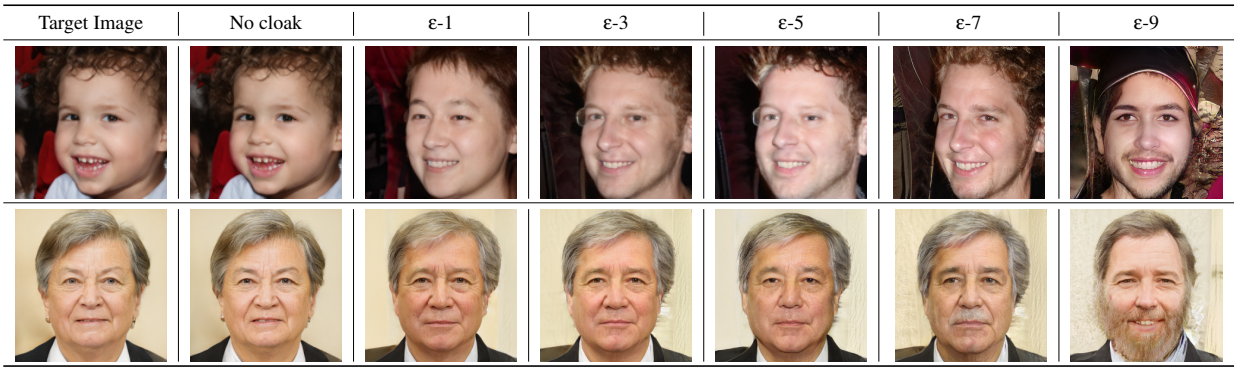
Target Samples. We first evaluate `UnGANable` on generated images from each GAN model. The reason is that, as stated in previous works [3, 4, 60], and also shown in our experimental results, the generated images are more easily inverted into accurate latent codes. In other words, in the competition between attackers and defenders, we actually make a very strong advantageous assumption for the former. We investigate whether `UnGANable` can achieve acceptable or even superior performance in such a worst-case scenario. Thus, for each GAN model, we evaluate the performance of `UnGANable` on 500 randomly selected generated images that can be successfully reconstructed.

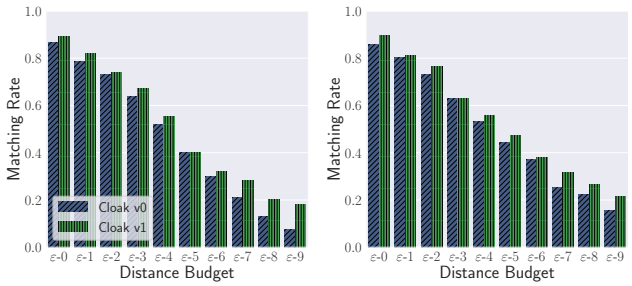
Evaluation Metrics. For evaluation metrics, we consider two perspectives: effectiveness and utility. Effectiveness measures the extent to which `UnGANable` jeopardizes the GAN inversion process. Given a target image, the sign of successful defense is a change in the identity of the reconstructed image, as shown in Figure 1. The reason is that once the identity of the reconstructed image changes, the defender no longer cares

[3] This process requires white-box access to the target generator G_t, as shown in Figure 2.

[4] The reason is that when iteratively searching in the image space, the defender needs to compute the cloaked image's gradient of each step with respect to the latent deviation by backpropagation, which is intractable through optimization-based inversion. The optimization-based inversion is just an inverted process, not an end-to-end model.

Table 3: Some visual examples of reconstructed images based on StyleGANv2. The defense method is Cloak v1.





(a) StyleGANv1 (b) StyleGANv2

Figure 4: The effectiveness performance of Cloak v0 and Cloak v1.

about the manipulation of the reconstructed image because the reconstructed image does not belong to the defender. To this end, we use *Matching Rate* to evaluate effectiveness:

$$\textit{Matching Rate} = \frac{\#\text{successful reconstructed images}}{\#\text{total images}}$$

Therefore, the lower the matching rate is, which means the more reconstructed images with changed identity, the better effectiveness `UnGANable` achieves. In our implementation, we utilize a popular open-source face verification/comparison tool FaceNet [42] to compute the defense success rate. Given the embedding distance of a pair of two face images, a pre-calibrated threshold is used to determine the classification of *same* and *different*, i.e., the two face images belong to the same person if the embedding distance is less than the threshold, otherwise different person. See more details on threshold selection in our technical report [32].

Utility measures whether the cloaked images searched by `UnGANable` is indistinguishable from the target images. To measure the utility, we use a variety of most widely-used similarity metrics, including mean squared error (MSE), structural similarity (SSIM) [51], and peak signal-to-noise ratio (PSNR). Here, the lower the MSE is, the higher the SSIM and PSNR are, then the better utility `UnGANable` achieves. More details about these metrics are presented in our technical report [32].

Table 4: The utility performance of `UnGANable` against optimization-based inversion.

Budget	Metric	Cloak v0	Cloak v1	Budget	Metric	Cloak v0	Cloak v1
ε-1	MSE	7.3e-05	7.2e-05	ε-7	MSE	0.0010	0.0014
	SSIM	0.9889	0.9891		SSIM	0.8802	0.8431
	PSNR	41.376	41.408		PSNR	30.118	28.532
ε-3	MSE	0.0003	0.0003	ε-9	MSE	0.0014	0.0022
	SSIM	0.9612	0.962		SSIM	0.8347	0.7820
	PSNR	35.684	35.716		PSNR	28.423	26.637
ε-5	MSE	0.0006	0.0006				
	SSIM	0.9228	0.9245				
	PSNR	32.419	32.455				

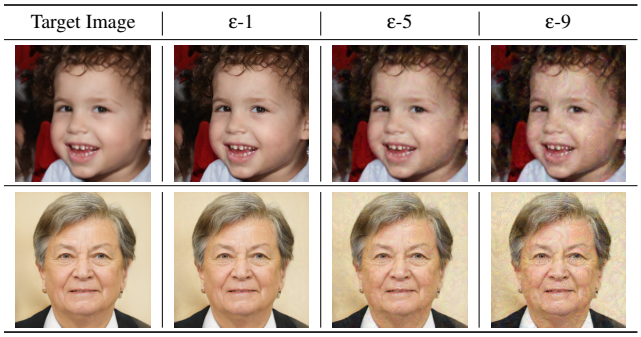
4.4 Results

Effectiveness Performance. In our `UnGANable`, we adopt a budget ε to limit distance between the cloaked and target image, aiming to ensure that the cloaked image is indistinguishable from the target image. Here, we first investigate the effectiveness of `UnGANable` by reporting matching rate under the effects of the distance budget ε. More concretely, we set 10 different distance budgets ε-0, ε-1, ... , ε-9 (uniformly ranging from 0.01 to 0.07 for DCGAN and WGAN, and from 0.01 to 0.1 for StyleGANv1 and StyleGANv2.[5]). Under each distance budgets, we perform grid search to find the optimum trade-off hyper-parameter κ. The exact settings for ε and κ can be found in our technical report [32].

Figure 4 depicts the effectiveness performance of Cloak v0 and Cloak v1 (see more results on DCGAN and WGAN in our technical report [32]). As we can see, with the increase of the budget ε, both Cloak v0 and Cloak v1 can significantly reduce matching rate. For example, in Figure 4 (Cloak v0,

[5]We conducted a pre-experiment and showed that only a small distance can jeopardize DCGAN and WGAN inversions, so we set the maximum magnitude of the distance budget to 0.07 for DCGAN and WGAN, and 0.1 for StyleGANv1/v2.

Table 5: Some visual examples of cloaked images searched by Cloak v1 performed on StyleGANv2 under different perturbation budgets.



StyleGANv2), the matching rate of ε-0 is 0.86, and that of ε-9 is 0.156, which drops sharply. These results imply that if we set a relatively high distance budget, UnGANable can achieve significant effectiveness against optimization-based inversion.

Besides above quantitative results, we further provide random qualitative examples to demonstrate the effectiveness of UnGANable performed on StyleGANv2. As shown in Table 3, we can observe that as ε increases, more and more facial attributes cannot be successfully reconstructed. The difference between the reconstructed image and the target image becomes more extensive, which implies the effectiveness is getting better.

Utility Performance. To evaluate the utility performance, we first quantitatively report a variety of similarity metrics (MSE/SSIM/PSNR) in Table 4. Typically, a SSIM value greater than 0.9 or a PSNR greater than 35 means a good quality of cloaked images. To elaborate more on utility performance, we show in Table 5 some qualitative samples of cloaked images searched by UnGANable performed on StyleGANv2. We can observe that when distance budget is set as ε-1 (0.02) and ε-3 (0.04), which represents a completely imperceptible perturbation, UnGANable can achieve acceptable effectiveness performance (see qualitative reconstructed examples in Table 3). In addition, we acknowledge that some perturbations are perceptible to our naked eye when the distance budget is set to ε-7 (0.08) or ε-9 (0.1). But note that these visual results are performed on the images generated by their corresponding GAN models. In the following Section 6, we further conduct experiments on real images. It is encouraging that UnGANable can apply a much lower distance budget to obtain excellent effectiveness performance while guaranteeing the visual quality of the cloaked image.

The Effect of Latent/Feature Deviation. We further investigate the effect of latent/feature deviation on the performance of UnGANable. In the white-box scenario (Cloak v0), UnGANable search for the cloaked images which can maximize both latent and feature deviations, while in the black-box scenario (Cloak v1) only feature deviations are maximized. As shown in Figure 4, we can obverse that Cloak v0 achieve better effectiveness performance than Cloak v1 under each distance budget. However, we cannot prematurely claim that Cloak v0 is better because we need to consider whether Cloak v0 is at least as good as Cloak v1 in terms of utility performance. Table 4 reports the utility performance of UnGANable on the StyleGANv2. First, we can observe that Cloak v0 performs at least on-par with Cloak v1 under budget ε-1, ε-3, and ε-5. More encouragingly, under budget ε-7 and ε-9, Cloak v0 achieves better utility performance than Cloak v1. These results show that Cloak v0 outperforms Cloak v1 in terms of both effectiveness and utility, and convincingly demonstrate that the additional latent deviation we introduce for Cloak v0 does improve performance.

Comparison with Baselines. To elaborate on UnGANable's performance in a more convincing manner, we compare UnGANable extensively with thirteen baseline distortion methods, as shown in Table 6. For each baseline method, we evaluate both effectiveness and utility performance with a wide variety of different magnitude of the budget. More detailed descriptions of each method are presented in our technical report [32]. Figure 5 displays the relationship between each baseline method's matching rate and MSE/SSIM/PSNR score (see more results in our technical report [32]). Thus, we can make the following observations.

First, as the budget increases (i.e., MSE becomes larger and SSIM/PSNR becomes smaller), all baseline methods can significantly reduce the matching rate, meaning that baseline methods that work only in image space can also achieve good effectiveness performance.

More encouragingly, the plot also clearly indicates the benefits of latent and feature deviations: among baseline methods with similar utility performance levels (similar MSE/SSIM/PSNR), our Cloak v0 and Cloak v1 consistently achieve better effectiveness (lower matching rate), as they benefit from maximizing latent and feature deviations. In other words, searching for cloaked images to maximize latent and feature deviations can further disenable GAN inversions at nearly no cost in utility. Another interesting finding is that when UnGANable is not an option, *GaussianNoise*, *GaussianBulr*, and *JPEGCompression* appear to perform better.

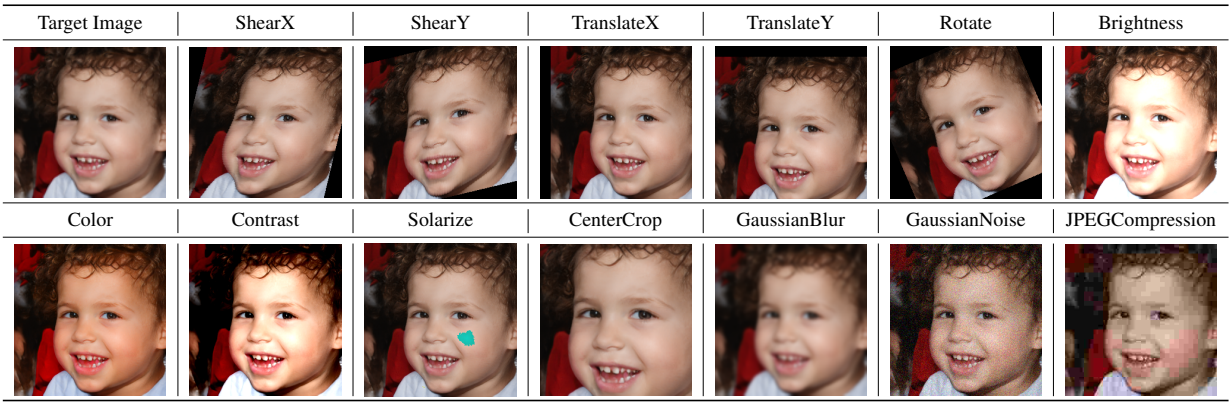
5 UnGANable Against Hybrid Inversion

We now present UnGANable against the second GAN inversion technique, i.e., hybrid inversion.

5.1 Defender's Knowledge

For hybrid inversion, we consider three different scenarios to characterize a defender's background knowledge. See more detailed explanation about background knowledge in our technical report [32].

Table 6: Visual examples of different baseline distortion methods.



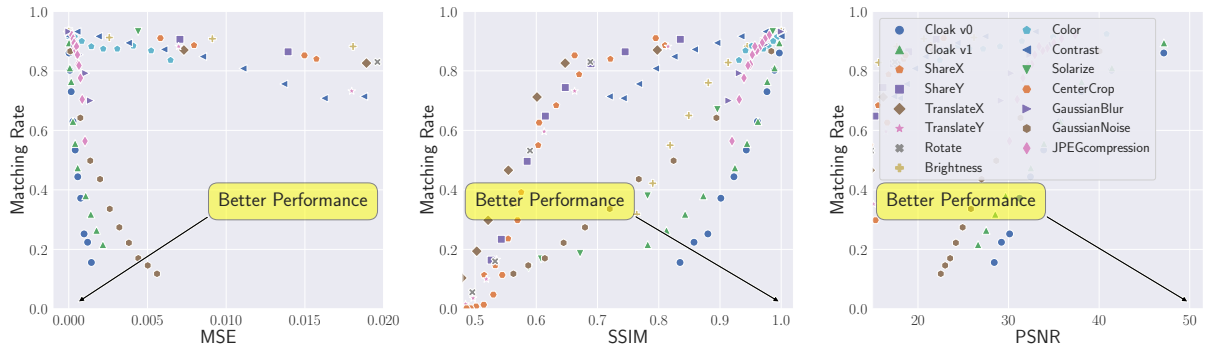


Figure 5: Comparison between all baseline methods and Cloak v0/v1 on generated images and StyleGANv2. The different points of each method represent different budgets.

White-Box (Cloak v2). Hybrid inversion actually adopts an encoder to provide a better initialization $\mathbf{z}$ for the following optimization step. Here, we assume that a defender has complete knowledge of the target encoder E_t to mislead the encoder, i.e., provide a worse initialization latent code $\mathbf{z}$ for the optimization. We give a quantitative illustration of this intuition in Section 5.2. Besides that, we also assume that the defender has access to a feature extractor F. Note that the defender does not need to have white-box access to the target generator G_t due to the design of this defense (see Section 5.2 more details).

Grey-Box (Cloak v3). Here, we relax the assumption that the defender has complete knowledge of the target encoder E_t. In particular, we assume that the defender can send many queries to the target encoder E_t and train a shadow encoder E_s to mimic the behavior of the target encoder E_t, and relies on the shadow encoder to act as the target encoder. Besides that, we assume that the defender has access to a feature extractor F for feature deviation.

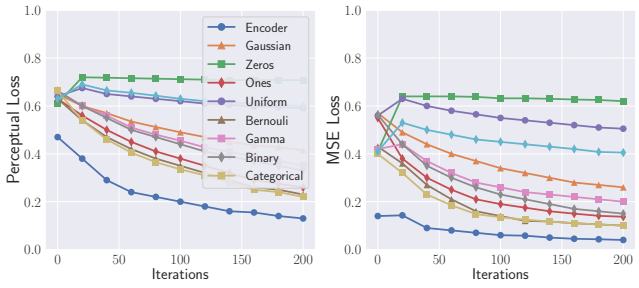
Black-Box (Cloak v4). Here, we assume the defender has no knowledge of the adversary's generator or encoder. Here, the defender only has access to a feature extractor F.

5.2 Methodologies

Here the defenses are also divided into three simultaneous components, namely maximizing latent deviation, maximizing feature deviation, and searching for cloaked images in the image space. In particular, we introduce a new novel method to maximize the latent deviation. The algorithms can be found in Appendix A.

New Perspective of Latent Deviation. As aforementioned in Section 2.1, an important issue for optimization-based inversion is initialization. Recent research [8,26,27,39] shows that using different initializations leads to a significant perceptual difference in generated images. Here, we conduct a pre-experiment on using different initializations to perform the optimization-based inversion, including Gaussian, zeros, etc (see [2] for each distribution). In particular, hybrid inversion adopts an encoder to provide initialization for optimization.

Figure 6 shows the trend of perceptual and MSE loss, respectively. First, the encoder indeed provides better initialization, which leads to better initial and final performance. Second, the trend of loss remains constant when the initialization is set to zero, which means it is quite difficult to invert the target image into the latent space. This observation sug-



(a) Perceptual Loss (b) MSE Loss

Figure 6: The loss trend under the effect of different initialization for optimization.

gests a new perspective on the latent deviation – misleading the encoder to provide zero initialization, or close to zero. In other words, our defense's goal against hybrid inversion should be to force the output of the encoder to zero. This is actually a special case of maximizing latent deviation, which provides the movement direction of the cloaked image in the latent space, i.e., towards zero.

White-Box (Cloak v2). In this scenario, we assume that the defender has full knowledge of the target encoder E_t, as well as an additional feature extractor F. As shown in the green part of Figure 7, the defender iteratively searches for $\hat{\mathbf{x}}$ in the image space by modify $\mathbf{x}$, in order to minimize the deviation between $E_t(\hat{\mathbf{x}})$ and zero, and maximize the deviation between $F(\hat{\mathbf{x}})$ and $F(\mathbf{x})$.

$$\max_{\hat{\mathbf{x}}} \kappa\Big(-\mathcal{L}_{\text{rec}}\big(E_t(\hat{\mathbf{x}}),0\big)\Big)+(1-\kappa)\Big(\mathcal{L}_{\text{rec}}\big(F(\hat{\mathbf{x}}),F(\mathbf{x})\big)\Big)$$
$$\text{s.t. } |\hat{\mathbf{x}}-\mathbf{x}|_\infty < \varepsilon$$
$$\kappa \in [0,1]$$

Grey-Box (Cloak v3). Here, we relax the assumption that the defender has complete knowledge of the target encoder E_t. The defender needs to build a shadow encoder E_s to match the predictions of E_t, i.e., find the shadow encoder's parameters that minimize the probability of errors between the shadow and target predictions.

As shown in the pink part of Figure 7, the defender builds a shadow generator G_s which is responsible for crafting some input images, and E_s serves as a discriminator while being trained to match target encoder's predictions on these images. In this setting, the two adversaries are E_s and G_s, which try to minimize and maximize the deviation between E_s and E_t respectively. Then, shadow encoder E_s becomes a functionally equivalent copy of target encoder E_t.

Finally, the defender iteratively searches for $\hat{\mathbf{x}}$ in the image space by modify $\mathbf{x}$, in order to minimize the deviation between $E_s(\hat{\mathbf{x}})$ and zero, and maximize the deviation between $F(\hat{\mathbf{x}})$ and $F(\mathbf{x})$.

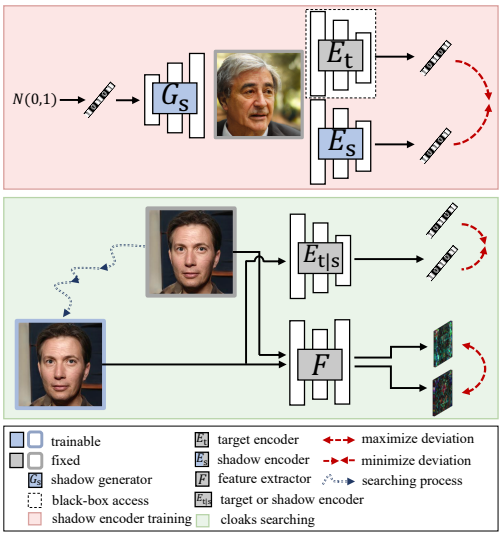


Figure 7: An illustration of white-box (Cloak v2), grey-box (Cloak v3) and black-box (Cloak v4).

$$\max_{\hat{\mathbf{x}}} \kappa\Big(-\mathcal{L}_{\text{rec}}\big(E_s(\hat{\mathbf{x}}),0\big)\Big)+(1-\kappa)\Big(\mathcal{L}_{\text{rec}}\big(F(\hat{\mathbf{x}}),F(\mathbf{x})\big)\Big)$$
$$\text{s.t. } |\hat{\mathbf{x}}-\mathbf{x}|_\infty < \varepsilon$$
$$\kappa \in [0,1]$$

Black-Box (Cloak v4). In this scenario, the defender has no knowledge of the target generator or target encoder or inversion techniques. The defender can only search for $\hat{\mathbf{x}}$ in the image space by modifying $\mathbf{x}$, to maximize the feature deviation between $\hat{\mathbf{x}}$ and $\mathbf{x}$.

$$\max_{\hat{\mathbf{x}}} \mathcal{L}_{\text{rec}}(F(\hat{\mathbf{x}}),F(\mathbf{x}))$$
$$\text{s.t. } |\hat{\mathbf{x}}-\mathbf{x}|_\infty < \varepsilon$$

5.3 Experimental Setup

For the manipulator/adversary, we follow the configurations of hybrid inversion (Zhu et al. [60]). Here, we again only consider the $\mathbf{z}$ space for all GAN models. We set up 100 iterations for the optimization step of inversion, and use perceptual loss and pixel-level MSE loss to reconstruct the target image in the optimization step.

As a defender, for Cloak v3, we build the shadow generator by using 1 linear layer to accept Gaussian noise, followed by five convolutional layers and five Batch Normalization [22] layers. Furthermore, we again use a random initialized ResNet-18 as the shadow encoder. For all Cloaks (v2/v3/v4), we again use a pretrained ResNet-18 [18] as the feature extractor. Besides, we fix the number of iterations as 500, to search for cloaked images. In addition, all other experimental settings are the same as described in Section 4.3.

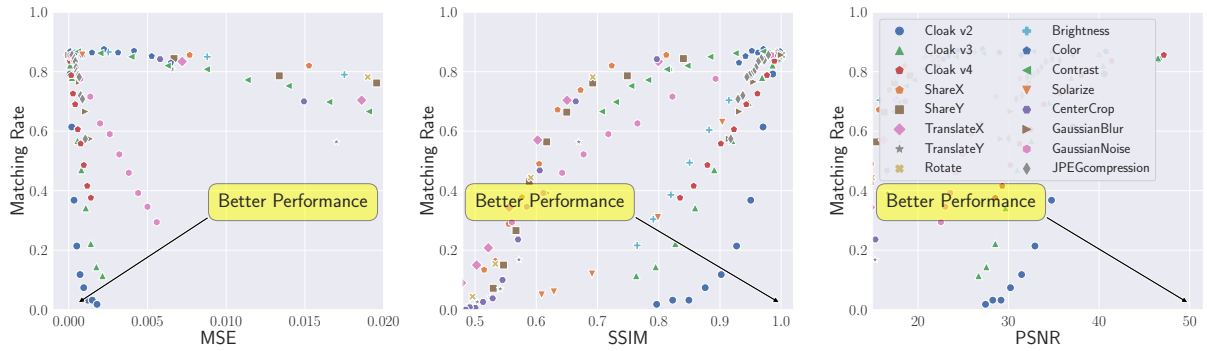
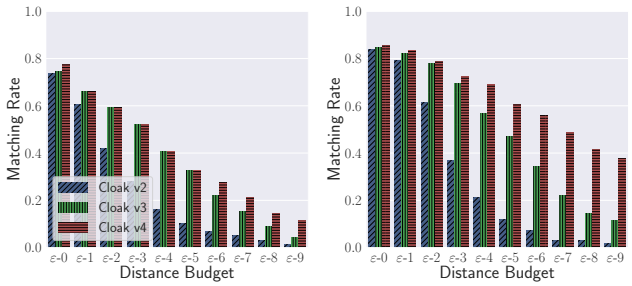


Figure 8: Comparison between all baseline methods and Cloak v2/v3/v4 on generated images and StyleGANv2. The different points of each method represent different budgets.



(a) StyleGANv1 (b) StyleGANv2

Figure 9: The effectiveness performance of Cloak v2, Cloak v3 and Cloak v4.

5.4 Results

Effectiveness Performance. To evaluate the effectiveness performance quantitatively, we use the same evaluation setup as presented in Section 4.4. Figure 9 depicts the effectiveness performance of Cloak v2/v3/v4 (See more results on DCGAN and WGAN in our technical report [32]). First, we again observe that as the budget increases, all Cloak v2/v3/v4 can significantly reduce the matching rate. These results indeed imply that UnGANable can achieve significant effectiveness against hybrid inversion. For qualitative results, the same perturbation budget will lead to similar reconstructed results, as shown in Table 3.

Utility Performance. Similarly, since we set the same distance budgets as adopted against optimization-based inversion, thus for the same perturbation budget will lead to similar quantitative and qualitative utility performance, as shown in Table 3 and Table 4.

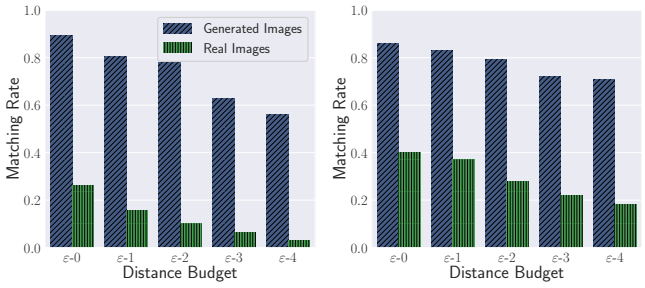
The Effect of Latent/Feature Deviation. In Figure 9a and Figure 9b, we can observe that searching for cloaked images to mislead the target encoder controlled by adversary (Cloak v2) leads to much better effectiveness performance. Furthermore, the larger the distance budget, the larger the gap between Cloak v2 and both Cloak v3 and Cloak v4, reflecting the fact that zero initialization can significantly jeopardize the process of GAN inversion. This convincingly verifies our new perspective of latent deviation–misleading the adversary's encoder to provide zero initialization, or close to zero.

Comparison with Baselines. We compare `UnGANable` extensively with thirteen baseline methods, as shown in Table 6. We use the same experimental setup as described in Section 4.4, such as the perturbation budget setting strategy and the result reporting metrics. We report comparisons between baseline methods and `UnGANable` in Figure 8, and we can make the similar observations as mentioned in Section 4.4. See more results on DCGAN/WGAN/StyleGANv1 in our technical report [32]. Here, we again emphasize that Cloak v2/v3/v4 achieves consistently better effectiveness (lower matching rate) and utility (lower MSE, higher SSIM and PSNR) performance than all baselines.

6 Evaluation on Real Images

To elaborate on `UnGANable`'s performance, here we investigate the performance of `UnGANable` on real facial images. Concretely, we consider the strictest setting in which the defender has no knowledge of the adversary-controlled entities. Thus, we only consider the black-box scenario against optimization-based and hybrid inversion, i.e., Cloak v1 and Cloak v4. In addition, the adversary-controlled GAN model is the state-of-the-art deepfake generative model StyleGANv2. We collect 200 real images from the FFHQ dataset, and these images are the most successfully inverted into the latent space among the whole FFHQ dataset.

Effectiveness Performance. We first present the effectiveness performance of `UnGANable`. We use the same evaluation setup as presented in Section 4.4. We set 5 different distance budgets ε-0/1/2/3/4, the same as adopted in previous evaluations. Figure 10 depicts the effectiveness performance of Cloak v1 and Cloak v4. First, we again observe that as the



(a) Optimization-based (b) Hybrid

Figure 10: The effectiveness performance of Cloak v1/v4 on generated and real images, respectively.

budget ε increases, both Cloak v1 and Cloak v4 can significantly reduce the matching rate. Then we can see that the matching rate of Cloak v4 is clearly higher than that of Cloak v1, which verifies that the encoder of hybrid inversion indeed leads to better reconstruction performance.

What is more encouraging is that `UnGANable` can achieve better effectiveness performance compared to that on generated images. For example, when the distance budget is set as ε-4 (0.05), the matching rate of Cloak v1/v4 on the real image is about 0.072/0.191, while that on the generated image is about 0.474/0.606. The results clearly show that `UnGANable` can apply a much lower perturbation budget to obtain better effectiveness performance, and this lower distance budget further benefits utility performance.

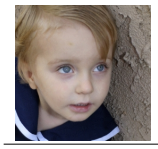
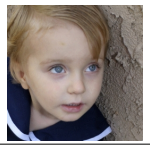
Utility Performance. For utility performance, we conduct the evaluations both quantitatively and qualitatively. we first quantitatively report a variety of similarity metrics (MSE/SSIM/PSNR) in Table 7. Generally, SSIM values of 0.97, 0.98, and 0.99 imply the excellent visual quality of the cloaked images. We then show in Table 8 some qualitative samples of cloaked images. We can observe that when the distance budget is set as ε-4 (0.05), which represents a completely imperceptible perturbation, `UnGANable` can achieve remarkable effectiveness performance (see matching rate in Figure 10b). Therefore, we claim that `UnGANable` provides acceptable protection for real images by much lower distance budgets and still yields good effectiveness and utility performance.

Comparison with Baselines. We then compare `UnGANable` extensively with thirteen baseline distortion methods, as shown in Table 6. For each baseline method, we evaluate both effectiveness and utility performance with a wide variety of different magnitude of the budget. Figure 12 displays the compassion between baseline methods and Cloak v1/v4, respectively (see more results of MSE/SSIM in our technical report [32]). Thus, we can make the same observations as `UnGANable` on generated images, i.e., our Cloak v1/v4 of `UnGANable` achieves consistently better effectiveness (lower matching rate) and utility (lower MSE, higher SSIM, and PSNR) performance compared to all baseline methods.

Table 7: The quantitative utility performance of `UnGANable` under Cloak v1 and v4 settings.

Budget	Metric	Cloak v1	Cloak v4	Budget	Metric	Cloak v1	Cloak v4
ε-0	MSE	1.9e-05	1.9e-05	ε-3	MSE	0.0003	0.0002
	SSIM	0.9968	0.9969		SSIM	0.9606	0.967
	PSNR	47.205	47.210		PSNR	35.783	35.783
ε-1	MSE	7.1e-05	7.2e-05	ε-4	MSE	0.0004	0.0004
	SSIM	0.9887	0.9887		SSIM	0.9422	0.9423
	PSNR	41.473	41.473		PSNR	33.983	33.982
ε-2	MSE	0.0002	0.0002				
	SSIM	0.9764	0.9764				
	PSNR	38.144	38.145				

Table 8: Some visual examples of cloaked real images searched by Cloak v4 performed on StyleGANv2.

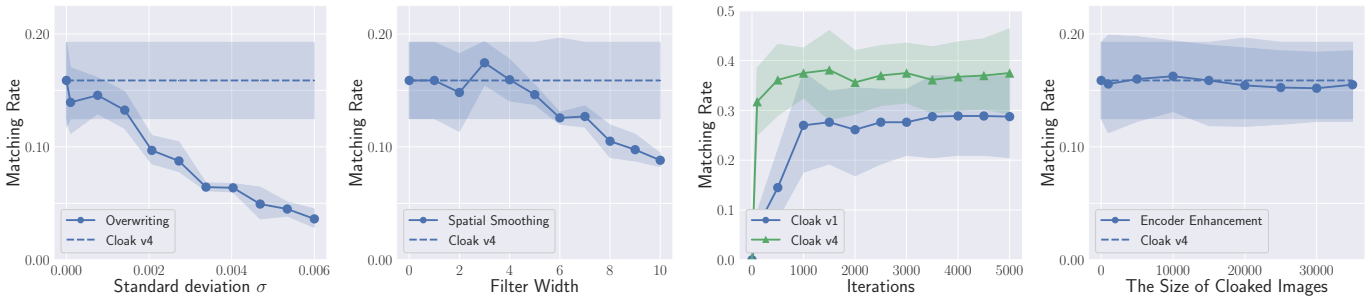
Target Image	ε-0	ε-2	ε-4
			
			

7 Possible Adaptive Adversary

Here, we explore four possible adaptive adversaries and empirically evaluate the performance of `UnGANable` on real facial images. We conduct extensive experiments under the black-box scenario against optimization-based and hybrid inversion, i.e., Cloak v1 and Cloak v4. Note that for the purpose of straightforward comparisons, we average the performance of `UnGANable` with a varying number of distance budgets, i.e., ε-0/1/2/3.

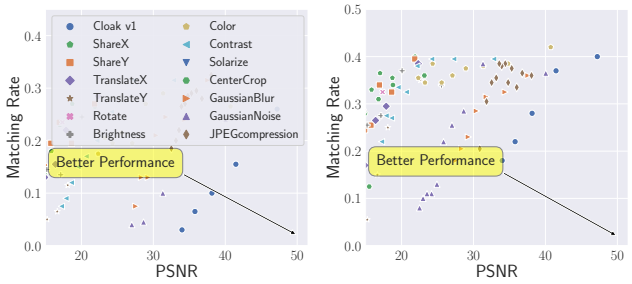
Cloak Overwriting. This adaptive adversary aims to disturb the cloaks, i.e., the imperceptible perturbation searched by `UnGANable`. The adversary samples random noise from a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ to overwrite the cloaks.

We report the matching rate by varying the standard deviation σ (set μ as 0 for simplicity) in Figure 11a (see more results of Cloak v1 in our technical report [32]). We can observe that as the standard deviation increases, the matching rate of cloak overwriting is significantly reduced. The reason is that the cloak overwriting actually introduces more noise in the image space on top of the imperceptible noise searched by the `UnGANable`, which further jeopardizes the GAN inversion process. These results indicate that cloak overwriting is not



(a) Cloak Overwriting (b) Cloak Purification (c) More Iterations (d) Encoder Enhancement

Figure 11: The effectiveness performance of UnGANable on real images under the effect of four possible adaptive adversaries.



(a) Optimization-based (b) Hybrid

Figure 12: Comparison between all baseline methods and Cloak v1/v4 on real images. The different points of each method represent different budgets.

an applicable adaptive strategy for adversaries.

Cloak Purification. This adaptive adversary aims to remove or purify the cloaks searched by UnGANable. As aforementioned, these cloaks actually are the imperceptible noise added to the images. Thus, we consider one of the most wide-used and easy-to-apply image noise reduction mechanisms, i.e., Spatial Smoothing [1]. Spatial Smoothing means that pixel values are averaged with their neighboring pixel values with a low-pass filter, leading to the sharp "edges" of the image becoming blurred and the spatial correlation within the data becoming more apparent.

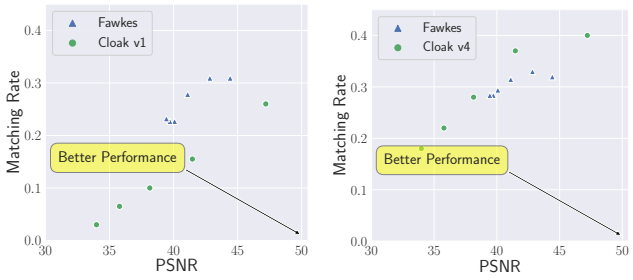
We report the matching rate by varying the filter widths of Spatial Spatial in Figure 11b (see more results of Cloak v1 in our technical report [32]). We can clearly observe that the matching rate increases at first and then decreases. These results indicate that Spatial Smoothing indeed can purify the imperceptible noise added by UnGANable to some extent. We should also note that even the optimal setting for Spatial Smoothing can only lead to a slightly increased matching rate, and they all drop further sharply when the filter width is very large, as the Spatial Smoothing destroys the pixel space of the original image. This observation implies that Spatial Smoothing is only a slightly effective adaptive strategy to reduce the jeopardy of UnGANable to GAN inversions.

More Iterations of Inversion. This adaptive adversary has significant computational resources to perform a huge number of optimization iterations to increase the matching rate. More specifically, we vary the number of optimization iterations from 0 to 5000 for both optimization-based and hybrid inversions. Note that the default settings for the number of iterations are 500 and 100 for optimization-based inversion and hybrid inversion, respectively.

Figure 11c shows the matching rate of UnGANable under the effect of numbers of iterations. As expected, we can find that the matching rate increases with the number of optimization iterations. Specifically, the matching rate increases sharply up to 1000/100 iterations and continues to increase slowly afterward. These results clearly demonstrate that more iterations of inversion indeed can reduce the jeopardy of UnGANable to GAN inversions. We should also note that a larger number of iterations (even up to 5000) does not lead to great effects, but is a huge cost in terms of resource usage.

Encoder Enhancement. We further consider another adaptive adversary where the adversary retrains the encoder to be more robust to imperceptible noise searched by UnGANable. More concretely, we assume that the adversary can collect a large number of cloaked images from crawler-accessible websites or social media. We consider various numbers of cloaked images from 5k to 35k that an adversary can collect. Note that the number of images in the full FFHQ dataset used to train StyleGANv2 is only 70k. Then the adversary retrains the encoder by a mixed set of original clean images and collected cloaked images.

Since the encoder is only employed for hybrid inversion, we only consider here Cloak v4, the black-box setting against hybrid inversion, for evaluation. Figure 11d reports the matching rate under the effect of the different numbers of cloaked images collected by the adversary. We can observe that the matching rate decreases slightly with increasing cloaked images, which means that retraining the encoder increases the jeopardy of UnGANable to GAN inversion. In a nutshell, en-



(a) Optimization-based (b) Hybrid

Figure 13: Comparison between Fawkes and Cloak v1/v4 on real images. The different points of each method represent different budgets.

coder enhancement is not an applicable adaptive strategy for adversaries to reduce the jeopardy of UnGANable to GAN inversions.

8 Discussion

Comparison with Fawkes. Recently, the countermeasures which aim to protect faces from being stolen by recognition systems have been studied. Fawkes [43], one of the representative works, adds pixel-level perturbations to users' photos by altering the feature space before uploading them to the Internet. The functionality of unauthorized facial recognition models trained on these photos with perturbations will be deteriorated seriously.

For a convincing evaluation, we leverage the original implementation of Fawkes to protect the same real facial images as used in above evaluation.[6] We set multiple different perturbation budgets to perturb these real images and evaluate the performance of Fawkes against both optimization-based and hybrid inversions. Figure 13 displays the comparison between Fawkes and Cloak v1/v4. First, we can observe that Fawkes indeed can jeopardize the process of GAN inversions. Further, we can also see that Fawkes provides worse protection against optimization-based inversion, and similar or slightly better protection against hybrid inversion, compared to UnGANable.

Here, we emphasize that except for the special black-box settings, we also propose white-box and gray-box settings, i.e., Cloak v0/v2/3. The extensive evaluation in Figure 4 and Figure 9 shows that Cloak v0/v2/3 actually achieves better performance than Cloak v1/v4, especially in Hybrid inversion, which is naturally better than Fawkes. That is, UnGANable performs better than Fawkes in most cases. More importantly, we should note that the goals of Fawkes and UnGANable are totally different: Fawkes aims to mislead the face recognition classifiers while UnGANable misleads the GAN inversion to prevent malicious face manipulation.

[6] https://github.com/Shawn-Shan/fawkes

Limitation. There are two major paradigms for image manipulation: GAN-inversion-based and image-translation-based. The latter, represented by StarGANv2 [11] and AttGAN [20], transforms an image from the source domain to the target domain without the GAN-inversion process. Therefore, our proposed UnGANable is not applicable to image-translation-based manipulation, as the key idea of UnGANable is to jeopardize the process of GAN inversion. Moreover, we emphasize here that GAN-inversion-based and image-translation-based are two orthogonal image manipulation techniques. Considering that the defense against the latter has been well studied [21, 31, 41, 55], the defense against GAN-inversion-based is still an open research problem. Our work is therefore well-motivated to complete this puzzle map.

Moreover, except for $\mathbf{z}$ space we consider in this work, recent works [3, 4, 6, 13, 28, 37, 48, 50, 60] also works on $\mathbf{w}$ space, which is transformed from $\mathbf{z}$ space, leading to a better inversion performance. We leave the in-depth exploration of more efficient UnGANable against $\mathbf{w}$ space for future work.

9 Conclusion

In this paper, we take the first step towards defending against GAN-inversion-based face manipulation by proposing UnGANable, a system that can jeopardize the process of GAN inversion. We consider two advanced GAN inversions: optimization-based and hybrid inversions, as well as five scenarios to comprehensively characterize the defender's background knowledge in multiple dimensions. We extensively evaluate UnGANable on four popular GAN models built on two benchmark face datasets of different sizes and complexity. The results show that UnGANable can achieve remarkable performance with respect to both effectiveness and utility. We further conduct a comparison of UnGANable with thirteen image distortion methods as well as Fawkes, and the results show that UnGANable generally outperforms all these methods. In addition, we explore four possible adaptive adversaries against UnGANable, and empirical evaluation shows that Spatial Smoothing and more iterations of inversion are slightly effective.

Acknowledgements

We thank all anonymous reviewers for their constructive comments. This work is partially funded by the Helmholtz Association within the project "Trustworthy Federated Data Analytics" (TFDA) (funding number ZT-I-OO1 4).

References

[1] https://support.brainvoyager.com/brainvoyager/functional-analysis-preparation/29-pre-processing/86-spatial-smoothing.

[2] https://github.com/graykode/distribution-is-all-you-need.

[3] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2StyleGAN: How to Embed Images Into the StyleGAN Latent Space? In *IEEE International Conference on Computer Vision (ICCV)*, pages 4431–4440. IEEE, 2019.

[4] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2StyleGAN++: How to Edit the Embedded Images? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8293–8302. IEEE, 2020.

[5] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. MesoNet: a Compact Facial Video Forgery Detection Network. In *IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7. IEEE, 2018.

[6] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit H. Bermano. HyperStyle: StyleGAN Inversion with HyperNetworks for Real Image Editing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18511–18521. IEEE, 2022.

[7] Christian Bartz, Joseph Bethge, Haojin Yang, and Christoph Meinel. One Model to Reconstruct Them All: A Novel Way to Use the Stochastic Noise in StyleGAN. *CoRR abs/2010.11113*, 2020.

[8] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large Scale GAN Training for High Fidelity Natural Image Synthesis. In *International Conference on Learning Representations (ICLR)*, 2019.

[9] Anton Cherepkov, Andrey Voynov, and Artem Babenko. Navigating the GAN Parameter Space for Semantic Image Editing. *CoRR abs/2011.13786*, 2020.

[10] Yunjey Choi, Min-Je Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8789–8797. IEEE, 2018.

[11] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. StarGAN v2: Diverse Image Synthesis for Multiple Domains. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8185–8194. IEEE, 2020.

[12] Emily Denton, Ben Hutchinson, Margaret Mitchell, and Timnit Gebru. Detecting Bias with Generative Counterfactual Face Attribute Augmentation. *CoRR abs/1906.06439*, 2019.

[13] Tan M. Dinh, Anh Tuan Tran, Rang Nguyen, and Binh-Son Hua. HyperInverter: Improving StyleGAN Inversion via Hypernetwork. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11389–11398. IEEE, 2022.

[14] Lore Goetschalckx, Alex Andonian, Aude Oliva, and Phillip Isola. GANalyze: Toward Visual Definitions of Cognitive Image Properties. In *IEEE International Conference on Computer Vision (ICCV)*, pages 5743–5752. IEEE, 2019.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 2672–2680. NIPS, 2014.

[16] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved Training of Wasserstein GANs. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 5767–5777. NIPS, 2017.

[17] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. GANSpace: Discovering Interpretable GAN Controls. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*. NeurIPS, 2020.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. IEEE, 2016.

[19] Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. Fast Matrix Factorization for Online Recommendation with Implicit Feedback. In *International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 549–558. ACM, 2016.

[20] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. AttGAN: Facial Attribute Editing by Only Changing What You Want. *IEEE Transactions on Image Process*, 2019.

[21] Qidong Huang, Jie Zhang, Wenbo Zhou, Weiming Zhang, and Nenghai Yu. Initiative Defense against Facial Manipulation. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 1619–1627. AAAI, 2021.

[22] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning (ICML)*, pages 448–456. PMLR, 2015.

[23] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976. IEEE, 2017.

[24] Ali Jahanian, Lucy Chai, and Phillip Isola. On the "Steerability" of Generative Adversarial Networks. In *International Conference on Learning Representations (ICLR)*, 2020.

[25] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In *European Conference on Computer Vision (ECCV)*, pages 694–711. Springer, 2016.

[26] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive Growing of GANs for Improved Quality, Stability, and Variation. In *International Conference on Learning Representations (ICLR)*, 2018.

[27] Tero Karras, Samuli Laine, and Timo Aila. A Style-Based Generator Architecture for Generative Adversarial Networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4401–4410. IEEE, 2019.

[28] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and Improving the Image Quality of StyleGAN. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8107–8116. IEEE, 2020.

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 1106–1114. NIPS, 2012.

[30] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face X-Ray for More General Face Forgery Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5000–5009. IEEE, 2020.

[31] Yuezun Li, Xin Yang, Baoyuan Wu, and Siwei Lyu. Hiding Faces in Plain Sight: Disrupting AI Face Synthesis with Adversarial Perturbations. *CoRR abs/1906.09288*, 2019.

[32] Zheng Li, Ning Yu, Ahmed Salem, Michael Backes, Mario Fritz, and Yang Zhang. UnGANable: Defending Against GAN-based Face Manipulation. *CoRR abs/2210.00957*, 2022.

[33] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep Learning Face Attributes in the Wild. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3730–3738. IEEE, 2015.

[34] Falko Matern, Christian Riess, and Marc Stamminger. Exploiting Visual Artifacts to Expose Deepfakes and Face Manipulations. In *Winter Conference on Applications of Computer Vision (WACV)*, pages 83–92. IEEE, 2019.

[35] Huy H. Nguyen, Junichi Yamagishi, and Isao Echizen. Use of a Capsule Network to Detect Fake Images and Videos. *CoRR abs/1910.12467*, 2019.

[36] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. StyleCLIP: Text-Driven Manipulation of StyleGAN Imagery. *CoRR abs/2103.17249*, 2021.

[37] John Pavlopoulos, Leo Laugier, Alexandros Xenos, Jeffrey Sorensen, and Ion Androutsopoulos. From the Detection of Toxic Spans in Online Discussions to the Analysis of Toxic-to-Civil Transfer. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3721–3734. ACL, 2022.

[38] Albert Pumarola, Antonio Agudo, Aleix M. Martínez, Alberto Sanfeliu, and Francesc Moreno-Noguer. GANimation: One-Shot Anatomically Consistent Facial Animation. In *International Joint Conferences on Artifical Intelligence (IJCAI)*, pages 698–713. IJCAI, 2020.

[39] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. In *International Conference on Learning Representations (ICLR)*, 2016.

[40] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics++: Learning to Detect Manipulated Facial Images. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1–11. IEEE, 2019.

[41] Nataniel Ruiz, Sarah Adel Bargal, and Stan Sclaroff. Disrupting Deepfakes: Adversarial Attacks Against Conditional Image Translation Networks and Facial Manipulation Systems. In *European Conference on Computer Vision (ECCV)*, pages 236–251. Springer, 2020.

[42] Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A Unified Embedding for Face Recognition and Clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823. IEEE, 2015.

[43] Shawn Shan, Emily Wenger, Jiayun Zhang, Huiying Li, Haitao Zheng, and Ben Y. Zhao. Fawkes: Protecting Privacy against Unauthorized Deep Learning Models. In *USENIX Security Symposium (USENIX Security)*, pages 1589–1604. USENIX, 2020.

[44] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the Latent Space of GANs for Semantic Face Editing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9240–9249. IEEE, 2020.

[45] Yujun Shen and Bolei Zhou. Closed-Form Factorization of Latent Semantics in GANs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1532–1540. IEEE, 2021.

[46] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations (ICLR)*, 2015.

[47] Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2Face: Real-Time Face Capture and Reenactment of RGB Videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2387–2395. IEEE, 2016.

[48] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for StyleGAN image manipulation. *ACM Transactions on Graphics*, 2021.

[49] Binxu Wang and Carlos R. Ponce. A Geometric Analysis of Deep Generative Image Models and Its Applications. In *International Conference on Learning Representations (ICLR)*, 2021.

[50] Tengfei Wang, Yong Zhang, Yanbo Fan, Jue Wang, and Qifeng Chen. High-Fidelity GAN Inversion for Image Attribute Editing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11379–11388. IEEE, 2022.

[51] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image Quality Assessment: from Error Visibility to Structural Similarity. *IEEE Transactions on Image Process*, 2004.

[52] Tianyi Wei, Dongdong Chen, Wenbo Zhou, Jing Liao, Weiming Zhang, Lu Yuan, Gang Hua, and Nenghai Yu. A Simple Baseline for StyleGAN Inversion. *CoRR abs/2104.07661*, 2021.

[53] Le Wu, Peijie Sun, Yanjie Fu, Richang Hong, Xiting Wang, and Meng Wang. A Neural Influence Diffusion Model for Social Recommendation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 235–244. ACM, 2019.

[54] Zhiliang Xu, Xiyu Yu, Zhibin Hong, Zhen Zhu, Junyu Han, Jingtuo Liu, Errui Ding, and Xiang Bai. FaceController: Controllable Attribute Editing for Face in the Wild. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 3083–3091. AAAI, 2021.

[55] Chin-Yuan Yeh, Hsi-Wen Chen, Shang-Lun Tsai, and Shang-De Wang. Disrupting Image-Translation-Based DeepFake Algorithms with Adversarial Attacks. In *Winter Conference on Applications of Computer Vision (WACV)*, pages 53–62. IEEE, 2020.

[56] Zili Yi, Hao (Richard) Zhang, Ping Tan, and Minglun Gong. DualGAN: Unsupervised Dual Learning for Image-to-Image Translation. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2868–2876. EEE, 2017.

[57] Oguz Kaan Yüksel, Enis Simsar, Ezgi Gülperi Er, and Pinar Yanardag. LatentCLR: A Contrastive Learning Approach for Unsupervised Discovery of Interpretable Directions. *CoRR abs/2104.00820*, 2021.

[58] Pan Zhang, Bo Zhang, Dong Chen, Lu Yuan, and Fang Wen. Cross-Domain Correspondence Learning for Exemplar-Based Image Translation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5142–5152. IEEE, 2020.

[59] Peng Zhou, Xintong Han, Vlad I. Morariu, and Larry S. Davis. Two-Stream Neural Networks for Tampered Face Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1831–1839. IEEE, 2017.

[60] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-Domain GAN Inversion for Real Image Editing. In *European Conference on Computer Vision (ECCV)*, pages 592–608. Springer, 2020.

[61] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A. Efros. Generative Visual Manipulation on the Natural Image Manifold. In *European Conference on Computer Vision (ECCV)*, pages 597–613. Springer, 2016.

[62] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251. IEEE, 2017.

[63] Peiye Zhuang, Oluwasanmi Koyejo, and Alexander G. Schwing. Enjoy Your Editing: Controllable GANs for Image Editing via Latent Space Navigation. In *International Conference on Learning Representations (ICLR)*, 2021.

A Algorithms of `UnGANable`

Algorithm 1 is for Cloak-0. Algorithm 2 is for Cloak-1. Algorithm 3 is for Cloak-2. Algorithm 4 is for Cloak-3. Algorithm 2 is for Cloak-4.

Table 9: List of notations.

Notation	Description
$\mathbf{z}$	Latent code
$\mathbf{x}$	Target image (uncloaked)
$\hat{\mathbf{x}}$	Cloaked version of the target image $\mathbf{x}$
δ	Cloak (or perturbation) between $\mathbf{x}$ and $\hat{\mathbf{x}}$
ε	Perturbation budget
κ	Trade-off hyperparameter
I	GAN inversion technique
G	Generator
E	Encoder for the latent space
F	Feature extractor for the feature space
$\mathcal{L}$	Loss function
G_t	Target generator controlled by the adversary
G_s	Shadow generator controlled by the defender
E_t	Target encoder controlled by the adversary
E_s	Shadow encoder controlled by the defender
I_o	Optimization-based inversion
I_h	Hybrid inversion
$\mathcal{L}_\mathrm{rec}$	Reconstruction loss
$\mathcal{L}_\mathrm{percept}$	Perceptual loss
$\mathcal{L}_\mathrm{cos}$	Cosine similarity loss
$\mathcal{L}_\mathrm{mse}$	MSE similarity loss

B GAN Models and Datasets

DCGAN. DCGAN [39] uses convolutions in the discriminator and fractional-strided convolutions in the generator.
WGAN. WGAN [16] minimizes the Wasserste in distance between the generated and real data distributions, which offers more model stability and makes the training process easier.
StyleGANv1/v2. StyleGANv1 [27] implicitly learns hierarchical latent styles for image generation. It takes per-block incorporation of style vectors and stochastic variation as inputs to generate a synthetic image. The StyleGANv2 [28] further improves the image quality by proposing weight demodulation, path length regularization, redesigning generator, and removing progressive growing.

CelebA. CelebA [33] is a face dataset consisting of 200K celebrity images with 40 attribute annotations each.
FFHQ. Flickr-Faces-HQ (FFHQ) [27, 28] is a high-quality image dataset of human faces crawled from Flickr, which consists pixels and contains considerable variation in terms of age, of 70,000 high-quality human face images of 1024 × 1024 ethnicity, and image background.

Algorithm 1: Cloaking Facial Image of Cloak-0

Input: A target image $\mathbf{x}$ to cloak; a pre-trained target generator $G_\mathrm{t}(\cdot)$; a shadow encoder $E_\mathrm{s}(\cdot)$; a pre-trained ResNet feature extractor F; cosine similarity $\mathcal{L}_\mathrm{cos}(\cdot,\cdot)$; MSE similarity $\mathcal{L}_\mathrm{mse}(\cdot,\cdot)$; minibatch m; perturbation budget ε; trade-off κ.

Output: The trained shadow encoder E_s and the cloaked image $\hat{\mathbf{x}}$.

Initialize $\mathcal{L}_\mathrm{rec}(\cdot,\cdot) = -\mathcal{L}_\mathrm{cos}(\cdot,\cdot) + \mathcal{L}_\mathrm{mse}(\cdot,\cdot)$;
for *number of training iterations* **do**
  sample a minibatch of latent codes $\mathbf{z}' \in \mathcal{N}(0,1)$;
  $\min_{\Theta_{E_\mathrm{s}}} \mathcal{L}_\mathrm{rec}\Big(E_\mathrm{s}\big(G_\mathrm{t}(\mathbf{z}')\big), \mathbf{z}'\Big)$
end
Initialize $\mathbf{x}_\mathrm{t}$ = optimization-based inversion($\mathbf{x}$);
Initialize $\delta \in \mathcal{N}(0,1)$ and $|\delta|_\infty < \varepsilon$;
Initialize κ;
for *number of optimized iterations* **do**
  $\max_\delta \kappa\Big(\mathcal{L}_\mathrm{rec}\big(E_\mathrm{s}(\mathbf{x}+\delta), \mathbf{x}_\mathrm{t}\big)\Big) + (1-\kappa)\Big(\mathcal{L}_\mathrm{rec}\big(F(\mathbf{x}+\delta), F(\mathbf{x})\big)\Big)$;
  clip δ for $|\delta|_\infty < \varepsilon$;
  clip $\mathbf{x}+\delta$ for $\mathbf{x}+\delta \in [0,1]$;
end
$\hat{\mathbf{x}} = \mathbf{x} + \delta$;
return $E_\mathrm{s}, \hat{\mathbf{x}}$

Algorithm 2: Cloaking Facial Image of Cloak-1/4

Input: A target image $\mathbf{x}$ to cloak; a pre-trained ResNet feature extractor F; cosine similarity $\mathcal{L}_\mathrm{cos}(\cdot,\cdot)$; MSE similarity $\mathcal{L}_\mathrm{mse}(\cdot,\cdot)$; perturbation budget ε.

Output: The cloaked image $\hat{\mathbf{x}}$.

Initialize $\mathcal{L}_\mathrm{rec}(\cdot,\cdot) = -\mathcal{L}_\mathrm{cos}(\cdot,\cdot) + \mathcal{L}_\mathrm{mse}(\cdot,\cdot)$;
Initialize $\delta \in \mathcal{N}(0,1)$ and $|\delta|_\infty < \varepsilon$;
for *number of optimized iterations* **do**
  $\max_\delta \mathcal{L}_\mathrm{rec}\big(F(\mathbf{x}+\delta), F(\mathbf{x})\big)$;
  clip δ for $|\delta|_\infty < \varepsilon$;
  clip $\mathbf{x}+\delta$ for $\mathbf{x}+\delta \in [0,1]$;
end
$\hat{\mathbf{x}} = \mathbf{x} + \delta$;
return $\hat{\mathbf{x}}$

Algorithm 3: Cloaking Facial Image of Cloak-2

Input: A target image $\mathbf{x}$ to cloak; a pre-trained target encoder $E_t(\cdot)$; a pre-trained ResNet feature extractor F; cosine similarity $\mathcal{L}_{\cos}(\cdot,\cdot)$; MSE similarity $\mathcal{L}_{\text{mse}}(\cdot,\cdot)$; perturbation budget ε; trade-off κ.

Output: The cloaked image $\hat{\mathbf{x}}$.

Initialize $\mathcal{L}_{\text{rec}}(\cdot,\cdot) = -\mathcal{L}_{\cos}(\cdot,\cdot) + \mathcal{L}_{\text{mse}}(\cdot,\cdot)$;
Initialize $\delta \in \mathcal{N}(0,1)$ and $|\delta|_\infty < \varepsilon$;
Initialize κ;
for *number of optimized iterations* **do**
  $\max_\delta \kappa\Big(-\mathcal{L}_{\text{rec}}\big(E_t(\mathbf{x}+\delta),0\big)\Big) + (1-\kappa)\Big(\mathcal{L}_{\text{rec}}\big(F(\mathbf{x}+\delta),F(\mathbf{x})\big)\Big)$;
  clip δ for $|\delta|_\infty < \varepsilon$;
  clip $\mathbf{x}+\delta$ for $\mathbf{x}+\delta \in [0,1]$;
end
$\hat{\mathbf{x}} = \mathbf{x}+\delta$;
return $\hat{\mathbf{x}}$

Algorithm 4: Cloaking Facial Image of Cloak-3

Input: A target image $\mathbf{x}$ to cloak; a pre-trained target encoder $E_t(\cdot)$; a shadow encoder E_s; a shadow generator G_s; a pre-trained ResNet feature extractor F; cosine similarity $\mathcal{L}_{\cos}(\cdot,\cdot)$; MSE similarity $\mathcal{L}_{\text{mse}}(\cdot,\cdot)$; perturbation budget ε; trade-off κ.

Output: The trained shadow encoder E_s, the trained shadow generator G_s and the cloaked image $\hat{\mathbf{x}}$.

Initialize $\mathcal{L}_{\text{rec}}(\cdot,\cdot) = -\mathcal{L}_{\cos}(\cdot,\cdot) + \mathcal{L}_{\text{mse}}(\cdot,\cdot)$;
for *number of training iterations* **do**
  sample a minibatch of latent codes $\mathbf{z}' \in \mathcal{N}(0,1)$;
  $\min_{\Theta_{E_s}} \mathcal{L}_{\text{rec}}\Big(E_s\big(G_s(\mathbf{z}')\big),\mathbf{z}'\Big)$;
  $\max_{\Theta_{G_s}} \mathcal{L}_{\text{rec}}\Big(E_s\big(G_s(\mathbf{z}')\big),\mathbf{z}'\Big)$;
end
Initialize $\delta \in \mathcal{N}(0,1)$ and $|\delta|_\infty < \varepsilon$;
Initialize κ;
for *number of optimized iterations* **do**
  $\max_\delta \kappa\Big(-\mathcal{L}_{\text{rec}}\big(E_s(\mathbf{x}+\delta),0\big)\Big) + (1-\kappa)\Big(\mathcal{L}_{\text{rec}}\big(F(\mathbf{x}+\delta),F(\mathbf{x})\big)\Big)$;
  clip δ for $|\delta|_\infty < \varepsilon$;
  clip $\mathbf{x}+\delta$ for $\mathbf{x}+\delta \in [0,1]$;
end
$\hat{\mathbf{x}} = \mathbf{x}+\delta$;
return $E_s, G_s, \hat{\mathbf{x}}$