1 2 90

UNIVERSIDADE D
COIMBRA

Inês Marques Pereira

DISRUPTING DEEPFAKES – A CROSS-MODEL EVALUATION

Dissertação no âmbito de Mestrado em Engenharia Eletrotécnica e de Computadores, no ramo de Robótica, Controlo e Inteligência Artificial, orientada pelo Professor Doutor Jorge Manuel Moreira de Campos Pereira Batista e apresentada ao Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

Fevereiro de 2024

FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE Đ
COIMBRA

Disrupting DeepFakes - A Cross-Model Evaluation

Inês Marques Pereira

Coimbra, February 2024

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE Đ COIMBRA

Disrupting DeepFakes - A Cross-Model Evaluation

Supervisor:

Prof. Dr. Jorge Manuel Moreira de Campos Pereira Batista

Jury:

Prof. Dr. Luís Alberto da Silva Cruz

Prof. Dr. Jorge Manuel Moreira de Campos Pereira Batista

Prof. Dr. Cristiano Premebida

Dissertation submitted in partial fulfillment for the degree of Master of Science in Electrical and Computer Engineering.

Coimbra, February 2024

Agradecimentos

Esta dissertação não teria sido possível sem a orientação e apoio de várias pessoas, que de uma forma ou de outra, contribuíram para a conclusão deste trabalho e de todo o meu percurso académico.

Em primeiro lugar, gostaria de agradecer ao meu orientador Professor Jorge Batista pela sua orientação e apoio ao longo deste processo. A sua dedicação e visão foram fundamentais para o desenvolvimento deste trabalho. Ao meu colega de mestrado Henrique, obrigada por todo o companheirismo e amizade, sem ti teria sido mais desafiador. Aos meus colegas da Universidade - Marta, Matias, Mara e Luís - um obrigada por todos os momentos, tanto de estudo como de diversão.

Às minhas amigas mais chegadas - Anaísa, Maria, Beatriz Costa e Beatriz Lopes - obrigada pela vossa amizade, por todos os momentos partilhados e pelas palavras de apoio. A vossa amizade foi essencial durante todo este percurso. Ao meu namorado Rúdi por todo o amor, paciência e ajuda que me deu nesta reta final. Foste o meu porto de abrigo e, por isso, um obrigada especial.

Por último, mas não menos importante, um obrigada muito especial à minha família. Aos meus pais, obrigada por nunca desistirem de mim e por me terem dado sempre apoio incondicional, mesmo nos momentos mais difíceis. Nada disto teria acontecido sem vocês e, por isso, estarei eternamente grata. Aos meus avós por me receberem sempre com amor e um sorriso, por terem sempre depositado confiança em mim, e por mostrarem querer estar sempre presentes.

A todos, o meu sincero obrigada.

Abstract

In recent years, with the advance of generative models, DeepFake has become a real risk to society and introduced potential threats to individual privacy and political security. In response to this escalating concern, considerable efforts have been devoted to the development of defense mechanisms against DeepFake manipulation.

While DeepFake Detection, a passive defense strategy, has been employed as an ex-post countermeasure, its efficacy in preventing the spreading of misinformation is limited. To address this problem, researchers have explored proactive defense techniques, such as the introduction of adversarial noises into source data, aiming to disrupt DeepFake manipulation, making it impossible to generate realistic images. However, existing studies on DeepFake Disruption often overlook the critical aspects of the transferability of adversarial attacks and their resilience against image reconstruction methods.

Unfortunately, most current disruption methods fail to be effective in real-world scenarios, where the specific DeepFake model and the targeted attribute for manipulation are unknown. Consequently, this dissertation seeks to critically examine existing disruption methods, evaluating their capacity to transfer seamlessly across diverse DeepFake models and domains. Additionally, this research aims to assess the robustness of these methods against various image reconstruction techniques, thereby contributing to the development of more effective and versatile defenses against the growing threat of DeepFake technology.

Keywords— DeepFake Disruption, GANs, Adversarial Attacks, Cross-Modality

Resumo

Nos últimos anos, com o avanço dos modelos generativos, a técnica DeepFake tornou-se um risco para a sociedade e introduziu potenciais ameaças à privacidade individual e à segurança política. Em resposta a esta preocupação crescente, foram dedicados consideráveis esforços ao desenvolvimento de mecanismos de defesa contra a manipulação de DeepFake.

Enquanto a deteção de DeepFake, uma estratégia de defesa passiva, tem sido utilizada como uma contramedida ex-post, a sua eficácia em prevenir a propagação de desinformação é limitada. Para lidar com este problema, estudos atuais têm explorado técnicas de defesa proativas, como a adição de perturbações adversariais a imagens, com o objetivo de perturbar a manipulação de DeepFake, tornando impossível a geração de imagens realistas. No entanto, a maioria dos métodos de Disrupção de DeepFake negligencia aspetos críticos para uma defesa resistente, nomeadamente a transferibilidade dos ataques adversariais e a sua resistência contra métodos de reconstrução de imagem.

Infelizmente, a maioria dos métodos atuais de disrupção falha em ser eficaz em cenários do mundo real, onde o modelo específico de DeepFake e o atributo alvo para manipulação são desconhecidos. Posto isto, esta tese tem como objetivo examinar criticamente os métodos de disrupção existentes, avaliando a sua capacidade de transferência entre modelos e domínios diversos de DeepFake. Para além disso, este estudo visa avaliar a robustez deste tipo de métodos contra várias técnicas de reconstrução de imagem, contribuindo assim para o desenvolvimento de defesas mais eficazes e versáteis contra a ameaça crescente da tecnologia DeepFake.

Keywords— Disrupção de DeepFake, *GANs*, Ataques Adversariais, *Cross-Modality*

“The potential benefits of artificial intelligence are huge, so are the dangers."

Dave Waters

Contents

List of Acronyms

AI Artificial Intelligence

GAN Generative Adversarial Network

FGSM Fast Gradient Sign Method

PGD Projected Gradient Descent

PRNU Photo Response Non-Uniformit

DNN Deep Neural Networks

AREN Adaptive Residuals Network

CNN Convolutional Neural Networks

I-FGSM Iterative Fast Gradient Sign Method

CW Carlini and Wagner Attack

MI-FGSM Momentum Iterative Fast Gradient Sign Method

OGAN Oscillating GAN

CDMAA Cross-Domain and Model Adversarial Attack

MGDA Multiple Gradient Descent Algorithm

TCA-GAN Transferable Cycle Adversary Generative Adversarial Network

MagDR Mask-guided Detection and Reconstruction

SSIM Structural Similarity Index Measure

PSNR Peak Signal-to-Noise Ratio

MSE Mean Squared Error

TPE Tree-Structured Parzen Estimator

SMBO Sequential Model-Based Optimization

EI Expected Improvement

EDSR Enhanced Deep Super-Resolution

DWT Discrete Wavelet Transform

List of Figures

List of Tables

1 Introduction

1.1 Context and Motivation

The emergence of DeepFake technology, characterized by its ability to create or manipulate facial appearances through sophisticated generative approaches, has ushered in a new era of digital manipulation. DeepFake technology allows individuals to alter a person's identity, expression, and attributes within an image or video in a remarkably realistic manner. While its applications can be benign, such as aiding the film industry in recreating appearances for deceased celebrities, the malicious misuse of DeepFake technology poses significant threats to our digital society [21]. This introduction delves into the multifaceted landscape of DeepFake technology, addressing its potential for impersonation, manipulation of political events, creation of explicit content, and its far-reaching impact on cybersecurity and privacy.

One notable example is the creation of a DeepFake video of Barack Obama by comedian Jordan Peele in collaboration with BuzzFeed in April 2018. This public service announcement (PSA) aimed to raise awareness about the technology's capabilities [21]. However, this technology is not limited to political impersonations but extends to the creation of DeepFake pornography and fake nude images, predominantly targeting women. DeepFake pornography has become the most prevalent form of malicious DeepFake content, negatively impacting the lives of many, including celebrities and private individuals [22] [21].

Beyond the realm of misinformation and explicit content, DeepFake technology is also reshaping the landscape of cybersecurity by facilitating fraud and online scams. The increasing accessibility and realism of DeepFake technology make it a formidable threat to the security and privacy of individuals. The popular mobile application, FaceApp, widely used for posting Instagram selfies, exemplifies the accessibility of these technological advancements. All that is required to use this app is a smartphone and its installation. With a simple click, users can transform their selfies into younger or older versions, representing a form of deepfake imagery. The creation of deepfakes on the web is escalating at a rapid pace. To illustrate, around February 2021, approximately 60,000 synthetic videos of this nature were circulating online. Therefore, it becomes imperative to develop effective countermeasures to combat DeepFake technology.

In response to the growing concerns surrounding DeepFake technology, various defense methods have been proposed, both passively and proactively. Passive DeepFake defense strategies aim to identify whether an image or video has been generated artificially using AI or captured naturally by a camera, allowing for the differentiation between genuine and DeepFake data [21]. Proactive DeepFake defense, on the other hand, involves introducing adversarial perturbations to the source images, disrupting the DeepFake creation process and producing distorted results [1] [17]. This technique, known as DeepFake Disruption, distinguishes itself as a promising approach in the fight against DeepFakes. The difference between these two types of defenses is illustrated in Figure 1.1.

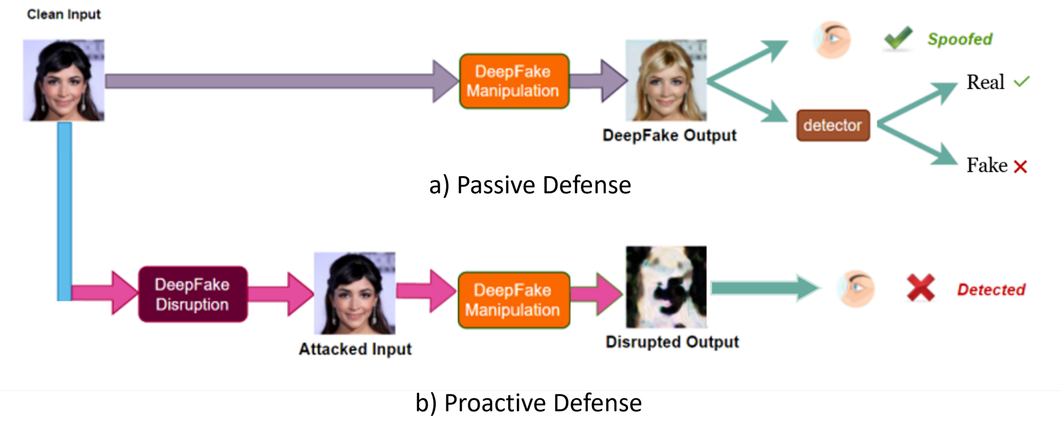


Figure 1.1: Illustration of DeepFake defenses. Adapted from [1].

Although DeepFake Detection has played a vital role in identifying DeepFakes, it serves as an ex-post countermeasure, offering limited ability to prevent the spread of misinformation. By the time a DeepFake is detected, the harm has already been inflicted. As such, this dissertation focuses on studying techniques that prevent attackers from synthesizing DeepFake images, with a particular emphasis on DeepFake Disruption. This method, being relatively recent, has shown considerable promise in disrupting DeepFake creation by introducing human-imperceptible perturbations to the images, resulting in visually noticeable artifacts that fail to deceive the human eye, as illustrated in Figure 1.2.

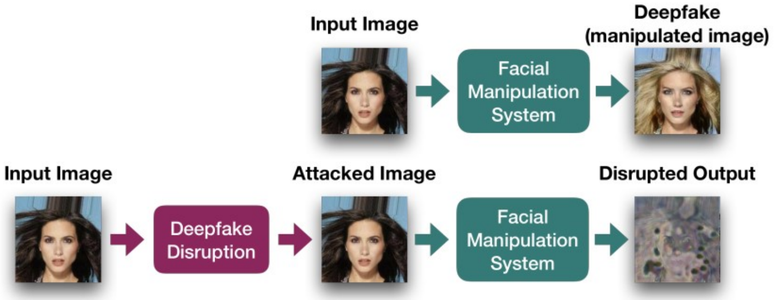

Input Image
Facial Manipulation System
Deepfake (manipulated image)
Input Image
Deepfake Disruption
Attacked Image
Facial Manipulation System
Disrupted Output

Figure 1.2: Illustration of DeepFake disruption. Taken from [2].

The success of DeepFake Disruption is contingent upon the creation of images that are sufficiently deteriorated to the point where they must be discarded or where the modifications are perceptually evident [2]. In this dissertation, there will be an exploration of the theoretical foundations, practical applications, and the efficacy of DeepFake Disruption as a potent weapon against the proliferation of DeepFake technology and its associated threats.

1.2 Challenges and Breakthroughs

Over the last few years, the spotlight has increasingly turned toward the development of DeepFake Disruption methods as a means of safeguarding against the creation of DeepFake. DeepFake, generated by various Generative Adversarial Networks (GANs), can manipulate a wide array of attributes in face images, ranging from changing hair color to altering gender and age, among others. The complexity of the task lies in developing a robust defense method that can effectively combat a broad spectrum of DeepFake models and domains.

Most studies focus on the concept of gray-box adversarial attacks, which implies assuming the specific DeepFake model that will be used. These approaches train their adversarial attacks with a predefined model in mind, leading to impressive results when tested on that very model. However, this methodology falters when confronted with a different model, as each one has its distinct approach to DeepFake generation. What proves effective for one model may prove ineffective for another. Further complicating matters, some studies use a white-box adversarial attack, assuming what domain will be used as well. While effective within their predetermined domain, these approaches lose efficiency when the situation changes, resulting in reduced practicality due to their susceptibility to variations. So, without the knowledge of what DeepFake model will be employed or what conditional variables will be set to tamper with images, these adversarial attacks have great limitations in practice [9] [18].

In a real-life scenario, the identity of the DeepFake model and the domain employed by attackers are unknown, making it essential to devise defenses that function seamlessly in any conceivable scenario. The answer lies in the development of an effective defense that operates under the premise of a black-box adversarial attack. Such an attack operates without prior knowledge of the model or domain in use, closely resembling the challenges of real-life situations. The ideal DeepFake Disruption employing a black-box adversarial attack should effectively disrupt all DeepFake models and perform consistently across diverse domains. To achieve this, research focuses on training adversarial attacks across a range of DeepFake models and domains, culminating in cross-model perturbations. When applied to images, these perturbations prevent attackers from producing realistic DeepFakes.

As DeepFake Disruption evolves and progresses, attackers simultaneously develop countermeasures to eliminate adversarial attacks and neutralize the disruption. The majority of adversarial attacks, such as Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD), rely on gradient-based or optimization-based strategies to generate subtle adversarial perturbations. Unfortunately, research indicates that such perturbations can be easily removed or destroyed through a simple input reconstruction, enabling attackers to erase the perturbation and continue their DeepFake endeavors [12] [17].

1.3 Objectives

This dissertation aims to conduct a comprehensive examination of existing DeepFake disruption techniques, with a primary emphasis on their adaptability to a diverse range of Deepfake models and application domains. Simultaneously, it will assess their resilience against image reconstruction methods. The central objective of this research is to perform a cross-model evaluation, making comparisons among various adversarial attack types (white-box, grey-box, and black-box) and determining which method holds the most promise for real-world scenarios.

Despite notable advancements in DeepFake disruption, research in this field remains somewhat limited in terms of cross-model perspectives and the robustness of solutions against image reconstruction challenges. Consequently, this dissertation seeks to offer a thorough understanding of the strides made by prior studies while emphasizing the unresolved issues that persist. The goal is to illuminate the path forward in the ongoing struggle against DeepFake threats and identify the essential steps necessary for the development of a robust and universally applicable defense mechanism.

In pursuing these objectives, this dissertation aims to make a substantial contribution to the preservation of trust and security in an era characterized by the continuously evolving landscape of technological deception.

1.4 Document Structure

The outline of this dissertation is the following:

- **Chapter 2:** State-of-the-Art reviews the most relevant works in the literature and provides a theoretical background about the subject.

- **Chapter 3:** Background Materials introduces essential methodologies, establishing a contextual foundation for the primary research focus of this work.

- **Chapter 4:** Methodology describes the implementation and the methodologies used during the course of this work.

- **Chapter 5:** Results and Discussion provides a detailed examination of the experimental results.
- **Chapter 6:** Conclusion and Future Work summarises what was concluded from the results and suggests improvements.

2 State-of-the-Art

This chapter reviews the pertinent literature contributions related to the field of DeepFake Disruption, aiming to incorporate the notable advancements within the existing body of literature and granting an extensive insight into the state-of-the-art practices employed in DeepFake Disruption's research.

Initially, in Section 2.1 the discussion initiates with an exploration of DeepFake creation to enhance comprehension regarding what constitutes DeepFake and the diverse methodologies employed for its generation. This foundational understanding sets the stage for a more nuanced examination of the subsequent sections.

Moving on to Sections 2.2 and 2.3, these segments will introduce various aspects of DeepFake defenses. In light of the heightened concerns and risks associated with DeepFake technology, countermeasures have been developed. Section 2.2 introduces passive DeepFake Defense, also known as DeepFake Detection, offering insights into different detection types and their efficacy. Lastly, Section 2.3 introduces DeepFake Disruption as a proactive defense strategy, exploring the background of DeepFake Disruption and elucidating its concept. Additionally, it details the methodologies currently in use. This comprehensive approach provides a thorough examination of both the passive and proactive dimensions of countering DeepFake threats.

2.1 DeepFake Creation

In the past years, Generative Adversarial Networks have made remarkable progress in image synthesis and manipulation. DeepFake, building upon the success of GANs, generates forgery facial images and videos, presenting potential threats to individual privacy and political security. There are four common types of DeepFake creation: entire face synthesis, attribute manipulation, identity swap, and expression swap. Figure 2.1 graphically summarises each facial manipulation group.

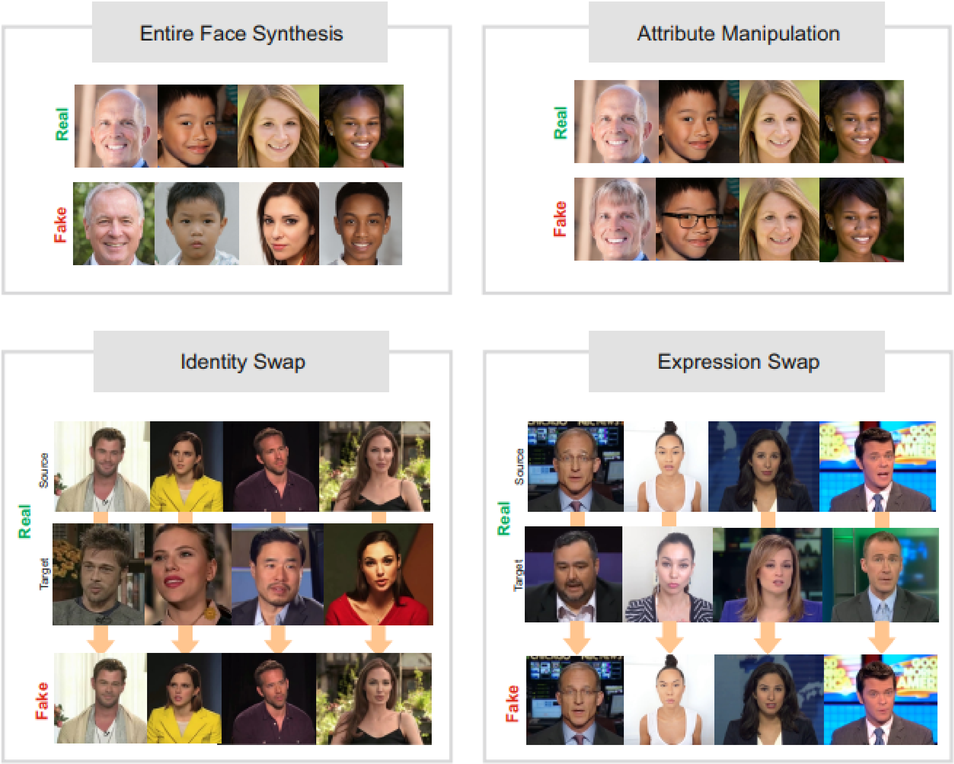


Figure 2.1: Real and fake examples of each facial manipulation group. Taken from [3].

Entire synthesis generates fabricated images without any basis in reality, usually through powerful GANs, like the StyleGAN approach detailed in [23]. These techniques deliver remarkable results, achieving high-quality facial images characterized by an impressive level of realism. While this manipulation holds potential for diverse sectors like the video game and 3D-modeling industries, it also carries the risk of misuse. For instance, it could be exploited to fabricate highly realistic fake profiles on social networks, facilitating the spread of misinformation. Attribute manipulation involves faltering specific facial features, modifying both simple attributes (such as hair color or baldness) and complex ones (like gender or age), typically facilitated by GANs such as the StarGAN method outlined in [14] and STGAN proposed in [24]. This technology can be used by consumers to try on a wide range of products such as makeup, glasses, and hairstyles in a virtual setting. The identity swap, popularly known as face swap, consists of replacing one person's face in a video with another person's face. This type of manipulation can be beneficially used in the film industry, but it could also be used for malicious purposes such as the creation of DeepFake pornography, financial fraud, and others. DeepFaceLab [25] is a popular and available tool for identity swapping. Similarly, expression swap, also known as face reenactment, involves altering the facial expression of an individual. GANimation [26] is an example of an available expression swap tool.

2.2 DeepFake Detection

Determining the authenticity of facial images by distinguishing between real and fake images is a simple way of defending against DeepFakes. However, DeepFake detection poses many challenges, such as generalization to tackle unknown synthetic techniques and vulnerability to evading adversarial attacks. There are three main types of DeepFake detection: Spatial-based Detection, Frequency-based Detection, and Biological Signal-based Detection.

2.2.1 Spatial-based Detection

Spacial-based Detection consists of analyzing spatial discrepancies within an image or video frame to detect potential manipulations. This method examines spatial elements such as features, textures, and alignments, to identify anomalies that indicate the presence of a DeepFake. Within Spatial-based Detection, there are five main types of detection:

- **Image forensics based detection:** This approach uses traditional forensics-based techniques, conducting a thorough examination of images and videos at the pixel level to detect irregularities. Li *et al.* [27] stated that the distinctions between computer-generated faces and authentic ones become apparent in the chrominance components, particularly within the residual domain. To counter unfamiliar GANs, they suggest training a one-class classifier using real faces and leveraging these chrominance differences. However, the effectiveness of this method against image alterations, such as perturbation attacks, remains uncertain. An alternative study, detailed in [28], utilizes the Photo Response Non-Uniformity (PRNU) pattern to differentiate real from fake. PRNU refers to a noise pattern present in digital images, originating from the camera's light sensor.

- **DNN-based detection:** This method relies entirely on data-driven approaches, using either established or newly designed Deep Neural Network (DNN) models. These models extract spatial features to enhance detection's effectiveness and its ability to generalize across different scenarios. However, ongoing research is focused on enhancing the resilience of DNN-based detection methods against adversarial attacks, prompted by recent studies highlighting their vulnerability to additive noises and lack of robustness. A study made by Liu *et al.* [29] proposed a new architecture called Gram-Net. This technique uses comprehensive global image texture representations to bolster the detection of fake images, and experimental results showed a strong robustness against downsampling, JPEG compression, blur and noise.

- **Obvious artifacts clues:** Generated DeepFakes often contain perceptible imperfections that can be exploited for detection using simple DNN models. These artifacts typically manifest within specific local patches. Chai *et al.*'s study proposed a convolutional method, as referenced in [30], for training classifiers to concentrate on these image patches. Their approach demonstrates robustness and generalization across various network architectures, image datasets, and more.

- **Detection and localization:** Some studies focus on identifying the manipulated regions, contributing evidence to forensic practices and sparking further research to develop better DeepFake Detection capabilities. FakeLocator, referenced in [31], investigated the architecture of existing GANs and observed that upsampling methods exhibit obvious clues for detection and forgery localization. Using an encoder-decoder network, this method extracted forged textures while devising a grayscale predictive map, significantly improving detection and localization accuracy. FakeLocator demonstrated strong adaptability across various GANs and showed resilience against perturbation attacks such as compression and blur.

- **Facial image preprocessing:** Certain studies argue in favor of preprocessing facial images before their submission to binary classifiers, asserting that this step enhances DeepFake Detection. The processed DeepFakes may reveal their manipulated textures, enabling straightforward identification by simpler classifiers, such as traditional machine learning methods. Guo *et al.* conducted a study, as cited in [32], where they designed an adaptive residuals network (AREN) to suppress image content by learning prediction residuals through an adaptive convolution layer. Subsequently, they constructed a fake face detector, ARENnet, by combining AREN with CNN, specifically to address fake videos afflicted by degradations. This approach demonstrates robustness against perturbation attacks and displays strong generalization capabilities across different GANs.

2.2.2 Frequency-based Detection

Frequency-based detection involves examining the frequency domain aspects of images or videos to spot potential signs of manipulation. This method focuses on analyzing the distribution of frequencies, such as specific patterns or alterations that might indicate tampering. By detecting irregularities in the frequency spectrum, algorithms can spot anomalies that suggest the presence of a DeepFake. An illustration of this technique can be seen in Figure 2.3.

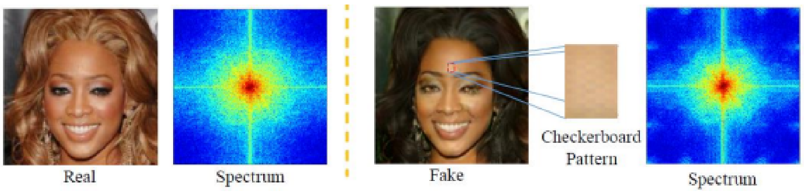


Figure 2.2: The difference between real and fake from the frequency domain. Taken from [4].

Within Frequency-based Detection, there are two main types of detection:

- **GAN-based artifacts:** Some researchers have shifted their attention away from examining visual artifacts and instead are investigating the inherent design flaws present in current GANs. These imperfections offer distinct cues to differentiate between authentic and synthetic faces. While their exploration primarily occurs within the frequency domain, they encounter universal flaws inherent in existing GANs. AutoGAN, referenced in [33], identifies a distinct artifact inherent in GANs stemming from their common upsampling design. To simulate these artifacts without accessing pre-trained GANs, the researchers propose a GAN simulator. This simulator enhances the generalization ability of existing detectors. The identified artifacts manifest as duplicated spectra in the frequency domain. A classifier is then trained using this frequency spectrum to discern GAN-generated fake faces. The author, Zhang *et al.*, suggests that these observed GAN-based artifacts display good generalization across various synthetic techniques with similar architectures. However, their resilience against perturbation attacks remains unexplored.
- **Frequency domain feature:** Discrepancies between genuine and artificially generated faces can also emerge within the frequency domain. A study made by Frank *et al.*, referenced in [34], conducted a comprehensive analysis of images generated by various GANs. This study noted that during the upsampling process, GANs frequently introduced noticeable irregularities into the images. Experiments demonstrated that a classifier with a simple linear model and a CNN-based model could both achieve promising results on the frequency domain. Additionally, the classifier trained on the frequency domain is robust against perturbation attacks.

2.2.3 Biological Signal-based Detection

Biological Signal-based Detection involves studying the biological signals of an image or video. These signals exist in both real and synthesized fake videos, and they differ from one another. In real videos, these signals exhibit natural and realistic characteristics, whereas in synthesized videos, they often lack quality and most perceptual biological signals vanish. This includes discrepancies between visual and audio cues, leading to inconsistencies in synthetic videos.

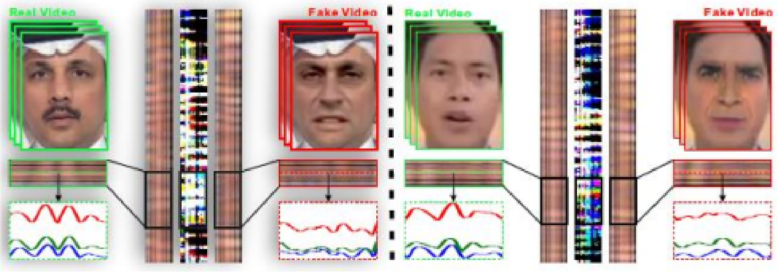


Figure 2.3: The difference between real and fake from the biological signal domain. Taken from [5].

Within Biological Signal-based Detection, there are three main types of detection:

- **Visual-audio inconsistency:** Some studies use visual-audio inconsistency in videos to detect DeepFakes. A study made by Mittal *et al.*, referenced in [35], developed a siamese network for modeling the visual and audio in videos with a combination of two triplets loss functions for measuring similarity. One of the loss functions computes the similarity between visual and audio, and the other calculates the effect cues, like perceived emotion. Experiments show a better performance than conventional DNN-based methods for DeepFake Detection.
- **Visual inconsistency:** Differences in visual consistency suggest that the generated faces lack naturalness, particularly in their shape, facial attributes, and distinctive landmarks. A study, referenced in [36], noted that the synthesized faces are always in fixed sizes due to the limitation of computation resources and the production time of DeepFake algorithms. This leaves artifacts in warping to match the source face, which can be leveraged for DeepFake detection, by training a CNN model to detect these anomalies.
- **Biological signal in video:** Biological signals are challenging to duplicate in videos. FakeCatcher, referenced in [37], leverages six distinct biological signals, extracting them to exploit the spatial and temporal consistency, thereby verifying the authenticity of real videos taken by cameras.

2.3 DeepFake Disruption

Adversarial attacks, which initially gained attention for their impact on the vulnerabilities of deep neural networks (DNNs), have also been studied for their potential effects on DeepFakes. Recent studies delve into using adversarial techniques as a defensive strategy against the potential risks associated with synthetic media. Researchers have explored various adversarial attack methods, like the Fast Gradient Sign Method (FGSM) [38], Iterative Fast Gradient Sign Method (I-FGSM) [39], Projected Gradient Descent (PGD) [40], and Carlini and Wagner Attack (C&W) [41].

- **Fast Gradient Sign Method (FGSM):** Goodfellow *et al.* [38] introduced the Fast Gradient Sign Method (FGSM) algorithm, aiming to demonstrate how the presence of adversarial examples stems from the high-dimensional linearity inherent in deep neural networks. This algorithm operates on the principle of generating adversarial perturbations based on the maximum gradient change direction within the deep learning model [42]. These perturbations are then added to the image, generating adversarial examples. The equation for FGSM to generate perturbations is as follows:

$$\delta = \epsilon sign(\nabla_x J_\theta(\theta, x, y)), \tag{2.1}$$

  where δ represents the generated perturbation; θ and x are the parameters of the model and the input to the model, respectively; y denotes the target associated with x; J_θ is the loss function during model training; and ϵ denotes a constant. This solution is motivated by linearizing the cost function and solving for the perturbation that maximizes the cost subject to an L∞ constraint. The FGSM algorithm demonstrates rapid attack speed as it employs a single-step approach. However, this method might encounter lower success rates when generating adversarial examples due to its single-step nature.

- **Iterative Fast Gradient Sign Method (I-FGSM):** Kurakin *et al.* [39] proposed the Iterative Fast Gradient Sign Method (I-FGSM) which is an extension of the Fast Gradient Sign Method (FGSM) used in adversarial attacks against deep neural networks. While FGSM generates adversarial examples in a single step, I-FGSM operates by applying FGSM iteratively across multiple steps. Its iterative nature allows for the generation of more impactful adversarial examples by repeatedly adjusting the input data in the direction that maximizes the model's error, while still aiming to maintain the perturbations within specific bounds.

- **Projected Gradient Descent (PGD):** The Projected Gradient Descent (PGD) [40] is an iterative gradient-based method that adjusts input data incrementally by following the direction that maximizes the model's loss. Its focus is to make small, calculated changes to input data iteratively, attempting to generate potent adversarial examples while staying within specific constraints. The PGD attack is essentially the same as I-FGSM attack. The only difference is that PGD initializes the example to a random point in the ball of interest (decided by the L∞ norm) and does random restarts, while I-FGSM initializes to the original point.

- **Carlini and Wagner Attack (C&W):** The Carlini and Wagner Attack (C&W) [41] is one of the most powerful attacks, which uses three different vector norms: 1) the L2 attack uses a smoothing of clipped gradient descent approach, displaying low distortion; 2) the L0 attack uses an iterative algorithm that, at each iteration, fixes the pixels that do not have much effect on the classifier and finds the minimum amount of pixels that need to be altered; and 3) the L∞ attack also uses an iterative algorithm with an associated penalty, penalizing every perturbation that exceeds a predefined value, formally defined as:

$$\min c \cdot f(x+\delta) + \sum_i [(\delta_i - \tau)^+], \tag{2.2}$$

where δ is the perturbation, τ is the penalty threshold (initially 1, decreasing in each iteration), and c is a constant. The value for c starts as a very low value (e.g., 10^{-4}), and each time the attack fails, the value for c is doubled. If c exceeds a threshold (e.g., 10^{10}), it aborts the search.

These attacks can be useful to disrupt DeepFakes, by incorporating adversarial disruptions in the generation or training stages of DeepFake models and thus diminishing the quality or precision of the resulting DeepFake. By introducing noise or distortions while creating, it could render the DeepFake less persuasive or more challenging for the model to generate realistic outcomes.

2.3.1 DeepFake Disruption Methodologies

In recent years, DeepFake Disruption has emerged as a critical area of research, attracting significant attention. This methodology consists of generating an adversarial perturbation η, which is subsequently incorporated into an input image:

$$\tilde{x} = x + \eta, \tag{2.3}$$

where $\tilde{x}$ is the generated disrupted input image and x is the input image. By feeding the original image or the disrupted input image to a generator the mappings $G(x) = y$ and $G(\tilde{x}) = \tilde{y}$ are obtained, respectively, where y and $\tilde{y}$ are the translated output images and G is the generator of the image translation GAN [2]. A disruption is considered successful when it introduces perceptible corruptions or modifications onto the output $\tilde{y}$ of the network leading a human observer to notice that the image has been altered and therefore distrust its source.

Studies in this field can be categorized into three main groups: white-box attack, gray-box attack, and black-box attack. White-box attacks involve generating perturbations designed for a particular domain (such as hair color or gender) and a specific DeepFake model. Gray-box attacks assume the DeepFake model that will be used. Meanwhile, black-box attacks generate perturbations without assuming any specific domain or method utilized in DeepFake creation.

2.3.1.1 White-Box Attacks

Sun *et al.* [6] introduced the Landmark Breaker, marking the initial dedicated technique to disrupt facial landmark extraction, aiming to disrupt the generation of DeepFake videos. Several

DeepFake methodologies rely on landmark extraction. By integrating adversarial perturbations intended to disrupt facial landmark extraction, the alignment of the input face is affected, consequently degrading the quality of the resulting DeepFake, as can be seen in Figure 2.4. Facial landmarks serve as crucial reference points, encompassing key locations such as the tips and midpoints of the eyes, nose, mouth, eyebrows, and contours. The adversarial attack aims to mislead DNN-based facial landmark extractors, particularly in predicting landmark heat-maps, a common initial step in many contemporary DNN-based facial landmark extraction methods. To this end, a loss function is introduced aiming to enlarge the error between predicted heat-maps and original heat-maps, while optimizing it using the gradient MI-FGSM [43]. While highly effective, this method's efficacy is limited to DeepFake models that use landmark extraction techniques.

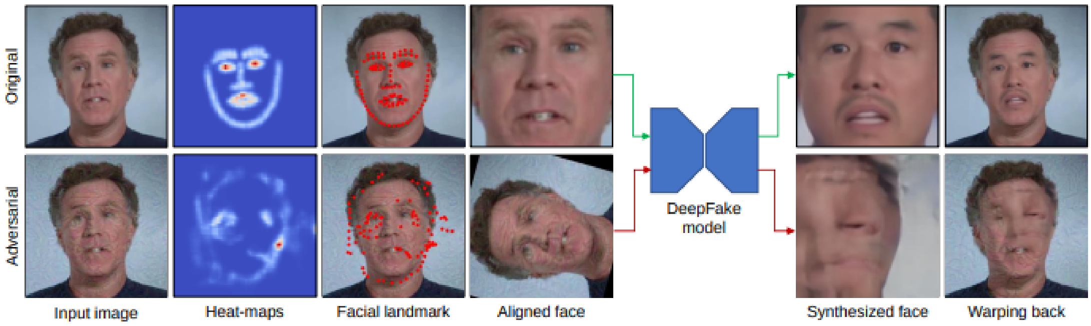


Figure 2.4: Illustration of the Landmark Breaker method. Taken from [6].

Inspired by the Distorting Attack [44], Segalis *et al.* [7] proposed the Oscillating GAN (OGAN), which is a novel attack optimized to be training-resistant, which introduces spatial-temporal distortions to the output of face swapping auto-encoders. In OGAN, a target distortion is applied to each frame of an image sequence (e.g., a video). Subsequently, each distorted image undergoes generation through a dedicated adversarial generator. When processed by a face-swapping autoencoder, these generated images manifest the introduced distortion. This process involves a simultaneous training of the generator and the face-swapping model through an iterative, alternating optimization technique. The optimization problem is solved by training the generator and the face-swapping model simultaneously using an iterative process of alternating optimization. An adapted version of the dfl-h128 autoencoder architecture was used for the adversarial generator, which is the autoencoder used in FaceSwap. This specific choice was made based on its association with the training process. A detailed description of the generator network's architecture is shown in Figure 2.5. While OGAN demonstrated superior efficiency compared to the Distorting Attack, its effectiveness remains primarily confined to face-swapping techniques.

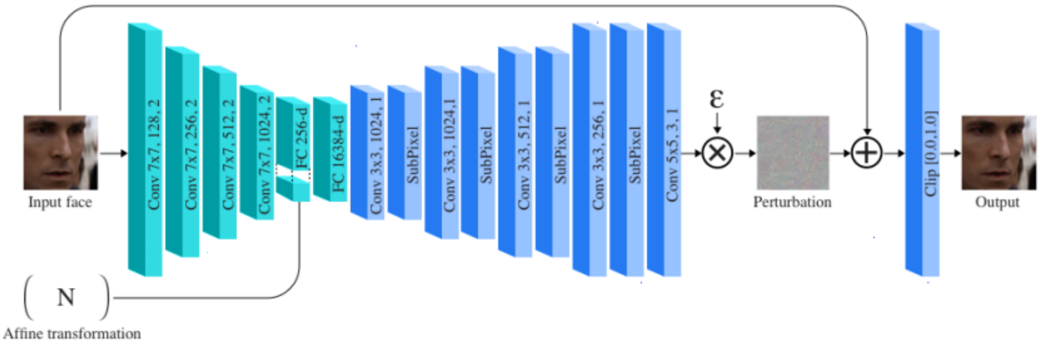


Figure 2.5: OGAN generator architecture. Taken from [7].

2.3.1.2 Gray-Box Attacks

A study made by Luochen Lv [8] proposed a watermark-based adversarial attack model. This model introduces imperceptible watermarks to images, leading to the generation of blurred images when processed by DeepFake models. This method comprises two key components: a watermark module and an attention module. The watermark module is used to embed watermarks to images so the images can defend against the StarGAN manipulation. Constructed with a fully convolutional network, this module employs a cascading structure of convolution followed by deconvolution. The convolutional structure extracts facial semantic information from the images, while the deconvolutional structure utilizes this information to generate watermarks and embed them at specific positions within the images. The input to the watermark module is the original image, and the output is the watermarked image. Simultaneously, the attention module is instrumental in guiding the training of the watermark. Its role is to ensure that the disruption caused by the watermarked image primarily affects the facial area. This is achieved by utilizing a face-detection network to identify facial positions and generate attention masks, as depicted by the red bounding box in Figure 2.6.

This approach proved more efficient and quicker than alternative watermark models. However, being tailored to a specific model limits its effectiveness in real-world scenarios.

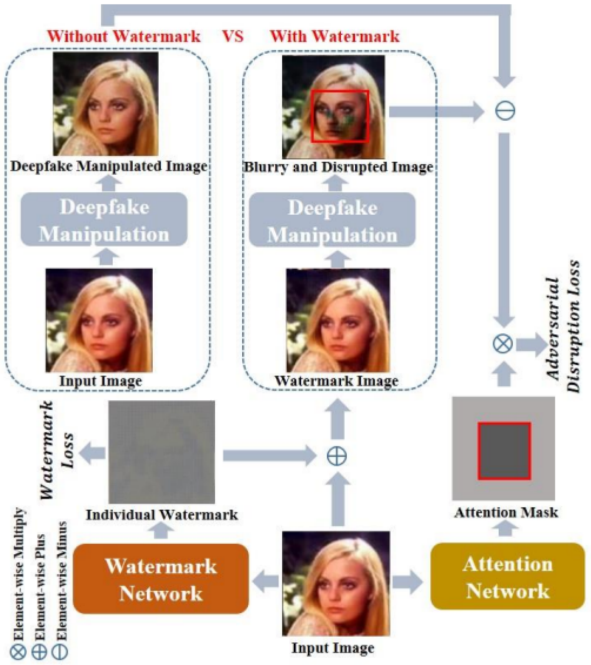


Figure 2.6: Illustration of the Smart Watermark algorithm. Taken from [8].

Wang *et al.* [1] proposed a novel framework, called DeepFake Disrupter, to defend against DeepFake with the help of the DeepFake detector. While existing disruption methods distort DeepFake outputs visually, experiments revealed that these altered samples could still deceive DeepFake detectors due to differences in decision logic between human perception and neural networks. Moreover, these disruption techniques often rely on time-consuming iteration-based adversarial attack algorithms like Iterative Fast Gradient Sign Method (I-FGSM) [39] and Projected Gradient Descent (PGD) [40] to determine perturbations for each data point. The proposed method addresses both issues encountered by these disruption methods. The pipeline consists of Perturbation Generator, DeepFake Generator, and DeepFake Detector, as illustrated in Figure 2.7.

The Perturbation Generator uses U-net [45] architectures, which are divided into two sections: an encoding section and a decoding section. The encoding section applies contraction blocks, which consists of a convolution and max-pooling layers to encode source inputs. The decoding section applies expansion blocks, which consists of a transpose convolution as well as normal convolutions. The DeepFake Generator is the DeepFake manipulation system chosen to disrupt, which can represent any model, given that this proposed model is designed to function universally across them. For the DeepFake Detector, backbones of several DeepFake detectors are employed, such as Xception [46] and ResNet [47], since this work is not focused on testing the effectiveness of the detection power.

The proposed pipeline can destroy the ability of DeepFake manipulation models both visually by the human eye and logically by DeepFake detectors. Though highly efficient, its training is specific to a particular model, limiting its transferability across various DeepFake models and reducing its overall effectiveness.

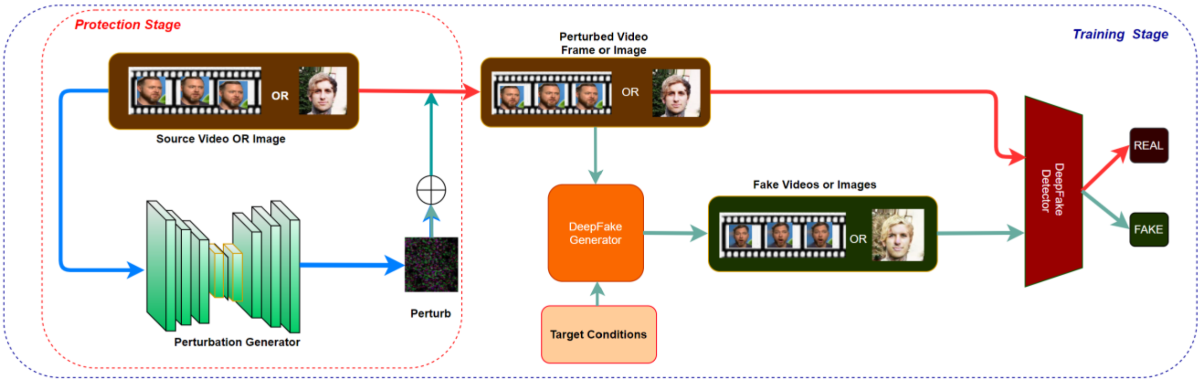


Figure 2.7: Overview of the DeepFake disrupter proposed by [1]. Taken from [1].

2.3.1.3 Black-Box Attacks

A study made by Qiu *et al.* [9] proposed a framework of an adversarial attack against DeepFakes called Cross-Domain and Model Adversarial Attack (CDMAA), which can expand the generalization of the generated adversarial examples in each domain of multiple models of DeepFake. This method can be applied to any gradient-based adversarial attack algorithms, such as I-FGSM [39], MI-FGSM [43] and others. Its primary focus is in obtaining a perturbation vector from gradients in multiple domains and models and then update the adversarial examples to ensure their ability to attack multiple models and domains.

An example of the CDMAA algorithm is illustrated in Figure 2.8. In this scenario, the I-FGSM method was used and four distinct DeepFake models contributed to generating a cross-model perturbation. Following the selection of these models, diverse domains within each model were leveraged to create a cross-domain perturbation. Afterward, the process involved gradient regularization and the application of the multiple gradient descent algorithm (MGDA) to derive the final perturbation.

Results show that the adversarial examples generated by CDMAA have high attack success rates and can effectively attack multiple DeepFake models at the same time. Since CDMAA needs to use the gradient-based adversarial attack algorithm, future work can focus on how to extend this framework to no-gradients-required adversarial attack algorithms, such as AdvGAN [48] or Boundary Attack [49]. The algorithm of the CDMAA method is illustrated in Figure 2.8.

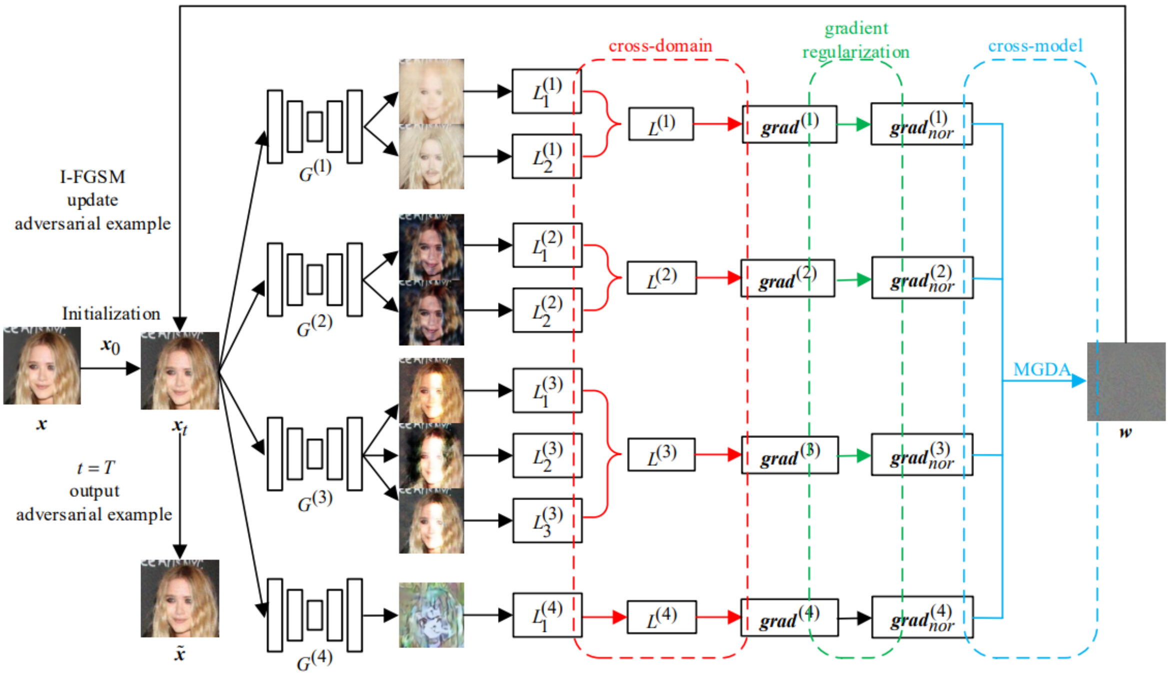


Figure 2.8: Illustration of the CDMAA method. Taken from [9].

Another study made by Dong *et al.* [10] designed the Transferable Cycle Adversary Generative Adversarial Network (TCA-GAN) to generate powerful adversarial examples against DeepFake systems. This method was developed to work in black-box settings, so a substitute model S was built for generalizable adversary generation, in order to simulate a DeepFake model. TCA-GAN employs a cyclic structure, comprising two generative models: G_P for generating transferable perturbations and G_R for removing adversarial perturbations. Illustrated in Figure 2.9, this framework establishes a cycle-consistent structure enabling both the addition and removal of adversarial perturbations, further enhancing the generalizability performance of the adversarial examples. Alongside the generators, two domain discriminators are constructed to distinguish adversarial examples and legitimate examples, D_A and D_L respectively. Lastly, to further enhance the transferability of the adversarial examples, a post-regularization is applied. This regularization can be seen as a distillation to obtain a second-best adversarial example towards the substitute model for better generalization.

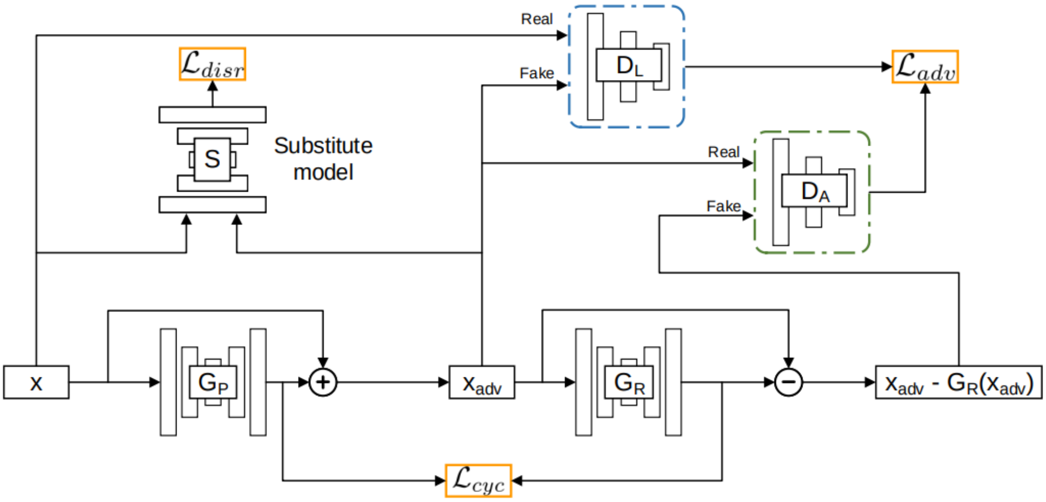


Figure 2.9: Illustration of the TCA-GAN algorithm. Taken from [10].

2.3.2 Image Reconstruction

As the sophistication of adversarial attack methods continues to progress, there is a concurrent evolution in defense strategies aiming to counter them. These defensive mechanisms, often termed as adversarial defenses or image reconstruction techniques, have undergone substantial diversification and advancement over time, leveraging advances in algorithms, deep learning, and computational power. Traditionally, adversarial defense research primarily concentrated on strengthening security in image classification tasks. However, with the rise of DeepFake technology, there has been a notable redirection. Recent efforts in this field are increasingly focusing on employing these defense strategies to defend against adversarial attacks targeted at DeepFakes, particularly through methods involving image reconstruction.

A study conducted by Zhang *et al.* [11], developed a defense mechanism for image classification tasks. This model consists of two components: an image reconstruction network $T(\cdot)$ and a feature extraction network $\varphi(\cdot)$. The image reconstruction network is a deep residual convolutional network that transforms an input perturbed imaged x^{adv} into an output reconstructed image $\hat{x}$ via the mapping $\hat{x} = T(x^{adv})$. To make the model more effective in defending against adversarial examples, this method uses a perceptual loss to measure the high-level feature differences between the reconstructed and clean images. To this end, a pretrained image classification network is used as a fixed feature extraction network for extracting high-level features from the reconstructed and clean images to calculate the loss function. An overview of the proposed defense model is shown in Figure 2.10.

This approach significantly minimizes the effect of adversarial alterations with minimal impact on prediction accuracy for clean images. Outperforming many existing defense strategies in testing, it exhibits superior generalization abilities. Moreover, this defense technique can seamlessly integrate with model-specific methods like adversarially trained models, offering a flexible and complementary enhancement.

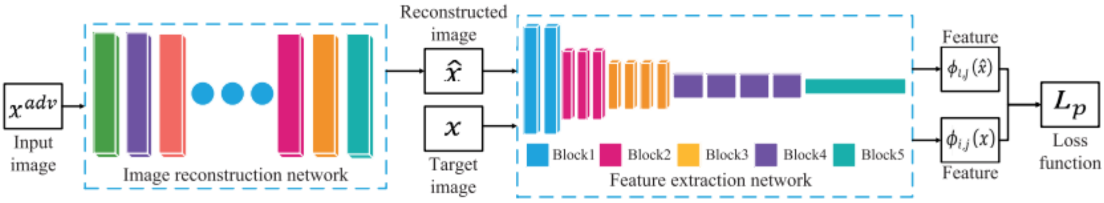


Figure 2.10: Defense mechanism proposed by [11]. Taken from [11].

To defend DeepFake against adversarial attacks, a method called Mask-guided Detection and Reconstruction (MagDR) was proposed by Chen *et al.* [12]. MagDR starts with defining a few criteria (e.g., SSIM, PSNR, etc.) that are sensitive to the abnormality of the outputs. Then, a mask-guided detector is trained to identify if the input image has been synthetically manipulated, by analyzing the output image. If yes, a reconstruction algorithm follows to eliminate the damage of the adversarial perturbations and recover the desired output.

This approach maintains multiple masks, leveraging them to offer supplementary insights in both detection and reconstruction phases. These masks are adaptable, acquired through individualized training processes, and correspond to specific segments of the human face. Using these masks as guides, the detector is split into two components: one targeting distortion and other focusing on inconsistencies, effectively pinpointing potentially compromised areas. For the reconstruction process, a pipeline is crafted comprising several modules, each equipped with customizable execution sequences and adjustable parameters. Subsequently, an adaptive optimization strategy is executed that subdues predefined criteria, ultimately generating the restored output. An illustration of the proposed defense model is shown in Figure 2.11.

The detection process involves two key components: a distortion detector and a consistency detector. The distortion detector compares alterations in attribute regions by measuring the variance between the region masks of the affected input (e.g., an adversarial image) and the affected output (e.g., a disrupted DeepFake). If this variance exceeds the desired conditional patch distance, it gets flagged as disrupted. While the distortion detector excels at identifying disruptions outside the specified attribute regions, it becomes less effective when the corruption spans the entire image. To address this, the consistency detector employs augmented versions of the affected outputs to compute a consistency score S_{cons}. If this score surpasses a predefined threshold, the input image is labeled as disrupted. These two processes are illustrated in Figure 2.11 (b) and (c), respectively.

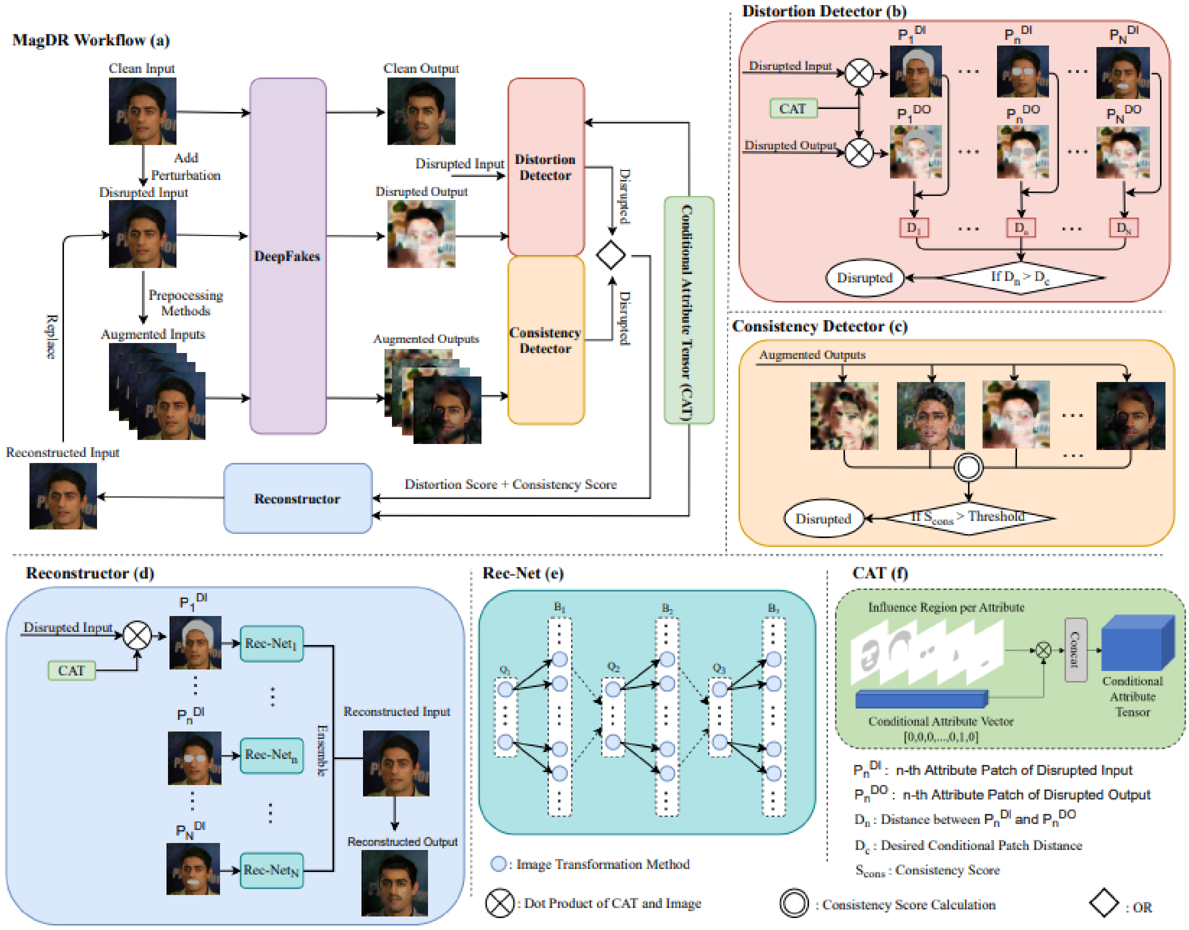


Figure 2.11: Illustration of MagDR. Taken from [12].

The reconstruction process aims to restore the accurate DeepFake output by mitigating the impact of introduced disturbances. In a formal sense, it is represented as the minimization of the distance between the original output and the reconstructed output:

$$\min D(G(x,c), T(\hat{x},c)), \tag{2.4}$$

where T(·) is an image transformation function. As can be seen in Figure 2.11 (d), this function has two components, which together form an effective pipeline that is difficult to bypass. First, the conditional region mask is applied to help obtain specific facial patches. Second, a multi-stage module Rec-Net is used, shown in Figure 2.11 (e), to enhance the image quality and simultaneously remove adversarial perturbations.

MagDR has been impressively successful in removing adversarial perturbations and possesses the capability to transfer seamlessly across diverse scenarios. It showcases its effectiveness in both white-box and black-box attacks, highlighting its adaptability and resilience.

3 Background Materials

This chapter aims to offer a comprehensive overview of the background materials crucial for this dissertation. It delves into essential components and pertinent research necessary to support and contextualize the study. Through this exploration, this chapter will provide a foundational understanding that aligns with the dissertation's scope and objectives.

3.1 Generative Adversarial Networks (GANs)

This section serves as an exploration into the application of GANs for studying and evaluating diverse methods developed to defend against the generation of DeepFake content. It aims to provide insight into the specific GAN models employed to assess and understand the efficacy of these proposed methods in countering the creation of DeepFake content.

To establish a foundational understanding of how GANs operate, a brief description of their architecture will be presented. Subsequently, the discussion will delve into the details of the DeepFake models — StarGAN [14], AttGAN [15], and AGGAN [16] - used to assess the robustness and applicability of DeepFake Disruption methods. This approach ensures a comprehensive overview, enabling a better grasp of both the underlying GAN mechanisms and the specific DeepFake models under consideration.

3.1.1 Architecture of GANs

GANs are an architecture composed of various neural networks, their objective is to replicate a data distribution in an unsupervised way. To achieve it, they are composed of two neural networks: a Generator and a Discriminator. The Generator (G) is in charge of creating new data samples replicating, but not copying, the origin data distribution; while the Discriminator (D) tries to distinguish real and generated data [13].

From a formal point of view, D estimates $p(y|x)$, that is, the probability of a label y given the sample x; while G generates a sample given a latent space z, which can be denoted as $G(z)$. An illustration of the architecture of a GAN model can be seen in Figure 3.1.

This process consists of both networks competing. While G tries to generate more realistic results, D improves its accuracy by detecting which samples are real and which are not. In this process, both competitors are synchronized, if G creates a better output, it will be more difficult for D to differentiate them. On the other hand, if D is more precise, it will be more difficult for G to fool D. This process is a minimax game in which D tries to maximize the accuracy and G tries to minimize it [13]. The formulation of the minimax game loss function can be denoted as:

$$\min_G \max_D L(D,G) = E_{x \sim p_r} \log[D(x)] + E_{z \sim p_z} \log[1 - D(G(x))], \tag{3.1}$$

where $x \sim p_r$ is the distribution of the real data and $z \sim p_z$ denotes the probability distribution of the latent space of G. $z \sim p_z$ is commonly a Gaussian or uniform noise that G uses to model new samples of data denoted as $G(z)$. D function is to differentiate between the real distribution $D(x)$ and the synthesized distribution $D(G(x))$.

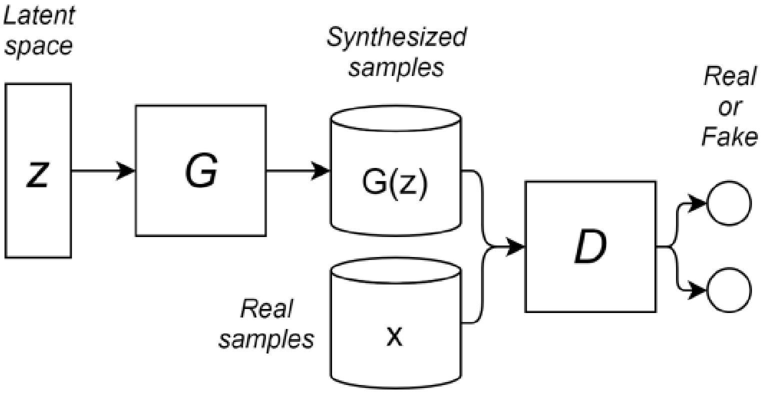


Figure 3.1: Architecture of a GAN model. Taken from [13].

3.1.2 StarGAN

StarGAN, proposed by Choi *et al.* [14], is a novel and scalable approach that can perform image-to-image translations for multiple domains using only a single model. Unlike many other image-to-image translation models that are designed for specific domains or attributes, StarGAN allows for seamless translation between various domains (such as different facial attributes like hair color, age, gender) using a unified architecture.

This model takes in training data of multiple domains, and learns the mappings between all available domains using only a single generator. Instead of learning a fixed translation (e.g., black-to-blond hair), the generator takes in as inputs both image and domain information, and learns to flexibly translate the image into the corresponding domain. To represent domain information, a label is used (e.g., binary or one-hot vector). During training, a target domain label is randomly generated and the model is trained to flexibly translate an input image into the target domain.

By doing so, the domain label can be controlled and the image can be translated into any desired domain at testing phase.

The goal is to train a single generator G that learns mappings among multiple domains. To achieve this, G is trained to translate an input image x into an output image y, conditioned on the target domain label c, $G(x, c) \rightarrow y$. The target domain is randomly generated so that G learns to flexibly translate the input image. Additionally, the discriminator produces probability distributions over both sources and domain labels, $D : x \rightarrow \{D_{src}(x), D_{cls}(x)\}$. Figure 3.2 illustrates the training process of StarGAN.

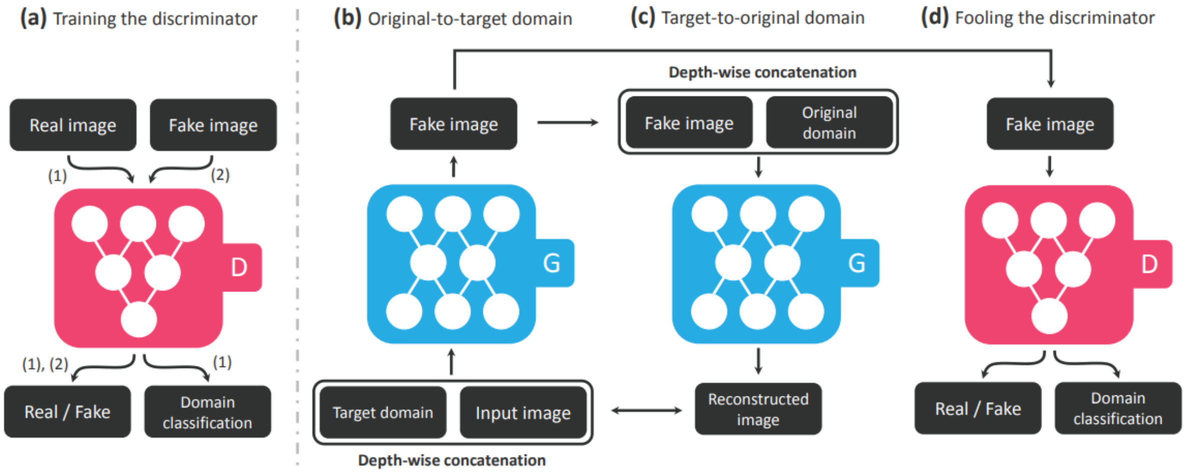


Figure 3.2: Overview of StarGAN. Taken from [14].

To make the generated images indistinguishable from real images, the following adversarial loss is adopted:

$$\mathcal{L}_{adv} = E_x[\log D_{src}(x)] + E_{x,c}[\log(1 - D_{src}(G(x, c)))], \tag{3.2}$$

where G generated an image $G(x, c)$ conditioned on both the input image x and the target domain label c, while D tries to distinguish between real and fake images. The generator G tries to minimize this objective, while the discrimination D tries to maximize it.

For the domain classification loss the objective is divided into two terms: a domain classification loss of real images used to optimize D, and a domain classification loss of fake images used to optimize G. In detail, the former is defined as:

$$\mathcal{L}^r_{cls} = E_{x,c'}[-\log D_{cls}(c'|x)], \tag{3.3}$$

where the term $D_{cls}(c'|x)$ represents a probability distribution over domain labels computed by D. By minimizing this objective, D learns to classify a real image x to its corresponding original domain c'. On the other hand, the loss function for the domain classification of fake images is defined as:

$$\mathcal{L}^f_{cls} = E_{x,c}[-\log D_{cls}(c|G(x, c))]. \tag{3.4}$$

In other words, G tries to minimize this objective to generate images that can be classified as the target domain c.

By minimizing the adversarial and classification losses, G is trained to generate images that are realistic and classified to its correct target domain. However, minimizing the losses does not guarantee that translated images preserve the content of its input images while changing only the domain-related part of the inputs. To alleviate this problem, a cycle consistency loss to the generator is applied, defined as:

$$\mathcal{L}_{rec} = E_{x,c,c'}[\|x - G(G(x, c), c')\|_1] \tag{3.5}$$

where G takes in the translated image $G(x, c)$ and the original domain label c' as input and tries to reconstruct the original image x. The L1 norm is applied as the reconstruction loss.

Finally, the objective functions to optimize G and D are written, respectively, as:

$$\mathcal{L}_D = -\mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}^r_{cls}, \tag{3.6}$$

$$\mathcal{L}_G = \mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}^f_{cls} + \lambda_{rec}\mathcal{L}_{rec}, \tag{3.7}$$

where λ_{cls} and λ_{rec} are hyper-parameters that control the relative importance of domain classification and reconstruction losses, respectively, compared to the adversarial loss.

3.1.3 AttGAN

He *et al.* [15] proposed AttGAN, which is a facial attribute editing that relies on encoder-decoder architecture. Previous research aimed to establish an attribute-independent latent representation, but this approach proved to be excessive as it limited the latent representation's capacity, potentially causing information loss. Consequently, it resulted in overly smooth and distorted generations. AttGAN takes a different approach by implementing an *attribute classification constraint* to the generated image. This constraint ensures the accurate alteration of desired attributes, essentially allowing users to "change what you want". Additionally, it introduces *reconstruction learning* to preserve attribute-excluding details, essentially enabling users to "only change what you want". Additionally, the *adversarial learning* is employed for visually realistic editing. These three components cooperate with each other forming an effective framework for high-quality facial attribute editing, as illustrated in Figure 3.3.

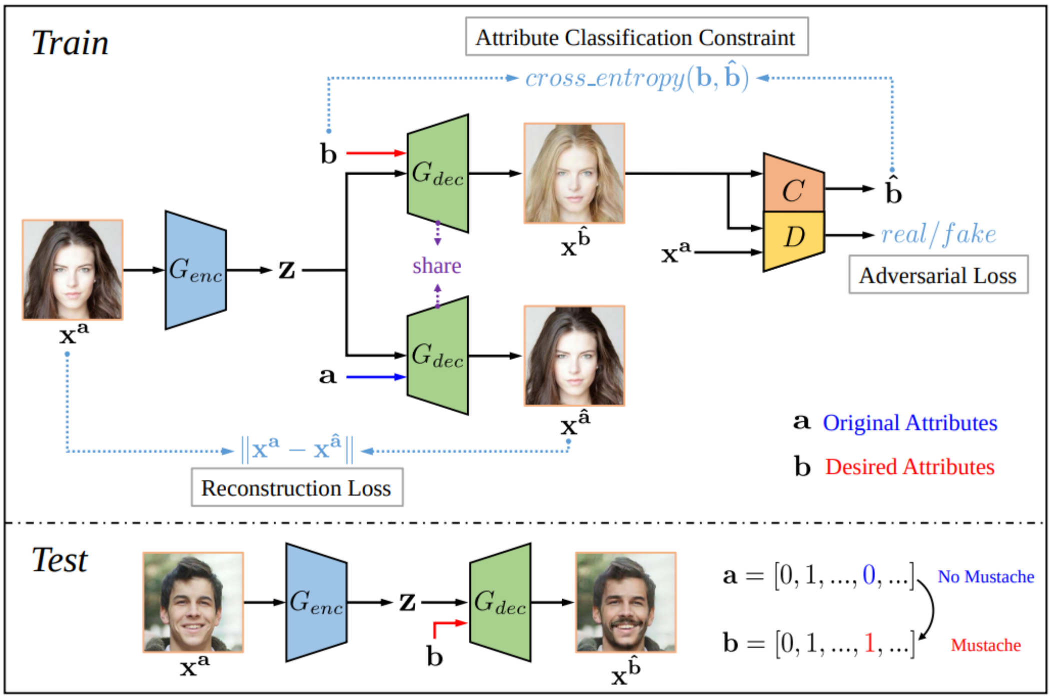


Figure 3.3: Overview of AttGAN. Taken from [15].

Given a face image x^a with n binary attributes $a = [a_1, ..., a_n]$, the encoder G_{enc} is used to encode x^a into the latent representation, denoted as:

$$z = G_{enc}(x^a). \tag{3.8}$$

Then the process of editing the attributes of x^a to another attributes $b = [b_1, ..., b_n]$ is achieved by decoding z conditioned on b, i.e.,

$$x^{\hat{b}} = G_{dec}(z, b), \tag{3.9}$$

where $x^{\hat{b}}$ is the edited image expected to own the attribute b. Thus the whole editing process is formulated as:

$$x^{\hat{b}} = G_{dec}(G_{enc}(x^a), b). \tag{3.10}$$

3.1.3.1 Attribute Classification Constraint

The editing on the given face image x^a is expected to produce a realistic image with attributes b. For this purpose, an attribute classifier is used to constraint the generated image $x^{\hat{b}}$ to correctly own the desired attributes, i.e., $C(x^{\hat{b}}) \rightarrow b$, formulated as follows:

$$\min_{G_{enc}, G_{dec}} \mathcal{L}_{cls_g} = E_{x^a \sim p_{data}, b \sim p_{attr}}[l_g(x^a, b)], \tag{3.11}$$

$$l_g(x^a, b) = \sum_{i=1}^{n} -b_i \log C_i(x^{\hat{b}}) - (1 - b_i) \log(1 - C_i(x^{\hat{b}})), \tag{3.12}$$

where p_{data} and p_{attr} indicate the distribution of real images and the distribution of attributes, $C_i(\hat{x^b})$ indicates the prediction of the i^{th}attribute, and $l_g(x^a, b)$ is the summation of binary cross entropy losses of all attributes.

The attribute classifier C is trained on the input images with their original attributes, by the following objective:

$$\min_{C} \mathcal{L}_{cls_c} = E_{x^a \sim p_{data}}[l_r(x^a, a)], \tag{3.13}$$

$$l_r(x^a, a) = \sum_{i=1}^{n} -a_i \log C_i(x^a) - (1 - a_i) \log(1 - C_i(x^a)). \tag{3.14}$$

3.1.3.2 Reconstruction Loss

An eligible attribute editing process should exclusively modify the desired attributes while leaving other details unaffected. For this purpose, *reconstruction learning* is introduced to make the latent representation z conserve enough information for the later recovery of the attribute-excluding details, and to enable the decoder G_{dec} to restore the attribute-excluding details from z. Specifically, for the given x^a, the generated image conditioned on its won attributes a, i.e.,

$$x^{\hat{a}} = G_{dec}(z, a) \tag{3.15}$$

should approximate x^a itself, i.e., $x^{\hat{a}} \to x^a$.

The learning objective is formulated as follows:

$$\min_{G_{enc}, G_{dec}} \mathcal{L}_{rec} = E_{x^a \sim p_{data}}[\|x^a - x^{\hat{a}}\|_1], \tag{3.16}$$

where the L2 loss is used, instead of the L1 loss, to suppress the blurriness.

3.1.3.3 Adversarial Loss

Adversarial learning is introduced between the generator (which includes the encoder and decoder) and the discriminator. Its purpose is to ensure that the generated image $x^{\hat{b}}$ appears visually realistic. Considering the foundations of WGAN [50], the adversarial losses for both the discriminator and generator are formulated as follows:

$$\min_{\|D\|_{L \leqslant 1}} \mathcal{L}_{adv_d} = -E_{x^a \sim p_{data}} D(x^a) + E_{x^a \sim p_{data}, b \sim p_{attr}} D(x^{\hat{b}}), \tag{3.17}$$

$$\min_{G_{enc}, G_{dec}} \mathcal{L}_{adv_g} = -E_{x^a \sim p_{data}, b \sim p_{attr}}[D(x^{\hat{b}})], \tag{3.18}$$

where D is the discriminator.

3.1.3.4 Overall Objective

By combining the attribute classification, the reconstruction loss, and the adversarial loss, a unified attribute GAN (AttGAN) is obtained, which can edit the desired attributes with the attribute-excluding details well preserved. The overall objective for the encoder and decoder is formulated as:

$$\min_{G_{enc},G_{dec}} \mathcal{L}_{enc,dec} = \lambda_1 \mathcal{L}_{rec} + \lambda_2 \mathcal{L}_{cls_g} + \mathcal{L}_{adv_g}, \tag{3.19}$$

and the objective for the discriminator and the attribute classifier is formulated as:

$$\min_{D,c} \mathcal{L}_{dis,cls} = \lambda_3 \mathcal{L}_{cls_c} + \mathcal{L}_{adv_d}, \tag{3.20}$$

where the discriminator and the attribute classifier share most layers, λ_1, λ_2, and λ_3 are the hyper-parameters for balancing the losses.

3.1.4 AttentionGAN

Tang *et al.* [16] proposed a novel Attention-Guided Generative Adversarial Network (AttentionGAN), also referenced as AGGAN, for the unpaired image-to-image translation task. The key advantage of AGGAN lies in its generator's ability to concentrate on the foreground of the target domain while effectively preserving the background from the source domain. Notably, these generators learn both foreground and background attentions. They employ foreground attention to choose the foreground regions from the generated output and utilize background attention to retain background information from the input image. This mechanism enables AttentionGAN to prioritize the most discriminative foreground elements while disregarding unwanted background aspects.

The devised generators come with an integrated attention module capable of separating discriminative semantic objects from undesired parts by generating an attention mask and a content mask, as illustrated in Figure 3.4. These masks are combined to derive the ultimate generation. Additionally, two innovative attention-guided discriminators have been crafted to focus on the attended foreground regions specifically.

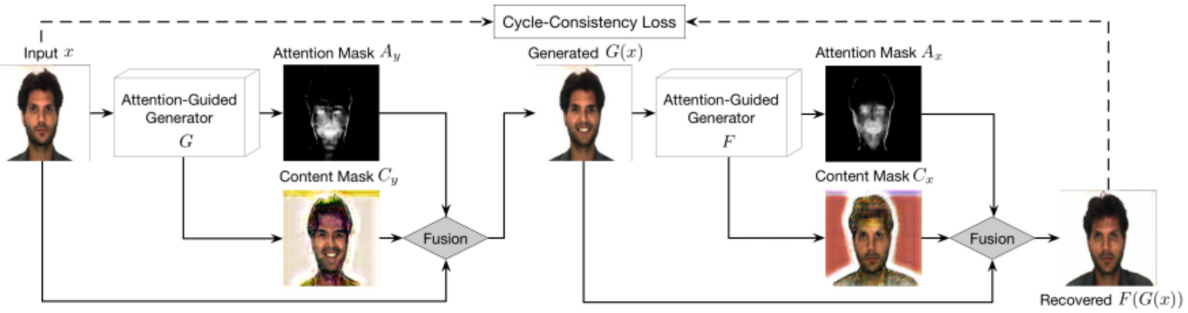


Figure 3.4: Overview of AttentionGAN. Taken from [16].

Furthermore, the proposed attention-guided generator and discriminator can be flexibly applied in other GANs to improve the multi-domain image-to-image translation tasks. In this dissertation, the AGGAN model will be integrated into StarGAN, involving the addition of attention mask and content mask generation to the StarGAN model.

3.1.4.1 Attention-Guided Generation

Let X and Y denote two different image domains. Generator G maps x from the source domain to the generated image $G(x)$ in the target domain Y and tries to fool the discriminator D_Y, whilst D_Y focuses on improving itself to be able to tell whether a sample is a generated sample or a real data sample.

A mapping is learned between domains X and Y via a generator with built-in attention mechanism, i.e., $G : x \rightarrow [A_y, C_y] \rightarrow G(x)$, where A_y and C_y are the attention mask and the content mask of image y, respectively; $G(x)$ is the generated image. The attention mask A_y defines a per pixel intensity specifying the contribution of each pixel of the content mask C_y) in the final rendered image. The higher intensity in the attention mask means a larger contribution to changing the expression.

The input of the generator is a three-channel image, and the outputs of the generator are an attention mask and a content mask. Thus, the final image $G(x)$ can be formulated as follows:

$$G(x) = C_y * A_y + x * (1 - A_y), \tag{3.21}$$

The attention mask A_y enables some specific areas where the domain changed to get more focus and applying it to the content mask C_y can generate images with clear dynamic area and unclear static area. The static area should be similar between the generated image and the original real image. Thus, the static area can be enhanced in the original image $x * (1 - A_y)$ and merged into $C_y * A_y$ to obtain the final result.

4 Methodology

This chapter aims to provide a comprehensive overview of the experimental work developed throughout this dissertation.

Section 4.1 will thoroughly explore the disruption methods under evaluation. This section will delve into the details of their algorithms and strategies, aiming to provide a comprehensive understanding of each method's approaches and methodologies.

Following this detailed examination, Section 4.2 will clarify the approach chosen to assess the resilience of the disruption methods to image reconstruction. This section will provide insights into the robustness of each disruption technique.

4.1 DeepFake Disruption Methods

In assessing the efficacy of present DeepFake Disruption models, three models were selected that excel in each category of attack: white-box, gray-box, and black box attacks.

4.1.1 White-Box Attack

For evaluating the ability of white-box attacks, the method proposed by Ruiz *et al.* [2] will be used. This approach presented a spread-spectrum adversarial attack that evades blur defenses. In essence, their method generates adversarial perturbations using blur, which cannot be subsequently eliminated using blur techniques, such as the median filter or Gaussian filter. To successfully disrupt a network in this scenario, they proposed a spread-spectrum evasion of blur defenses that transfers to different types of blur, using a modified I-FGSM update:

$$\tilde{x}_t = clip(\tilde{x}_{t-1} - \alpha sign[\nabla_{\tilde{x}} L(f_k(G(\tilde{x}_{t-1}), r)]) \tag{4.1}$$

where α is the step size, f_k is a blurring convolution operation and the constraint $\|\tilde{x} - x\|_\infty \leqslant \epsilon$ is enforced by the clip function. A set of K distinct blurring methods, each with varying magnitudes and types, is employed. In this work, the Average Smoothing filter and the Gaussian Smoothing filter were incorporated, resulting in $K = 2$. Initially, k is set to 1 and incremented with each iteration of the algorithm until it reaches K, signifying the total number of blur types and magnitudes. Then, k

is reset to 1. Since an adapted Projected Gradient Descent (PGD) [40] will be used, the disrupted image $\tilde{x}_0$ is initialized randomly inside the ϵ-ball around x and it will be updated using the I-FGSM update function.

Moreover, it is crucial to emphasize that the adversarial attack for a given sample image is designed for a specific attribute and a particular model. This implies that for each attribute and model combination, a distinct adversarially manipulated image will be generated.

4.1.2 Gray-Box Attack

To assess the efficacy of gray-box attacks, the method proposed by Wang *et al.* [17] will be used. This approach consists of adding perceptual-aware perturbations using the Lab color space, rather than operating on the RGB color space.

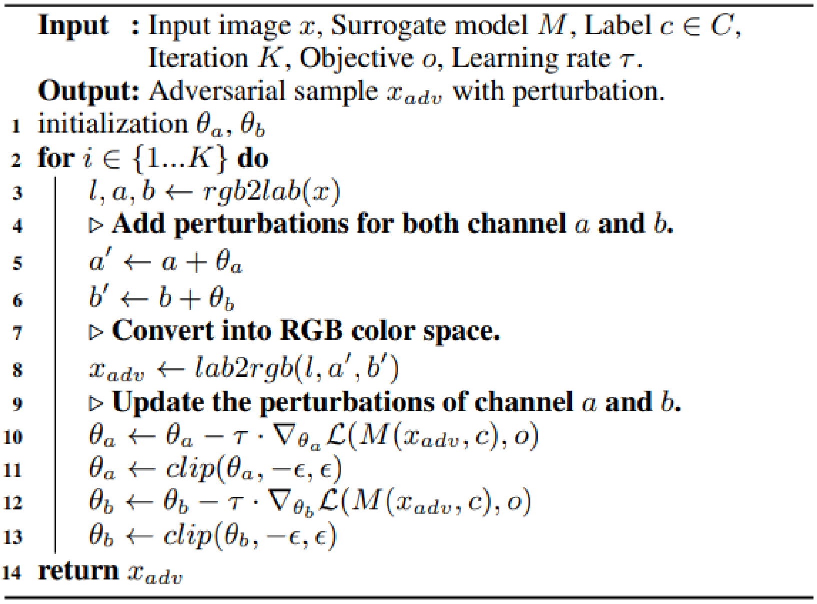
The Lab color space is a color model designed to approximate human vision and perception. Unlike the more commonly used RGB (Red, Green, Blue) and CMYK (Cyan, Magenta, Yellow, Black) color models, Lab represents colors based on perceptual uniformity, meaning that a change of the same amount in a color value should produce a similar perceptual change in color across the entire range. The Lab color space consists of three channels: a light channel L, and two color channels a and b. The L channel ranges from black (0) to white (100) representing the light, the a channel ranges from green (-128) to red ($+127$), and the b channels ranges from blue (-128) to yellow ($+127$).

Initially, the input image is converted from the RGB color space to the Lab color space to introduce perceptually uniform perturbations within the a and b channels. Then, these perturbations are refined by targeting the surrogate model M, which represents the DeepFake model, using the optimization-driven technique implemented in the C&W adversarial attack strategy. To accomplish this, the Adam optimizer was used with a learning rate set to $lr = 1 \times 10^{-4}$ and $\beta \in [0.9, 0.999]$.

For enhanced transferability when targeting various facial attributes, a distinct facial attribute label c is selected for each iteration. The objective function can be formulated as follows:

$$\min -\mathcal{L}(M(x_{adv}, c), o) \tag{4.2}$$

where $\mathcal{L}$ is the Mean Squared Error, x_{adv} is the adversarial image, and o is a regular translation image. This method's process is described in Figure 4.1.



```
Input  : Input image $x$, Surrogate model $M$, Label $c \in C$,
         Iteration $K$, Objective $o$, Learning rate $\tau$.
Output: Adversarial sample $x_{adv}$ with perturbation.
initialization $\theta_a$, $\theta_b$
for $i \in \{1...K\}$ do
    $l, a, b \leftarrow rgb2lab(x)$
    ▷ Add perturbations for both channel $a$ and $b$.
    $a' \leftarrow a + \theta_a$
    $b' \leftarrow b + \theta_b$
    ▷ Convert into RGB color space.
    $x_{adv} \leftarrow lab2rgb(l, a', b')$
    ▷ Update the perturbations of channel $a$ and $b$.
    $\theta_a \leftarrow \theta_a - \tau \cdot \nabla_{\theta_a} \mathcal{L}(M(x_{adv}, c), o)$
    $\theta_a \leftarrow clip(\theta_a, -\epsilon, \epsilon)$
    $\theta_b \leftarrow \theta_b - \tau \cdot \nabla_{\theta_b} \mathcal{L}(M(x_{adv}, c), o)$
    $\theta_b \leftarrow clip(\theta_b, -\epsilon, \epsilon)$
return $x_{adv}$
```

Figure 4.1: Algorithm description of the proposed method. Taken from [17].

4.1.3 Black-Box Attack

To evaluate the effectiveness of black-box attacks, the approach introduced by Huang *et al.* [18] will be employed. They designed a perturbation fusion strategy to alleviate the conflict of adversarial watermarks generated from different images and models in the attack process. Further, they analyzed the key problem of cross-model optimization and introduced an automatic step size tuning algorithm based on Tree-Structured Parzen Estimator (TPE) [51] to determine the overall optimization direction. To accomplish these, the CMUA-Watermark approach goes through three main steps:

- **Combating One Face Modification Model**

  In the initial phase, the process involves obtaining an adversarial perturbation for each DeepFake model. To achieve this, a batch of clean images $I_1 ... I_n$ is fed into the DeepFake model G to obtain the original outputs $G(I_1)...G(I_n)$. Then, an initial adversarial perturbation W is crafted to attack the clean images, resulting in the initial distorted outputs $G(I_1 + W)...G(I_n + W)$. Afterward, the Mean Squared Error (MSE) is used to measure the differences between the original outputs and the distorted outputs, as follows:

  $$\max_{W} \sum_{i=1}^{n} MSE(G(I_i), G(I_i + W)), s.t. \|W\|_\infty \leqslant \epsilon, \tag{4.3}$$

  where the ϵ is the upper bound magnitude of the adversarial watermark W. Lastly, the PGD method is used as the base attack method to update the adversarial perturbations at every attack iteration,

  $$I_{adv}^{r+1} = clip_{I,\epsilon}\{I_{adv}^{r} + asign(\nabla_I L(G(I_{adv}^{r}), G(I)))\}, \tag{4.4}$$

where I is the clean facial images, I_{adv}^{r} is the adversarial facial images in the rth iteration, a is the step size of the base attack, L is the loss function (MSE as formulated in Eq.(4.3), G is the face modification network under attack, and the operation *clip* limits the I_{adv} in the range $[I - \epsilon, I + \epsilon]$.

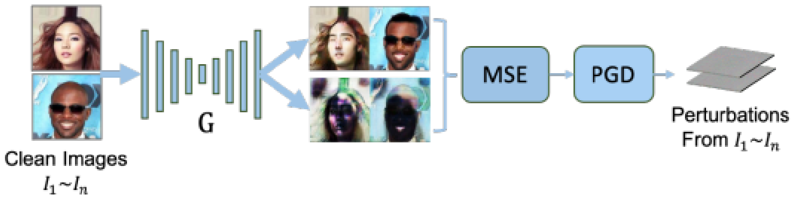


Figure 4.2: Detailed process of attacking one specific DeepFake model. Taken from [18].

- **Adversarial Perturbation Fusion**

After obtaining the adversarial perturbation for each DeepFake model, the next step is to employ a two-level perturbation fusion strategy. More precisely, when targeting a particular DeepFake model, an image-level fusion is implemented to average the signed gradients derived from a batch of facial images,

$$G_{avg} = \frac{\sum_{j}^{bs} sign(\nabla_{I_j} L(G(I_j^{adv}), G(I_j)))}{bs} \tag{4.5}$$

where bs is the batch size of facial images, and I_j^{adv} is the jth adversarial image of a batch. This operation will result in G_{avg} emphasizing common attributes shared among human faces rather than focusing on the distinctive features of a specific face. Then, PGD is used to generate the adversarial perturbation P_{avg} through G_{avg} as Eq. (4.5).

The model-level fusion consists of iteratively combining the perturbation P_{avg} of each DeepFake model to the W_{CMUA} in training. The initial W_{CMUA} is essentially the P_{avg} calculated from the first DeepFake model,

$$W_{CMUA}^{0} = P_{avg}^{0}, \tag{4.6}$$

$$W_{CMUA}^{t+1} = \alpha \cdot W_{CMUA}^{t} + (1 - \alpha) \cdot P_{avg}^{t}, \tag{4.7}$$

where α is a decay factor, P_{avg}^{t} is the average perturbation generated from the tth DeepFake model, and W_{CMUA}^{t} is the trained CMUA-Watermark after the tth attacked DeepFake model.

- **Automatic Step Size Tuning based on TPE**

The overall optimization direction is greatly influenced by the step sizes $a_1, ..., a_m$, and selecting them appropriately is a key problem for cross-model attacks. To resolve this, the TPE [51] algorithm is used for automatically searching the suitable step sizes for each DeepFake model. TPE is a hyper-parameter optimization method based on Sequential Model-Based Optimization (SMBO). In this case, the step sizes $a_1, ..., a_m$ are the input hyperparameters x and the success rate of the attack is the associated quality score y of TPE. The TPE uses $P(x|y)$ and $P(y)$ to model $P(y|x)$, and $p(x|y)$ is given by:

$$p(x|y) = \begin{cases} l(x), & if y < y*, \\ g(x), & if y \geqslant y*, \end{cases} \tag{4.8}$$

where $y*$ is determined by the historically best observation, $l(x)$ is the density formed with the observations $\{x^{(i)}\}$ such that the corresponding loss is lower than $y*$, and $g(x)$ is the density formed with the remaining observations. After modeling the $P(y|x)$, the search iterations continually seek improved step sizes by optimizing the Expected Improvement (EI) criterion in every search iteration, which is given by,

$$EI_{y*}(x) = \frac{\gamma y^* l(x) - l(x) \int_{-\infty}^{y*} p(y)\, dy}{\gamma l(x) + (1-\gamma) g(x)} \propto (\gamma + \frac{g(x)}{l(x)}(1-\gamma))^{-1} \tag{4.9}$$

where $\gamma = p(y < y*)$.

The overall pipeline of the CMUA approach is illustrated in Figure 4.3. Additionally, it is important to note that a pre-trained perturbation was used to evaluate this method, which was trained with StarGAN, AGGAN, AttGAN, and HiSD DeepFake models.

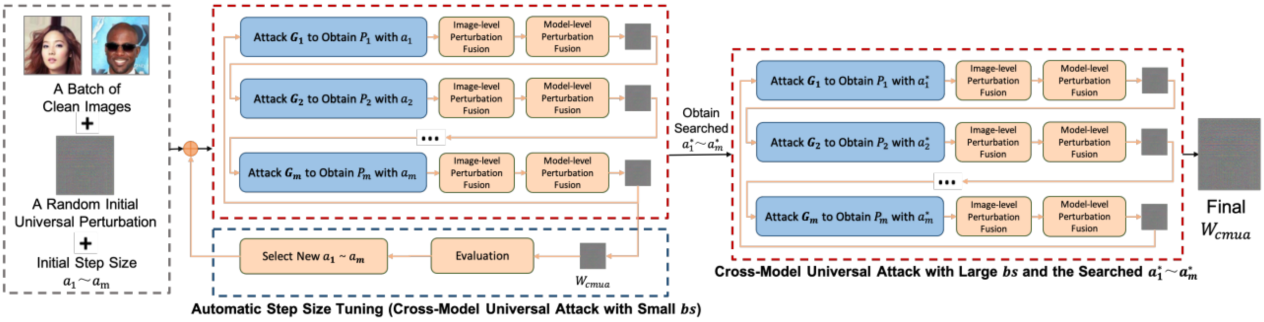

A Batch of Clean Images
+
A Random Initial Universal Perturbation
+
Initial Step Size
$a_1 \sim a_m$
Attack G_1 to Obtain P_1 with a_1
Attack G_2 to Obtain P_2 with a_2
Attack G_m to Obtain P_m with a_m
Image-level Perturbation Fusion
Model-level Perturbation Fusion
Select New $a_1 \sim a_m$
Evaluation
W_{cmua}
Automatic Step Size Tuning (Cross-Model Universal Attack with Small bs)
Obtain Searched $a_1^* \sim a_m^*$
Attack G_1 to Obtain P_1 with a_1^*
Attack G_2 to Obtain P_2 with a_2^*
Attack G_m to Obtain P_m with a_m^*
Cross-Model Universal Attack with Large bs and the Searched $a_1^* \sim a_m^*$
Final W_{cmua}

Figure 4.3: Overall pipeline of the CMUA approach. Taken from [18].

4.2 Disruption Resilience to Image Reconstruction

Assessing the resilience of the examined DeepFake Disruption models involves evaluating their capability to withstand image reconstruction, a process that removes the adversarial perturbations incorporated into the image.

To this end, the proposed method by Mustafa *et al.* [19] will be used, an image super-resolution (SR) technique, which consists of selectively adding high-frequency components to an image and removing noisy perturbations added by the adversary. The proposed approach has two components, which together form a non-differentiable pipeline that is difficult to bypass. Initially, wavelet denoising is employed to mitigate any noise patterns. The main component of this approach involves the super-resolution operation, which augments pixel resolution while concurrently eliminating adversarial patterns. Experiments show that employing image super-resolution alone adequately restores classifier convictions toward accurate categories. Nonetheless, the second step augments robustness, as it involves a non-differentiable denoising operation. An illustration of this method can be seen in Figure 4.4.

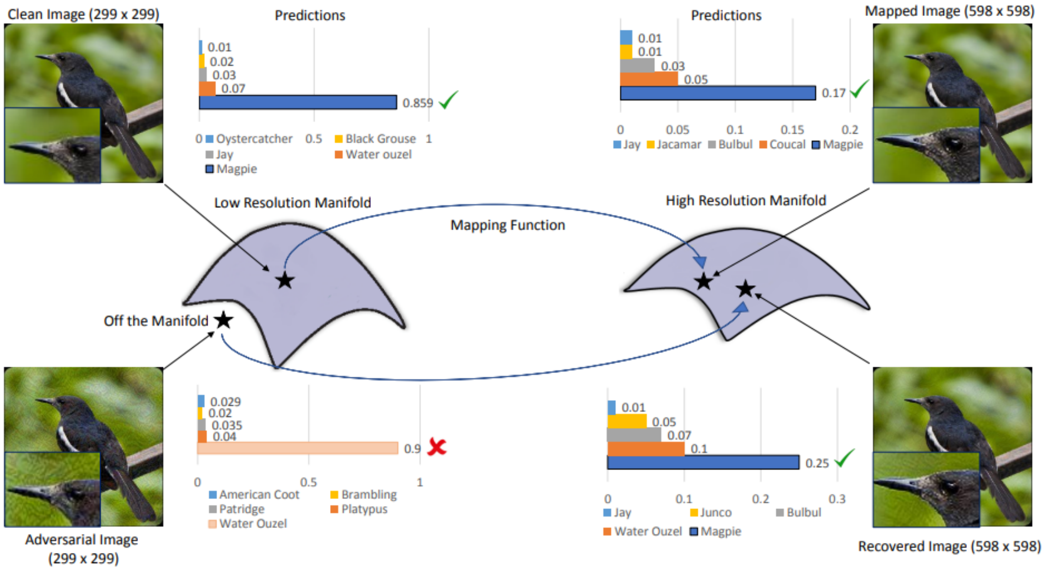


Figure 4.4: Super-resolution method proposed by Mustafa *et al.*. Taken from [19].

4.2.1 Super-resolution Network

As these perturbations typically involve high-frequency details, a super-resolution network is employed, employing residual learning specifically to target such details. These intricacies are incorporated into the low-resolution inputs within each residual block, culminating in the production of a high-quality, super-resolved image. The network considered in this approach is the Enhanced Deep Super-Resolution (EDSR) [20] network, with the architecture illustrated in Figure 4.5.

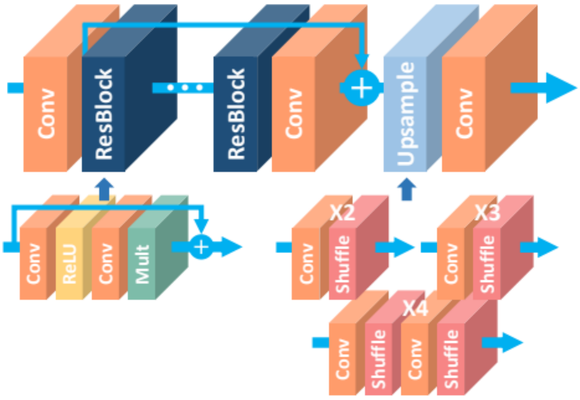


Figure 4.5: The architecture of EDSR network. Taken from [20].

By analyzing the frequency-domain spectrum of the clean, adversarial and recovered images in Figure 4.6, two primary advantages become evident: firstly, the newly added high-frequency patterns smooth the frequency response of the image, and secondly, the super-resolution destroys the adversarial patterns intended to deceive the model.

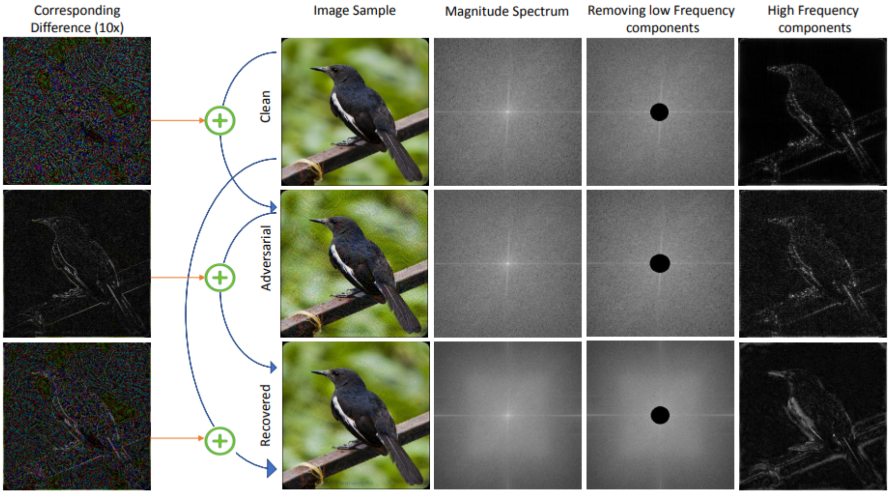

Corresponding Difference (10x)
Image Sample
Magnitude Spectrum
Removing low Frequency components
High Frequency components
Clean
Adversarial
Recovered

Figure 4.6: Effect of super-resolution on the frequency distribution of a sample image. Taken from [19].

4.2.2 Wavelet Denoising

As adversarial attacks uniformly introduce precisely crafted perturbations in the form of noise to an image, a proficient image denoising technique holds substantial potential in alleviating the impact of these perturbations, potentially even eliminating them entirely.

Removing noise from images, whether in the spatial or frequency domain, often leads to a loss of textural details. This loss undermines the objective of achieving denoised images that resemble clean, detailed images. To address this challenge, this method employs the Discrete Wavelet

Transform (DWT). Unlike traditional image smoothing methods that eliminate higher frequency components, DWTs of real-world images have large coefficients linked to important image features. By setting a threshold on smaller coefficients, DWT enables the removal of noise while preserving significant image characteristics.

The thresholding parameter dictates the effectiveness with which the wavelet coefficients are reduced, facilitating the removal of adversarial noise from an image. In this method, a soft thresholding technique is employed, as it reduces abrupt sharp changes, and it can be formulated as:

$$D(\hat{x}, t) = max(0, 1 - \frac{t}{|\hat{x}|})\hat{x}, \tag{4.10}$$

where each coefficient $\hat{x}$ is individually compared to a threshold value t. Selecting the ideal threshold value t, stands as the fundamental challenge in wavelet denoising. A significantly large threshold disregards larger wavelets, leading to an excessively smoothed image. In contrast, a small threshold permits noisy wavelets to persist, thereby failing to generate a denoised image upon reconstruction. *BayesShrink* [52] operates as an efficient wavelet shrinkage method that applies distinct thresholds to individual wavelet sub-bands, taking into account Gaussian noise characteristics. Let $\hat{x}_{adv} = \hat{x}_c + \hat{\rho}$ be the wavelet transform of an adversarial image, since $\hat{x}_c$ and $\hat{\rho}$ are mutually independent, the variances $\sigma^2_{x_{adv}}$, $\sigma^2_{x_c}$ and σ^2_{ρ} of $\hat{x}_{adv}$, $\hat{x}_c$, $\hat{\rho}$, respectively, follow: $\sigma^2_{x_{adv}} = \sigma^2_{x_c} + \sigma^2_{\rho}$. A wavelet sub-band variance for an adversarial image is estimated as:

$$\sigma^2_{x_{adv}} = \frac{1}{M} \sum_{m=1}^{M} W^2_m, \tag{4.11}$$

where W^2_m are the sub-band wavelets and M is the total number of wavelet coefficients in a sub-band. The threshold value for *BayesShrink* soft-thresholding is given as:

$$t_{bs} = \begin{cases} \sigma^2_{\rho}/\sigma_{x_c} & \text{if } \sigma^2_{\rho} < \sigma^2_{x_{adv}} \\ \max(|W_m|) & \text{otherwise.} \end{cases} \tag{4.12}$$

5 Results and Discussion

This chapter is divided into two main parts. The first part, consisting of sections 5.1, 5.2, 5.3, focuses on technical aspects related to the datasets, evaluation metrics and implementation details. The second part, section 5.4, provides a detailed examination of the experimental results, by covering the Disruption and Reconstruction results.

5.1 Datasets

All our experiments are conducted on a popular face dataset CelebFaces Attributes (CelebA) [53]. CelebA is a large-scale face attributes dataset with more than 200K celebrity images, each with 40 attribute annotations. The images in this dataset cover large pose variations and background clutter. CelebA has been widely used for creating fake faces (e.g., attribute editing, face reenactment, identity swap) via various GANs. In this dissertation, all the facial images were cropped to 256×256.

5.2 Evaluation Metrics

To thoroughly evaluate the methods under analysis, it is crucial to conduct a comparison within the resulting images of each method. To achieve this, three distinct metrics were employed, each assessing different aspects of images. These metrics include:

- **Mean Squared Error (MSE):** this metric takes the squared difference between corresponding pixels for every pixel in the images, sums up these squared differences, and then divides them by the total number of pixels. The result provides a measure of the average squared "error" or discrepancy between the pixel values of the two images. Lower MSE values indicate higher similarity between the images. The formulation is as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2, \tag{5.1}$$

where N is the number of pixels, and x_i and y_i represent the ith pixel of the two images being compared.

- **Structural Similarity Index (SSIM):** SSIM evaluates the structural similarities between two images by considering luminance, contrast, and structure. It offers a comprehensive measure of perceptual similarity. The formula for SSIM is as follows:

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{5.2}$$

  where x and y are the compared images; μ_x and μ_y are the averages of x and y, respectively; σ_x and σ_y are the standard deviations of x and y; σ_{xy} is the covariance of x and y; and c_1 and c_2 are constants to avoid instability when the denominator is close to zero. The SSIM index ranges from -1 to 1, where 1 indicates perfect similarity, 0 indicates no similarity, and -1 indicates complete dissimilarity. Higher SSIM values correspond to greater perceived similarity between the images.

- **Peak Signal-to-Noise Ratio (PSNR):** PSNR quantifies the quality of an image by comparing it to a reference image. It measures the ratio of the maximum possible power of a signal (image) to the power of the noise that affects the fidelity of the image. The formula for calculating PSNR is as follows:

$$PSNR = 10 \cdot \log_{10}(\frac{MAX^2}{MSE}), \tag{5.3}$$

  where MAX is the maximum possible pixel value of the image (usually 255 for 8-bit images). A higher PSNR value indicates a lower level of noise or distortion in the reconstructed image, suggesting better quality. Conversely, a lower PSNR value indicates higher distortion or a greater mismatch between the original and reconstructed images.

5.3 Implementation Details

To assess the transferability performance of individual DeepFake Disruption methods, the code was customized for each DeepFake model. With the exception of the CMUA-Watermark method, a perturbation was trained for each DeepFake model within each DeepFake Disruption method. Then, the transferability of the acquired perturbations was tested, by generating synthesized images with each DeepFake model. As an example, a perturbation using the approach detailed in [17] was trained to target the StarGAN model and subsequently tested the resulting adversarial image on StarGAN itself, AttGAN, and AGGAN. This analysis helps determine the extent to which the perturbation can be generalized across various DeepFake models. In the case of the CMUA-Watermark method, a single perturbation is trained for all DeepFake models, simplifying the testing process as it only requires evaluating one perturbation on each individual model.

To evaluate the resilience against image reconstruction, the reconstruction technique outlined in [19] was used, along with additional straightforward image processing methods. This process involved removing the perturbation and subsequently testing robustness by generating synthesized images for each DeepFake model using the reconstructed images as input.

The evaluation was conducted with 1999 images, and it considered five key attributes: black hair, blonde hair, brown hair, gender, and age. It is also important to note that pre-trained models were used for every DeepFake model and a pre-trained perturbation was used for the CMUA-Watermark method.

5.4 Validation Results

In this section, the outcomes will be categorized into two distinct groups: disruption results and reconstruction results. The disruption results will assess the effectiveness of the evaluated disruption methods and their transferability, whereas the reconstruction results will demonstrate their resilience to image reconstruction methods.

5.4.1 Disruption Results

5.4.1.1 White-Box Attack

To evaluate the effectiveness of the white-box attack methodology discussed in [2], a series of experiments were conducted to assess the transferability of perturbations across different DeepFake models. Initially, perturbations were trained for the StarGAN model, and their impact was tested on various models, including StarGAN itself, AttGAN, and AGGAN. The outcomes of this analysis, comparing the original and disrupted DeepFake images, are presented in Table 5.1. Subsequently, the process was replicated, with training conducted on the AttGAN model, and the findings are presented in Table 5.2. Finally, the training was carried out on the AGGAN model, and the results from this iteration are documented in Table 5.3. These tables offer a comprehensive comparison of original and disrupted DeepFake images across different models, illuminating the transferability aspects of the white-box attack approach.

StarGAN			
	StarGAN	**AttGAN**	**AGGAN**
MSE ↑	0.38324	0.00316	0.20852
SSIM ↓	0.52467	0.97265	0.71933
PSNR ↓	10.47293	33.14266	14.88754

Table 5.1: Comparison between authentic DeepFake images and perturbed DeepFake images achieved through adversarial perturbations trained on StarGAN. Using the disruption method referenced in [2].

AttGAN			
	StarGAN	**AttGAN**	**AGGAN**
MSE ↑	0.02753	0.10496	0.02435
SSIM ↓	0.82187	0.81959	0.87842
PSNR ↓	22.31103	17.73479	26.08475

Table 5.2: Comparison between authentic DeepFake images and perturbed DeepFake images achieved through adversarial perturbations trained on AttGAN. Using the disruption method referenced in [2].

AGGAN			
	StarGAN	**AttGAN**	**AGGAN**
MSE ↑	0.84873	0.00077	0.47262
SSIM ↓	0.30009	0.98919	0.66557
PSNR ↓	7.01917	38.95905	0.99850

Table 5.3: Comparison between authentic DeepFake images and perturbed DeepFake images achieved through adversarial perturbations trained on AGGAN. Using the disruption method referenced in [2].

The perturbation trained on StarGAN, as seen in Table 5.1, demonstrates commendable performance when tested on both StarGAN and AGGAN, despite a slight decline on AGGAN. However, its effectiveness significantly diminishes when applied to the AttGAN model. In contrast, the perturbation trained on AttGAN, highlighted in Table 5.2, excels in testing within the AttGAN model. Nevertheless, its effectiveness experiences a notable decline when transferred to testing on StarGAN and AGGAN. Moving on to the perturbation trained on AGGAN, detailed in Table 5.3, it exhibits formidable performance during testing on StarGAN, surpassing its efficacy even within the original AGGAN model. However, a significant drop in performance is observed when this perturbation is tested within the AttGAN model.

These findings provide insights into the nuanced dynamics of perturbation transferability among DeepFake models, revealing variations in performance across different architectures. This method's transferability is constrained when applied to diverse models. Notably, AttGAN stands out for its heightened resilience to adversarial perturbations, resulting in reduced overall transferability. When both trained and tested on AttGAN, the method exhibits a mean squared error (MSE) distance of 0.10496, while other models show MSE distances exceeding 0.3, suggesting higher resilience to adversarial attacks.

Additionally, perturbations trained on AttGAN have a slight transferability to other models, while perturbations from other models lack the same efficacy when tested on AttGAN. This highlights a model-specific quality in perturbation transferability, with AttGAN displaying distinctive resilience to DeepFake Disruption.

5.4.1.2 Gray-Box Attack

In evaluating the gray-box attack approach outlined in [17], the systematic assessment involved generating perturbations for each DeepFake model and testing them across the entire range of models. The initial phase involved training the perturbations for StarGAN, and subsequent evaluations were conducted by generating fake images using StarGAN, AttGAN, and AGGAN. The comparative analysis between original and disrupted DeepFake images is presented in Table 5.4. Subsequently, this process was replicated to train perturbations on AttGAN, and the outcomes of this process are detailed in Table 5.5. Continuing the examination, the perturbations were trained with AGGAN, and the detailed results of this process are provided in Table 5.6. These tables provide a comprehensive comparison between original and disrupted DeepFake images across different models, shedding light on the transferability of the gray-box attack approach.

StarGAN			
	StarGAN	**AttGAN**	**AGGAN**
MSE ↑	0.99536	6.85176e-06	0.83991
SSIM ↓	0.36214	0.99826	0.62673
PSNR ↓	6.43988	59.76377	10.25829

Table 5.4: Comparison between authentic DeepFake images and perturbed DeepFake images achieved through adversarial perturbations trained on StarGAN. Using the disruption method referenced in [17].

AttGAN			
	StarGAN	**AttGAN**	**AGGAN**
MSE ↑	0.07450	0.29962	0.05970
SSIM ↓	0.66884	0.64576	0.76863
PSNR ↓	17.68660	12.45820	21.86497

Table 5.5: Comparison between authentic DeepFake images and perturbed DeepFake images achieved through adversarial perturbations trained on AttGAN. Using the disruption method referenced in [17].

AGGAN			
	StarGAN	**AttGAN**	**AGGAN**
MSE ↑	0.64538	8.79295e-06	1.36311
SSIM ↓	0.50649	0.99976	0.59761
PSNR ↓	8.47144	58.81185	9.92502

Table 5.6: Comparison between authentic DeepFake images and perturbed DeepFake images achieved through adversarial perturbations trained on AGGAN. Using the disruption method referenced in [17].

The results presented in Table 5.4 demonstrate that the perturbation, specifically trained with the StarGAN model, performs well during testing on both the StarGAN and AGGAN models. However, its effectiveness significantly diminishes when applied to tests involving the AttGAN model. Notably, in Table 5.5, the perturbation trained on AttGAN performs well in AttGAN testing, but its effectiveness diminishes when tested on StarGAN and AGGAN. When the perturbation is trained on AGGAN, as seen in Table 5.6, it demonstrates formidable performance during testing on AGGAN, with a slight decrease observed when tested on StarGAN. However, a substantial decline in performance becomes apparent when subjecting this perturbation to testing within the AttGAN model.

Drawing conclusions from these insights, it becomes apparent that the transferability of this method encounters limitations when applied across a spectrum of models. Once again, the AttGAN method emerges as noteworthy for its heightened resilience to adversarial perturbations compared to other models, contributing to an overall reduced level of transferability. It is important to highlight that although perturbations trained on AttGAN may exhibit slight transferability to other models, the reverse is not true. Perturbations generated on other models do not exhibit any notable capacity to transfer to AttGAN.

5.4.1.3 Black-Box Attack

To assess the transferability of the black-box attack approach outlined in [18], a cross-model perturbation was generated, simultaneously trained on all DeepFake models, and subsequently tested its impact across the entire spectrum of models. The outcomes of this evaluation, illustrating the comparison between original and disrupted DeepFake images, are displayed in Table 5.7. The results will shed light on the effectiveness and transferability of the black-box attack approach in a cross-model context.

	StarGAN	**AttGAN**	**AGGAN**
MSE ↑	0.20119	0.05986	0.12635
SSIM ↓	0.59981	0.74296	0.73689
PSNR ↓	13.20751	18.87833	17.82545

Table 5.7: Comparison between authentic DeepFake images and perturbed DeepFake images achieved through a cross-model adversarial perturbation. Using the disruption method referenced in [18].

Analyzing the data presented in Table 5.7, it becomes apparent that the cross-model adversarial perturbation successfully disrupted both StarGAN and AGGAN. Notably, AttGAN exhibited a higher degree of resilience to perturbations, showcasing its consistent ability to withstand adversarial disruptions compared to other models.

5.4.1.4 Preliminary Conclusions

The white-box and gray-box attacks exhibited limited transferability, experiencing a decrease in disruption efficacy when applied to a model different from the one on which they were trained. In contrast, the black-box attack stands out as the only one demonstrating a strong ability to transfer, as anticipated. Given its nature as a cross-model perturbation crafted for a wide range of DeepFake models, the black-box attack proves effective across various model scenarios.

Additionally, it is observable that the disruption values of the black-box attack are lower when compared to disruptions achieved through other methods. This decline is expected, given that the cross-model perturbation was not tailored for any specific model but instead designed to have a broad impact across different models.

The visual outcomes of this evaluation are displayed in Appendix A. The results of the white-box, gray-box, and black-box attacks can be observed in A.1, A.2, and A.3, respectively. Examining these figures further validates the lack of transferability of the white-box and gray-box attack, and the ability of the black-box attack to transfer across various DeepFake models.

Moreover, the robustness of the AttGAN model against DeepFake disruption is apparent. This resilience can be attributed to its architecture, which incorporates an attribute classification constraint. This constraint ensures precise modifications exclusively to the targeted attributes, limiting alterations to those specific areas. As a consequence, adversarial attacks affect only these designated regions. This distinction becomes evident when contrasting AttGAN with the StarGAN and AGGAN models. Examining the disrupted images reveals that StarGAN and AGGAN exhibit disruptions across the entire image, whereas AttGAN restricts disruptions solely to the attribute area. Additionally, a closer analysis of DeepFake images from StarGAN, particularly those derived from the original images, exposes slight changes beyond the manipulated attribute. This underscores the impact of not restricting alterations to the attribute area. Although AGGAN utilizes attention masks to better confine changes to the attribute region compared to StarGAN, it falls short of achieving ideal performance.

5.4.2 Reconstruction Results

This section focuses on the reconstruction evaluation of each DeepFake Disruption method. Alongside the image reconstruction approach referenced in [19], noise mitigation techniques were incorporated such as the median filter and the concurrent upsampling and downsampling to assess the robustness of each method.

- **Median Filter:** The median filter, an image processing method, reduces noise by replacing each pixel's value with the median value within a designated neighborhood. This approach excels at preserving edges and intricate details in the image while effectively mitigating unwanted distortions.

- **Concurrent Upsampling and Downsampling:** this is a technique that involves simultaneously increasing and then restoring the resolution of an image. This process helps in reducing noise, especially fine details or high-frequency components, while maintaining the overall structure and content of the image. The upsampling phase acts as a smoothing step, suppressing noise, and the subsequent downsampling restores the image to its original size, resulting in a denoised version.

In the tables that follow, each one presents a comparison between genuine DeepFake images and reconstructed adversarial DeepFake images. Each table consists of four lines, with the adversarial line exclusively featuring the original adversarial images. In contrast, the reconstruction line integrates reconstructed adversarial images using the method outlined in [19]. The blur line employs a median filter to reconstruct adversarial images, while the resize line involves the upsampling and downsampling technique applied to the adversarial images.

These three distinct techniques collectively provide a comprehensive exploration of various reconstruction methods, offering insights into the robustness of each Disruption approach.

5.4.2.1 White-Box Attack

The reconstruction results of the white-box attack methodology, as discussed in [2], are detailed in tables 5.8, 5.9, and 5.10. Specifically, table 5.8 displays the results for the adversarial perturbation trained on StarGAN and evaluated on StarGAN. Likewise, table 5.9 showcases the results for the adversarial perturbation trained on AttGAN and tested on AttGAN. Finally, table 5.10 illustrates the outcomes for the adversarial perturbation trained on AGGAN and tested on AGGAN.

StarGAN - StarGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.38328	0.52466	10.47274
Reconstruction	0.07891	0.83332	17.86248
Blur	0.19371	0.68488	13.56543
Resize	0.29532	0.62348	11.66185

Table 5.8: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained and tested on StarGAN. Using the disruption method referenced in [2].

AttGAN - AttGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.10495	0.81965	17.73554
Reconstruction	0.07517	0.84870	19.27719
Blur	0.10943	0.80531	17.34337
Resize	0.11054	0.80614	17.26456

Table 5.9: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained and tested on AttGAN. Using the disruption method referenced in [2].

AGGAN - AGGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.47363	0.66550	12.26478
Reconstruction	0.13215	0.83246	18.36240
Blur	0.11339	0.76236	18.00875
Resize	0.34646	0.71282	13.47985

Table 5.10: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained and tested on AGGAN. Using the disruption method referenced in [2].

The results outlined in Table 5.8 indicate that the perturbation, exclusively trained on the StarGAN model, is easily removable. It shows a heightened sensitivity to the methodology outlined in [19]. Although still removable with the median filter and the upsampling and downsampling method, it exhibits a slightly higher level of resilience against these techniques. Moving to Table 5.9, the perturbation trained on AttGAN stands out for its high resilience to image reconstruction techniques. It shows a marginal decrease in response to the methodology in [19], and notably intensifies disruption when subjected to the median filter and upsampling and downsampling techniques. In Table 5.10, when the perturbation is trained on AGGAN, similar to the StarGAN model, it demonstrates ease of removal. It experiences a significant decrease in disruptive efficacy when the technique outlined in [19] and the median filter are applied. However, it exhibits higher resilience to the upsampling and downsampling technique. It becomes evident that this method lacks robustness against image reconstruction techniques. In contrast, AttGAN demonstrates notable resilience to these techniques, attributable to its unique characteristics. Specifically, when subjected to the image reconstruction technique detailed in [19] and considering the MSE, both StarGAN and AGGAN witnessed significant declines in their disruptive efficacy by 79.41% and 72.10% respectively. However, AttGAN exhibited a substantially lower decrease of 28.38%, highlighting its heightened resilience to the disruptive impact of image reconstruction techniques.

Further results can be found in Appendix B.1.1, where the resilience to image reconstruction was assessed using a DeepFake model different from the one the perturbation was initially trained on. The data in tables B.1, B.2, B.3, B.4, B.5, and B.6 provide additional confirmation for the previously discussed insights. The perturbation employed in this approach shows significant susceptibility to removal, and the constrained transferability of this technique amplifies the decrease in disruption efficacy, especially when exposed to image reconstruction methods. This highlights the vital importance of DeepFake Disruption methods having the ability to transfer seamlessly across various DeepFake models while also demonstrating resilience against image reconstruction techniques.

5.4.2.2 Gray-Box Attack

The reconstruction results of the gray-box attack methodology, as discussed in [17], are detailed in tables 5.11, 5.12, and 5.13. Specifically, table 5.11 displays the results for the adversarial perturbation trained on StarGAN and evaluated on StarGAN. Likewise, table 5.12 showcases the results for the adversarial perturbation trained on AttGAN and tested on AttGAN. Finally, table 5.13 illustrates the outcomes for the adversarial perturbation trained on AGGAN and tested on AGGAN.

StarGAN - StarGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.99536	0.36215	6.43988
Reconstruction	0.00410	0.96967	30.70151
Blur	0.03183	0.88934	22.25755
Resize	0.04631	0.88925	20.44494

Table 5.11: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained and tested on StarGAN. Using the disruption method referenced in [17].

AttGAN - AttGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.29961	0.64570	12.45810
Reconstruction	0.09877	0.81126	18.49833
Blur	0.04942	0.88106	22.36980
Resize	0.08432	0.83701	19.44018

Table 5.12: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained and tested on AttGAN. Using the disruption method referenced in [17].

AGGAN - AGGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	1.35428	0.59685	9.91585
Reconstruction	0.00348	0.97746	33.36913
Blur	0.02298	0.92216	25.90397
Resize	0.04240	0.93296	24.95990

Table 5.13: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained and tested on AGGAN. Using the disruption method referenced in [17].

The results outlined in Table 5.11 indicate that the perturbation, exclusively trained on the StarGAN model, is easily removable. Moving to Table 5.9, the perturbation trained on AttGAN stands out once again for an higher resilience to image reconstruction techniques compared to other DeepFake models. However, unlike the white-box attack, this model was sensitive to all image reconstruction techniques. In Table 5.10, when the perturbation is trained on AGGAN, similar to the StarGAN model, it demonstrates ease of removal. Although this method demonstrated higher disruption efficacy, evident from the elevated MSE values, it is apparent that the disruption is easily mitigated, approximately by 90%.

It becomes evident that this approach lacks robustness against image reconstruction techniques. In contrast to the white-box attack, AttGAN demonstrated higher sensitivity to image reconstruction methods, but still maintaining a higher resilience compared to other models. When subjected to the image reconstruction technique detailed in [19] and considering the MSE, all models exhibited a significant decline in their disruptive efficacy. StarGAN experienced a 99.59% decrease, AGGAN 99.74%, and AttGAN 67.03%.

Further results can be found in Appendix B.1.2, the resilience to image reconstruction was assessed using a DeepFake model different from the one the perturbation was initially trained on. The data in tables B.7, B.8, B.9, B.10, B.11, and B.12 provide additional confirmation for the previously discussed insights. The perturbation employed in this approach shows significant susceptibility to removal, and the constrained transferability of this technique amplifies the decrease in disruption efficacy, especially when exposed to image reconstruction methods. Once again, this proves the vital importance of DeepFake Disruption methods having the ability to transfer seamlessly across various DeepFake models while also demonstrating resilience against image reconstruction techniques.

5.4.2.3 Black-Box Attack

The reconstruction results of the black-box attack methodology, as discussed in [18], are detailed in tables 5.14, 5.15, and 5.16. Specifically, table 5.14 displays the results for the cross-model adversarial perturbation tested on StarGAN. Likewise, table 5.15 showcases the results for the cross-model adversarial perturbation testes on AttGAN. Finally, table 5.16 illustrates the outcomes for the cross-model adversarial perturbation tested on AGGAN.

StarGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.20119	0.59981	13.20751
Reconstruction	0.02984	0.86123	22.24719
Blur	0.05800	0.74377	19.06368
Resize	0.02434	0.88035	22.90689

Table 5.14: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. Tested on StarGAN, using the cross-model disruption method referenced in [18].

AttGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.05986	0.74296	18.87833
Reconstruction	0.02305	0.88079	23.89917
Blur	0.01133	0.92711	27.24718
Resize	0.02166	0.89400	24.24911

Table 5.15: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. Tested on AttGAN, using the cross-model disruption method referenced in [18].

AGGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.12635	0.73689	17.82545
Reconstruction	0.08420	0.79885	20.92040
Blur	0.12374	0.70397	18.30965
Resize	0.07887	0.80692	21.19172

Table 5.16: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. Tested on AGGAN, using the cross-model disruption method referenced in [18].

The observed results in perturbation effectiveness indicate the vulnerability of the CMUA-Watermark method to image reconstruction techniques across different DeepFake models. Notably, and considering the MSE, the cross-model perturbation tested on the StarGAN model exhibits the highest susceptibility, highlighting a significant 85.17% reduction in its impact when subjected to image reconstruction. AttGAN, while showing some resilience, still experiences a 61.49% decrease. AGGAN, although demonstrating more robustness compared to StarGAN and AttGAN, still registers a considerable 33.63% decrease.

Additionally, it is worth noting that the downsampling and upsampling technique outperforms the method referenced in [19] when tested on StarGAN and AGGAN, while the median filter achieves higher results when tested on AttGAN.

5.4.2.4 Preliminary Conclusions

The evaluation of these attack methods indicates their respective abilities to withstand image reconstruction techniques. The white-box and black-box attacks demonstrate, on average, a more resilient nature, with both showing a decrease in their effectiveness by approximately 60% when subjected to image reconstruction. This suggests that these attacks maintain a relatively higher level of perturbation impact even after the image undergoes reconstruction. On the other hand, the gray-box attack exhibits a notably higher average decrease of 88.79%. This implies that the gray-box attack under evaluation is more susceptible to image reconstruction techniques, experiencing a more substantial reduction in its effectiveness compared to the white-box and black-box attacks.

The visual outcomes of this evaluation are displayed in Appendix B. The outcomes of the white-box, gray-box, and black-box attacks are visible in B.2.1, B.2.2, and B.2.3, respectively. A more thorough analysis of these figures highlights the heightened vulnerability of the gray-box attack to image reconstruction techniques. Additionally, it is evident that both the white-box and black-box attacks demonstrate greater resilience to image reconstruction techniques. Specifically, the white-box attack proves highly efficient in mitigating blur, and upsampling and downsampling techniques.

5.4.3 Observations

While the majority of observations focused on MSE, further confirmation was obtained by examining additional metrics such as SSIM (Structural Similarity Index) and PSNR (Peak Signal-to-Noise Ratio). The results align with the observed trends, where higher SSIM and PSNR values indicate a greater resemblance between images, and as the Mean Squared Error (MSE) decreases, these metrics proportionally increase. This consistent pattern further validates the observations, emphasizing that as the distortion or error in the images diminishes (lower MSE), the similarity, as indicated by higher SSIM and PSNR values, becomes more pronounced. This alignment of metrics supports the conclusion that the perturbation's effectiveness diminishes as evidenced by the decreasing MSE, ultimately resulting in reduced image dissimilarity.

6 Conclusion

6.1 Conclusion

This dissertation aimed to evaluate the efficacy of various DeepFake disruption methods, focusing on their adaptability across diverse DeepFake models and their resistance to image reconstruction techniques. Three distinct approaches were analyzed, each employing different adversarial attack strategies: white-box attack, gray-box attack, and black-box attack.

The white-box attack showcased significant advantages in terms of image reconstruction, highlighting its strength in resilience, but still not reaching an ideal standard. Additionally, its limited transferability across diverse DeepFake models raised concerns. The need to generate a perturbation for each attribute and model proves inefficient, particularly considering the unpredictable nature of real-world scenarios where the attacker's choice of DeepFake model and attributes are unknown. The gray-box attack demonstrated exceptional disruptive capabilities but lacked efficient transferability. Furthermore, it exhibited a notable vulnerability in terms of image reconstruction robustness. In contrast, the black-box attack emerged as the most effective approach, excelling in both model transferability and image reconstruction robustness. Its cross-model disruption approach displayed remarkable adaptability across different DeepFake models, even though it did not achieve disruption levels as high as methods customizing perturbations for specific models. In terms of resilience to image reconstruction, the black-box attack demonstrated higher effectiveness compared to other methods, though it still falls short of the ideal standard.

Of all the evaluated methods, the CMUA-Watermark method (black-box attack), emerges as the most promising approach based on the observed results. This method exhibits superior suitability for real-life scenarios and presents the most effective strategy for addressing the threats posed by DeepFake technology.

In conclusion, the significance of possessing the capability to seamlessly transfer across a variety of DeepFake models and exhibiting robust resilience against image reconstruction techniques is undeniable. Creating an effective DeepFake Disruption method requires both transferability and resilience to image reconstruction techniques. Merely having transferability is insufficient, as a straightforward image reconstruction technique can diminish its efficacy. Similarly, relying solely on resilience to image reconstruction methods leaves the defense vulnerable, as employing an alternative DeepFake model can easily circumvent such defenses. Therefore, a potent DeepFake Disruption method needs a balanced combination of both transferability and resilience to image reconstruction techniques to ensure comprehensive and effective protection against the evolving challenges posed by DeepFake technology.

6.2 Future Work

Developing an adversarial perturbation that withstands removal attempts poses a considerable challenge. Unfortunately, the examined approaches fell short in showcasing promising results concerning resilience to image reconstruction. This highlights a notable gap in current methodologies, signaling the need for future research to delve deeper into enhancing the resilience of adversarial perturbations during image reconstruction processes. Addressing this aspect could contribute significantly to the development of more robust techniques in tackling DeepFakes.

It is noteworthy that existing defenses primarily focus on disruption efficacy and transferability, often neglecting the importance of image reconstruction resilience. As adversarial attacks continue to evolve, so too do the methods devised to counteract them. Thus, there exists an ongoing struggle to keep pace with the dynamic landscape of adversarial techniques, necessitating continuous research and innovation to develop effective defense mechanisms.

7 Bibliography

[1] X. Wang, J. Huang, S. Ma, S. Nepal, and C. Xu, "Deepfake Disrupter: The Detector of DeepFake Is My Friend," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (New Orleans, LA, USA), pp. 14900–14909, June 2022.

[2] N. Ruiz, S. A. Bargal, and S. Sclaroff, "Disrupting Deepfakes: Adversarial Attacks Against Conditional Image Translation Networks and Facial Manipulation Systems," in *Computer Vision – ECCV 2020 Workshops*, (Glasgow, UK), pp. 236–251, August 2020.

[3] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, "Deepfakes and Beyond: A Survey of Face Manipulation and Fake Detection," *Information Fusion*, vol. 64, pp. 131–148, December 2020.

[4] X. Zhang, S. Karaman, and S.-F. Chang, "Detecting and Simulating Artifacts in GAN Fake Images," in *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, (Delft, Netherlands), pp. 1–6, December 2019.

[5] H. Qi, Q. Guo, F. Juefei-Xu, X. Xie, L. Ma, W. Feng, Y. Liu, and J. Zhao, "DeepRhythm: Exposing DeepFakes with Attentional Visual Heartbeat Rhythms," in *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, (New York, NY, USA), p. 4318–4327, June 2020.

[6] P. Sun, Y. Li, H. Qi, and S. Lyu, "Landmark Breaker: Obstructing DeepFake By Disturbing Landmark Extraction," in *2020 IEEE International Workshop on Information Forensics and Security (WIFS)*, (New York, NY, USA), pp. 1–6, December 2020.

[7] E. Segalis and E. Galili, "OGAN: Disrupting Deepfakes with an Adversarial Attack that Survives Training," 2020. arXiv:2006.12247.

[8] L. Lv, "Smart Watermark to Defend Against Deepfake Image Manipulation," in *2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS)*, (Chengdu, China), pp. 380–384, April 2021.

[9] H. Qiu, Y. Du, and T. Lu, “The Framework of Cross-Domain and Model Adversarial Attack Against Deepfake,” *Future Internet*, vol. 14, January 2022.

[10] J. Dong, Y. Wang, J. Lai, and X. Xie, “Restricted Black-Box Adversarial Attack Against DeepFake Face Swapping,” *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 2596–2608, April 2023.

[11] S. Zhang, H. Gao, and Q. Rao, “Defense Against Adversarial Attacks by Reconstructing Images,” *IEEE Transactions on Image Processing*, vol. 30, pp. 6117–6129, July 2021.

[12] Z. Chen, L. Xie, S. Pang, Y. He, and B. Zhang, “Magdr: Mask-guided Detection and Reconstruction for Defending Deepfakes,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Nashville, TN, USA), pp. 9010–9019, June 2021.

[13] G. Iglesias, E. Talavera, and A. Díaz-Álvarez, “A Survey on GANs for Computer Vision: Recent Research, Analysis and Taxonomy,” *Computer Science Review*, vol. 48, May 2023.

[14] Y. Choi, M. Choi, M. Kim, J. Ha, S. Kim, and J. Choo, “StarGAN: Unified Generative Adversarial Networks for Multi-domain Image-to-Image Translation,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Salt Lake City, UT, USA), pp. 8789–8797, June 2018.

[15] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, “AttGAN: Facial Attribute Editing by Only Changing What You Want,” *IEEE Transactions on Image Processing*, vol. 28, pp. 5464–5478, November 2019.

[16] H. Tang, D. Xu, N. Sebe, and Y. Yan, “Attention-Guided Generative Adversarial Networks for Unsupervised Image-to-Image Translation,” in *2019 International Joint Conference on Neural Networks (IJCNN)*, (Budapest, Hungary), pp. 1–8, July 2019.

[17] R. Wang, Z. Huang, Z. Chen, L. Liu, J. Chen, and L. Wang, “Anti-Forgery: Towards a Stealthy and Robust DeepFake Disruption Attack via Adversarial Perceptual-aware Perturbations,” in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, (Vienna), pp. 761–767, July 2022.

[18] H. Huang, Y. Wang, Z. Chen, Y. Zhang, Y. Li, Z. Tang, W. Chu, J. Chen, W. Lin, and K.-K. Ma, “CMUA-Watermark: A Cross-Model Universal Adversarial Watermark for Combating Deepfakes,” in *The 36th AAAI Conference on Artificial Intelligence*, vol. 36, pp. 989–997, February 2022.

[19] A. Mustafa, S. H. Khan, M. Hayat, J. Shen, and L. Shao, “Image Super-Resolution as a Defense Against Adversarial Attacks,” *IEEE Transactions on Image Processing*, vol. 29, pp. 1711–1724, September 2020.

[20] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, “Enhanced Deep Residual Networks for Single Image Super-Resolution,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, (Honolulu, HI, USA), pp. 1132–1140, July 2017.

[21] F. Juefei-Xu, R. Wang, Y. Huang, Q. Guo, L. Ma, and Y. Liu, “Countering Malicious DeepFakes: Survey, Battleground, and Horizon,” *International Journal of Computer Vision*, vol. 130, pp. 1678–1734, July 2022.

[22] Sensity, “The State of Deepfakes: Landscape, Threats, and Impact,” October 2019.

[23] T. Karras, S. Laine, and T. Aila, “A Style-Based Generator Architecture for Generative Adversarial Networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 4217–4228, December 2021.

[24] M. Liu, Y. Ding, M. Xia, X. Liu, E. Ding, W. Zuo, and S. Wen, “STGAN: A Unified Selective Transfer Network for Arbitrary Image Attribute Editing,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Long Beach, CA, USA), pp. 3668–3677, June 2019.

[25] K. Liu, I. Perov, D. Gao, N. Chervoniy, W. Zhou, and W. Zhang, “Deepfacelab: Integrated, Flexible and Extensible Face-Swapping Framework,” *Pattern Recognition*, vol. 141, September 2023.

[26] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, “GANimation: Anatomically-Aware Facial Animation from a Single Image,” in *Computer Vision – ECCV 2018: 15th European Conference*, (Munich, Germany), p. 835–851, September 2018.

[27] H. Li, B. Li, S. Tan, and J. Huang, “Identification of Deep Network Generated Images using Disparities in Color Components,” *Signal Processing*, vol. 174, September 2020.

[28] M. Koopman, A. Macarulla Rodriguez, and Z. Geradts, “Detection of Deepfake Video Manipulation,” in *The 20th Irish Machine Vision and ImageProcessing Conference (IMVIP)*, (Ulster University, Northern Ireland), pp. 133—136, August 2018.

[29] Z. Liu, X. Qi, and P. H. Torr, “Global Texture Enhancement for Fake Face Detection in the Wild,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Seattle, WA, USA), pp. 8057–8066, June 2020.

[30] L. Chai, D. Bau, S.-N. Lim, and P. Isola, “What Makes Fake Images Detectable? Understanding Properties that Generalize,” in *Computer Vision – ECCV 2020*, (Glasgow, UK), pp. 103–120, August 2020.

[31] Y. Huang, F. Juefei-Xu, Q. Guo, Y. Liu, and G. Pu, “FakeLocator: Robust Localization of GAN-Based Face Manipulations,” *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 2657–2672, January 2022.

[32] Z. Guo, G. Yang, J. Chen, and X. Sun, “Fake Face Detection via Adaptive Manipulation Traces Extraction Network,” *Computer Vision and Image Understanding*, vol. 204, March 2021.

[33] X. Gong, S. Chang, Y. Jiang, and Z. Wang, “AutoGAN: Neural Architecture Search for Generative Adversarial Networks,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, (Seoul, Korea (South)), pp. 3223–3233, October 2019.

[34] J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, and T. Holz, “Leveraging Frequency Analysis for Deep Fake Image Recognition,” in *Proceedings of the 37th International Conference on Machine Learning*, ICML’20, July 2020.

[35] T. Mittal, U. Bhattacharya, R. Chandra, A. Bera, and D. Manocha, “Emotions Don’t Lie: An Audio-Visual Deepfake Detection Method using Affective Cues,” in *Proceedings of the 28th ACM International Conference on Multimedia*, MM ’20, (New York, NY, USA), p. 2823–2832, October 2020.

[36] Y. Li and S. Lyu, “Exposing DeepFake Videos by Detecting Face Warping Artifacts,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPR)*, (Long Beach, CA, USA), pp. 46–52, June 2019.

[37] U. A. Ciftci, I. Demir, and L. Yin, “FakeCatcher: Detection of Synthetic Portrait Videos using Biological Signals,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, July 2020.

[38] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and Harnessing Adversarial Examples,” in *3rd International Conference on Learning Representations (ICLR)*, (San Diego, CA, USA), May 2015.

[39] A. Kurakin, I. J. Goodfellow, and S. Bengio, “Adversarial Machine Learning at Scale,” in *5th International Conference on Learning Representations, ICLR*, (Toulon, France), April 2017.

[40] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards Deep Learning Models Resistant to Adversarial Attacks,” in *6th International Conference on Learning Representations, ICLR*, (Vancouver, BC, Canada), April 2018.

[41] N. Carlini and D. Wagner, "Towards Evaluating the Robustness of Neural Networks," in *2017 IEEE Symposium on Security and Privacy (SP)*, (San Jose, CA, USA), pp. 39–57, May 2017.

[42] H. Liang, E. He, Y. Zhao, Z. Jia, and H. Li, "Adversarial Attack and Defense: A Survey," *Electronics*, vol. 11, p. 1283, April 2022.

[43] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting Adversarial Attacks with Momentum," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Salt Lake City, UT, USA), pp. 9185–9193, June 2018.

[44] C.-Y. Yeh, H.-W. Chen, S.-L. Tsai, and S.-D. Wang, "Disrupting Image-Translation-Based DeepFake Algorithms with Adversarial Attacks," in *2020 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, (Snowmass, CO, USA), pp. 53–62, March 2020.

[45] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, (Munich, Germany), pp. 234–241, October 2015.

[46] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Honolulu, HI, USA), pp. 1800–1807, July 2017.

[47] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Las Vegas, NV, USA), pp. 770–778, June 2016.

[48] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song, "Generating Adversarial Examples with Adversarial Networks," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, (Stockholm, Sweden), p. 3905–3911, July 2018.

[49] W. Brendel, J. Rauber, and M. Bethge, "Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models," in *6th International Conference on Learning Representations (ICLR)*, (Vancouver, BC, Canada), April 2018.

[50] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein Generative Adversarial Networks," in *Proceedings of the 34th International Conference on Machine Learning (PMLR)*, (Sydney, NSW, Australia), pp. 214–223, August 2017.

[51] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for Hyper-Parameter Optimization," in *25th Annual Conference on Neural Information Processing Systems (NIPS'11)*, (Granada, Spain), p. 2546–2554, December 2011.

[52] S. Chang, B. Yu, and M. Vetterli, “Adaptive Wavelet Thresholding for Image Denoising and Compression,” *IEEE Transactions on Image Processing*, vol. 9, pp. 1532–1546, September 2000.

[53] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep Learning Face Attributes in the Wild,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, (Santiago, Chile), pp. 3730–3738, December 2015.

Appendix A

Visual Results of Disruption

A.1 White-Box Attack

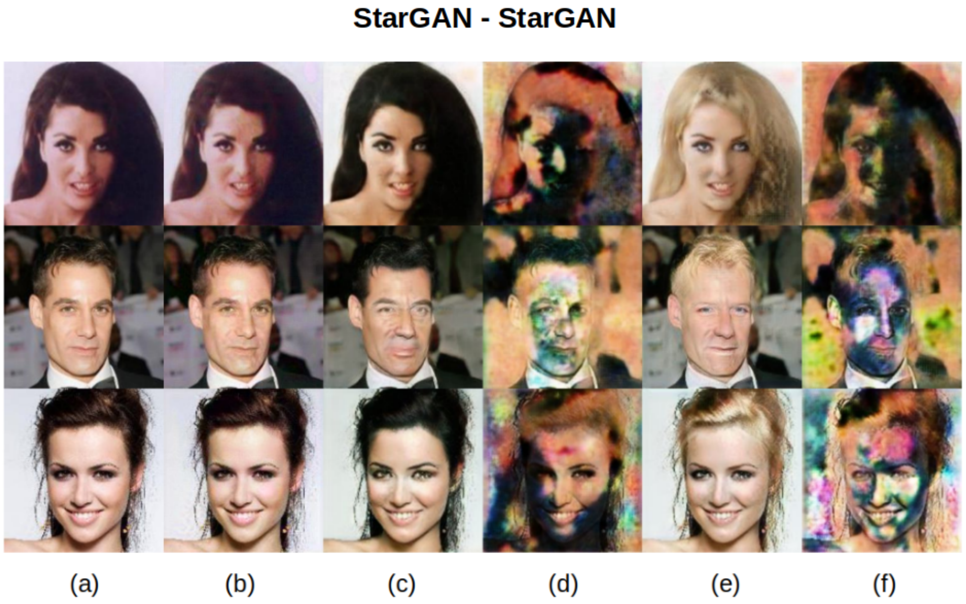


Figure A.1: Disruption results obtained for the method outlined in [2] using a StarGAN - StarGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

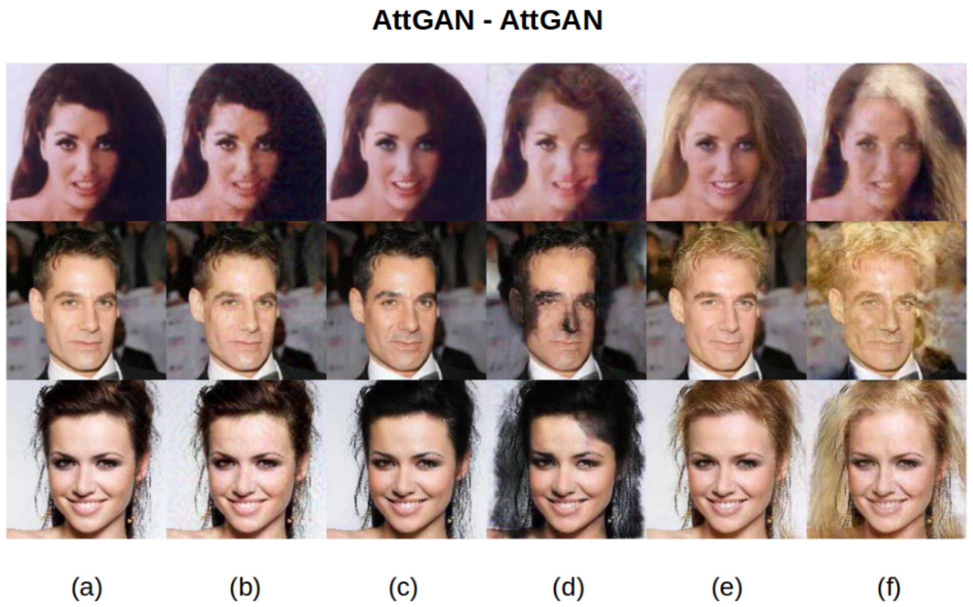


Figure A.2: Disruption results obtained for the method outlined in [2] using an AttGAN - AttGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

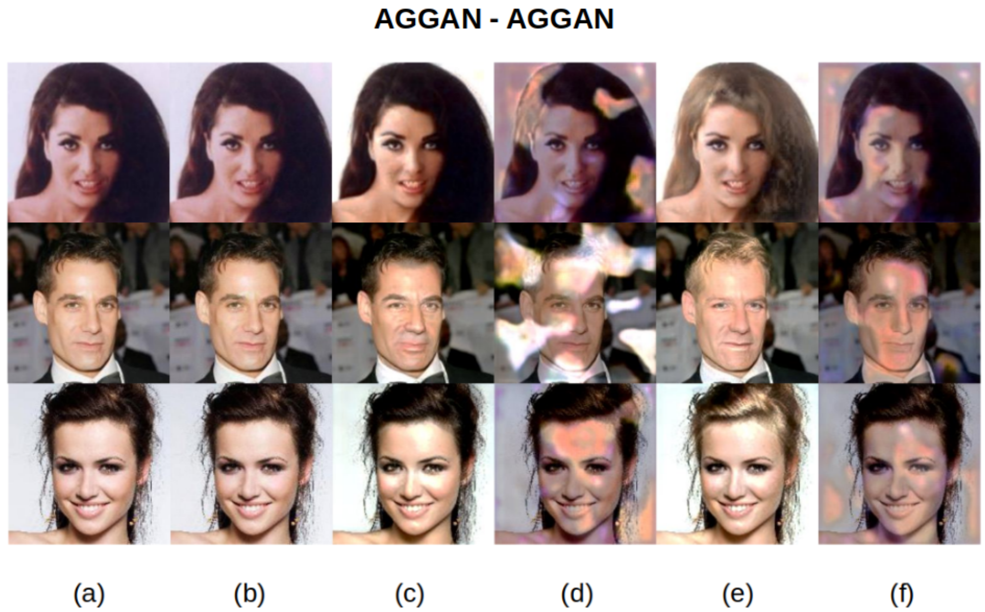


Figure A.3: Disruption results obtained for the method outlined in [2] using an AGGAN - AGGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

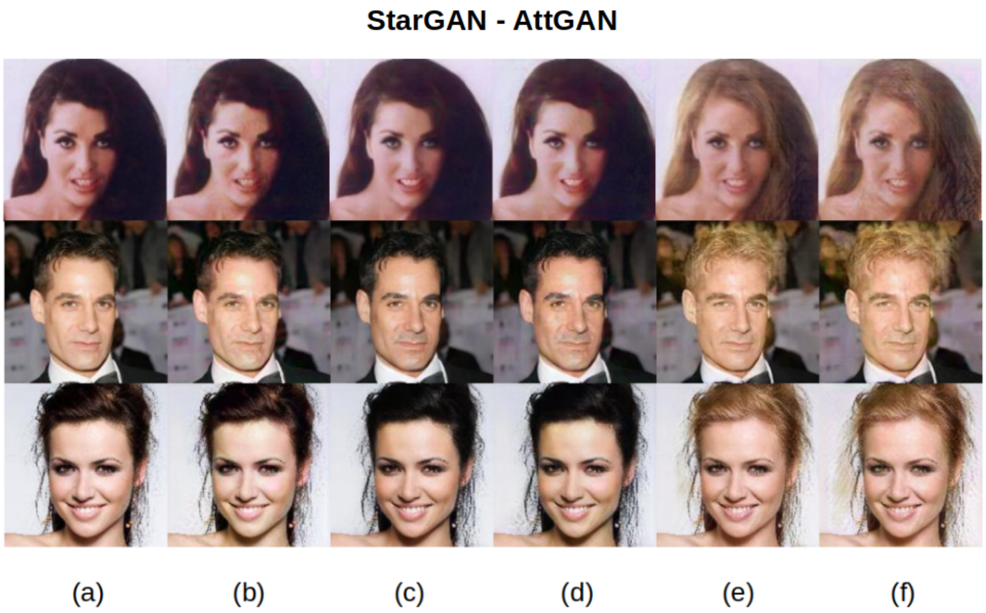


Figure A.4: Disruption results obtained for the method outlined in [2] using a StarGAN - AttGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

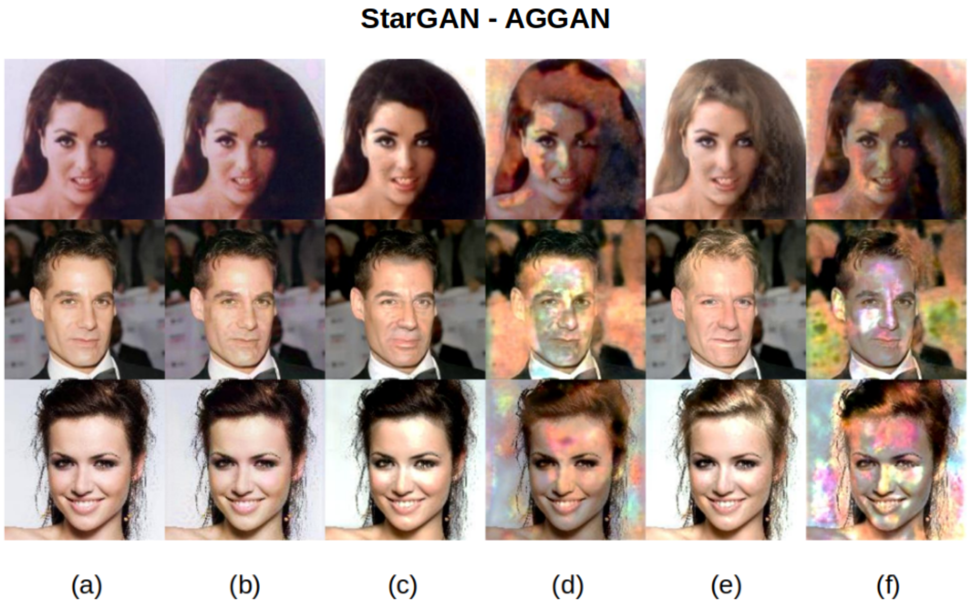


Figure A.5: Disruption results obtained for the method outlined in [2] using a StarGAN - AGGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

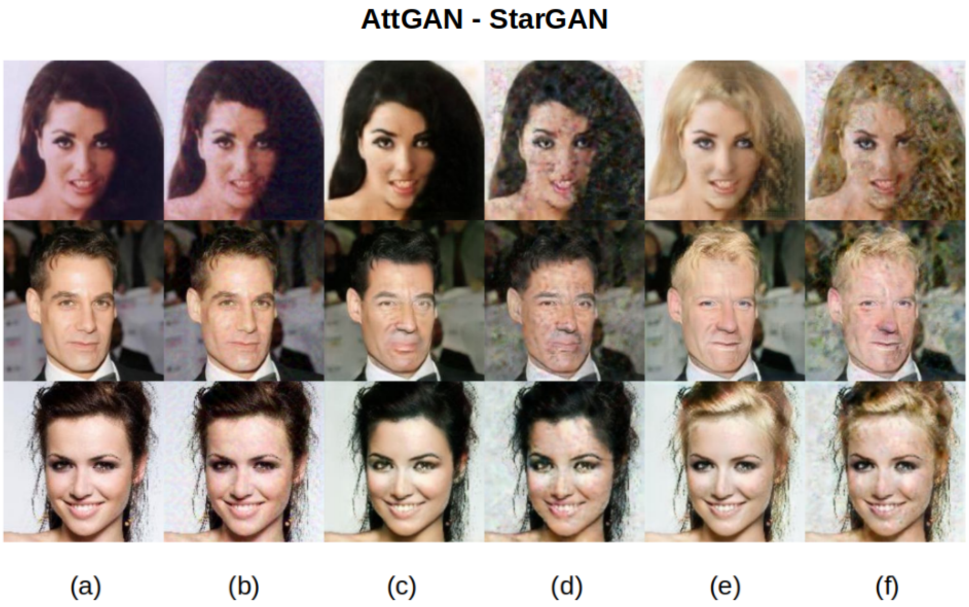


Figure A.6: Disruption results obtained for the method outlined in [2] using a AttGAN - StarGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

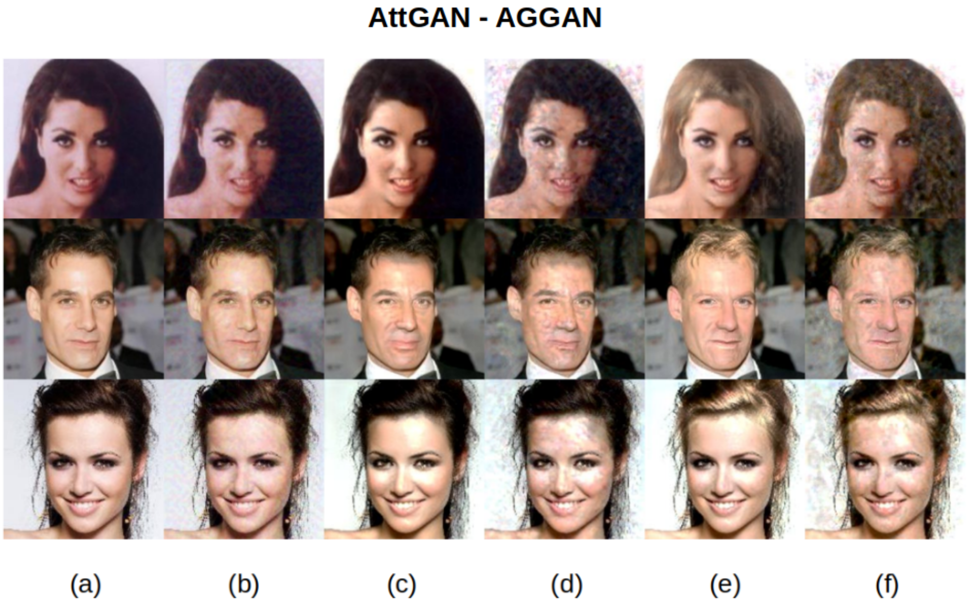


Figure A.7: Disruption results obtained for the method outlined in [2] using a AttGAN - AGGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

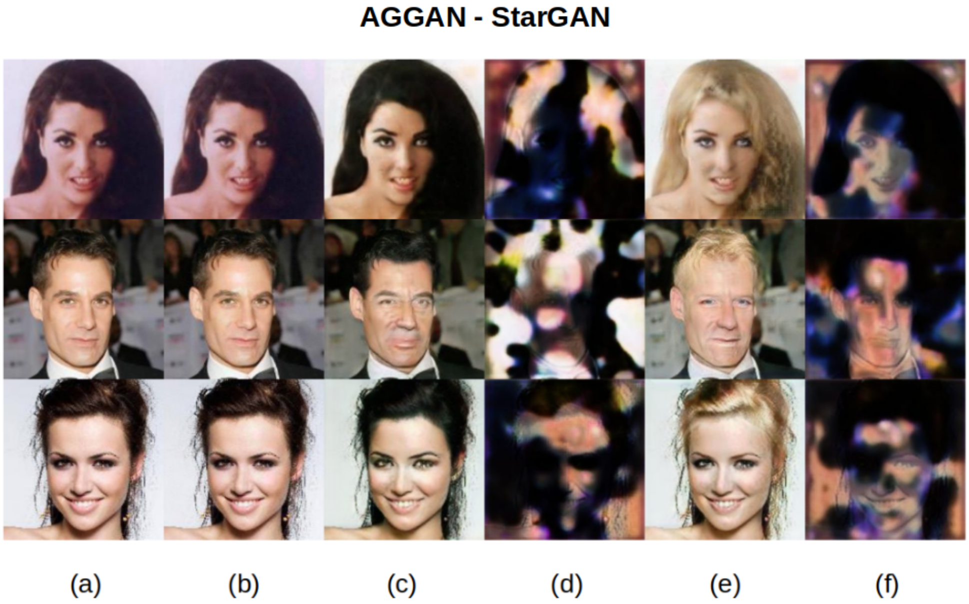


Figure A.8: Disruption results obtained for the method outlined in [2] using a AGGAN - StarGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

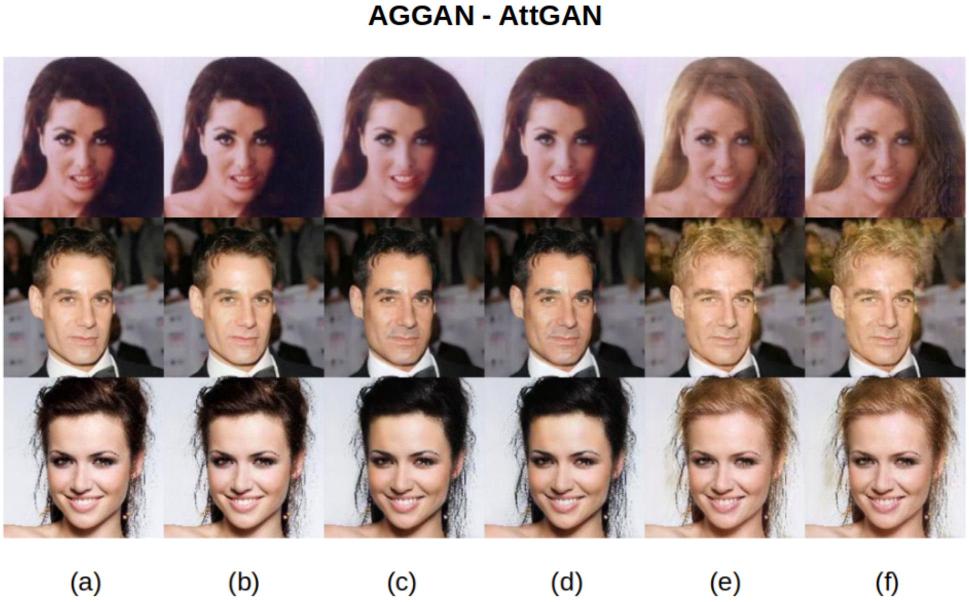


Figure A.9: Disruption results obtained for the method outlined in [2] using a AGGAN - AttGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

A.2 Gray-Box Attack

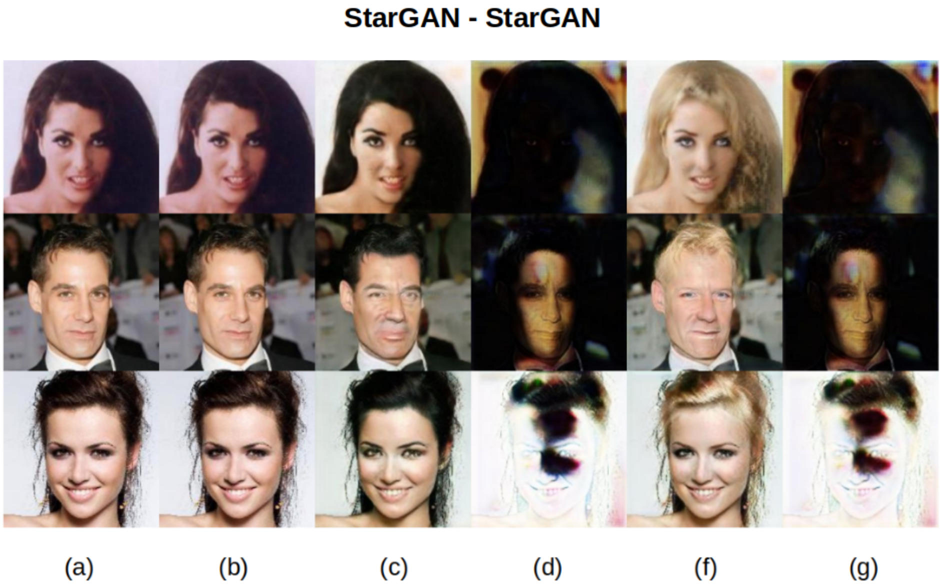


Figure A.10: Disruption results obtained for the method outlined in [17] using a StarGAN - StarGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

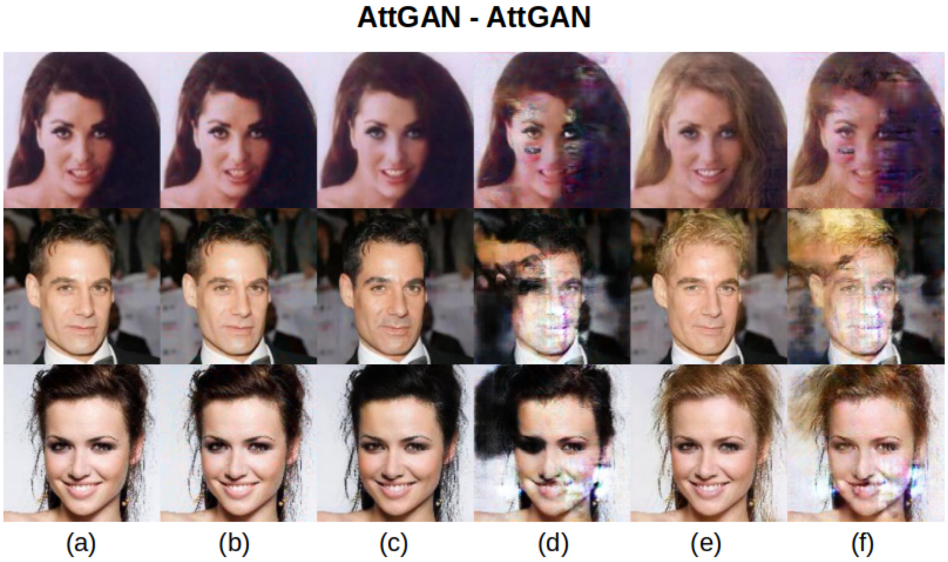


Figure A.11: Disruption results obtained for the method outlined in [17] using an AttGAN - AttGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

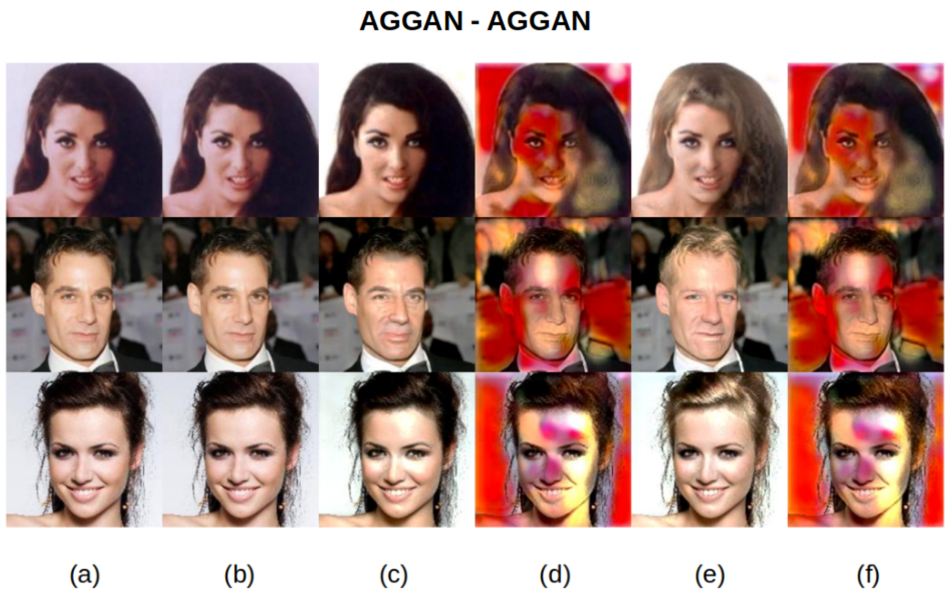


Figure A.12: Disruption results obtained for the method outlined in [17] using an AGGAN - AGGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

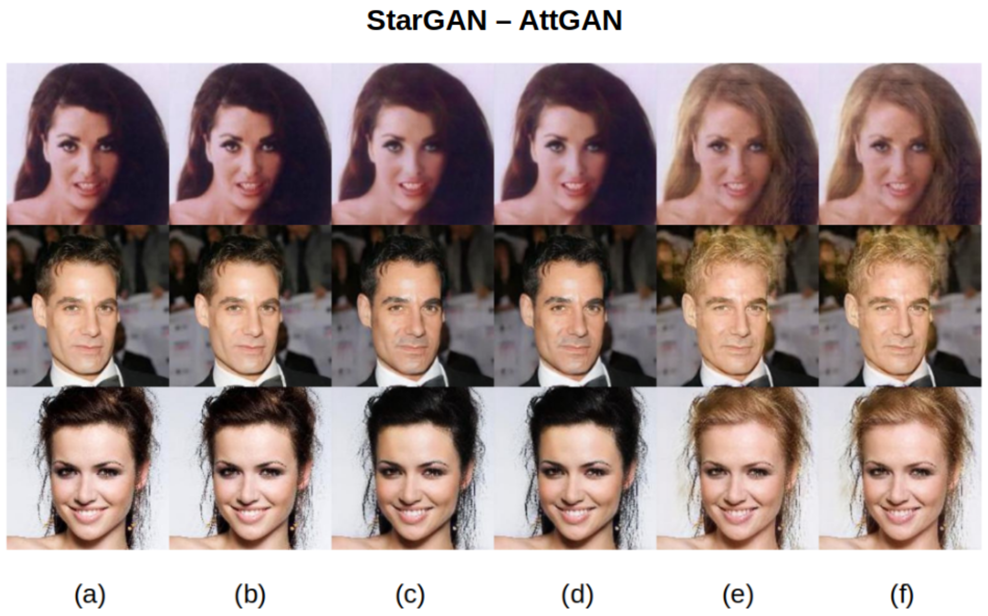


Figure A.13: Disruption results obtained for the method outlined in [17] using a StarGAN - AttGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

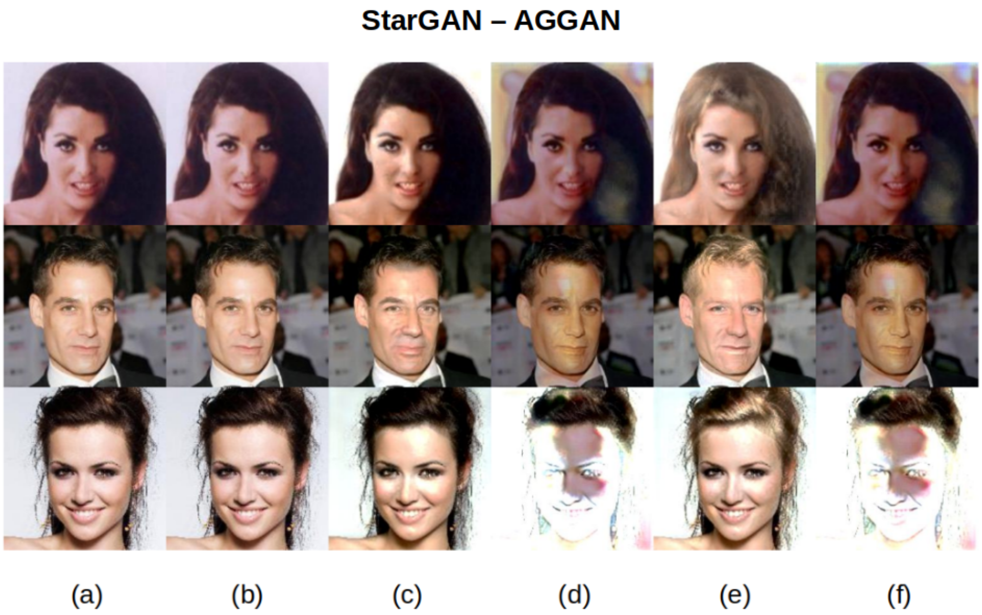


Figure A.14: Disruption results obtained for the method outlined in [17] using a StarGAN - AGGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

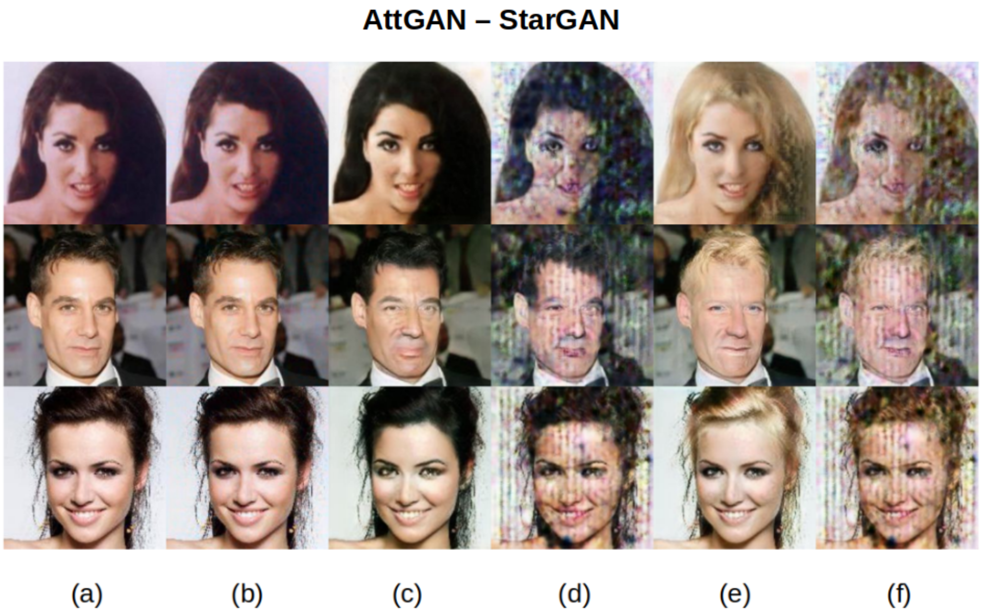


Figure A.15: Disruption results obtained for the method outlined in [17] using a AttGAN - StarGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

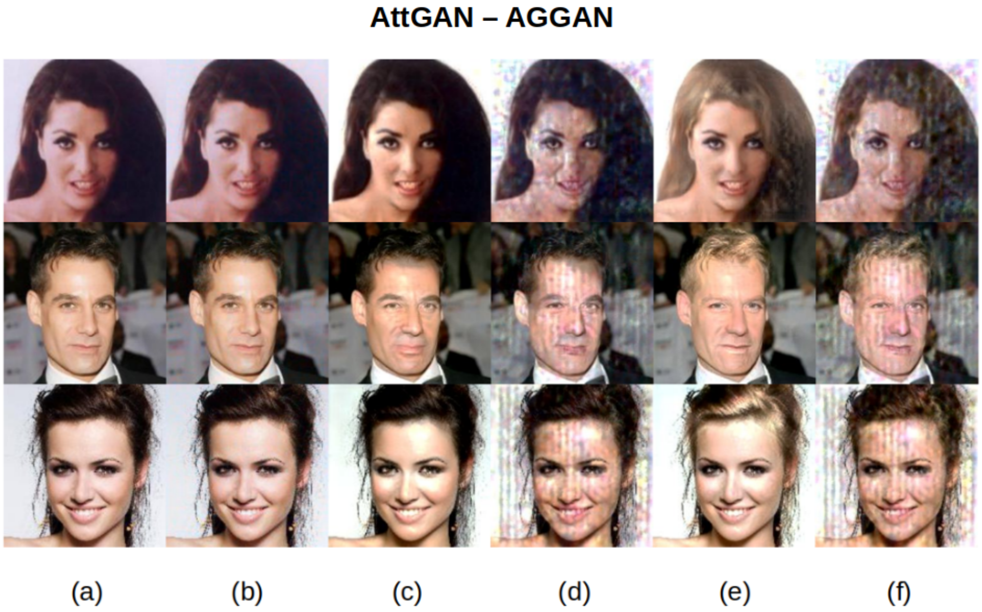


Figure A.16: Disruption results obtained for the method outlined in [17] using a AttGAN - AGGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

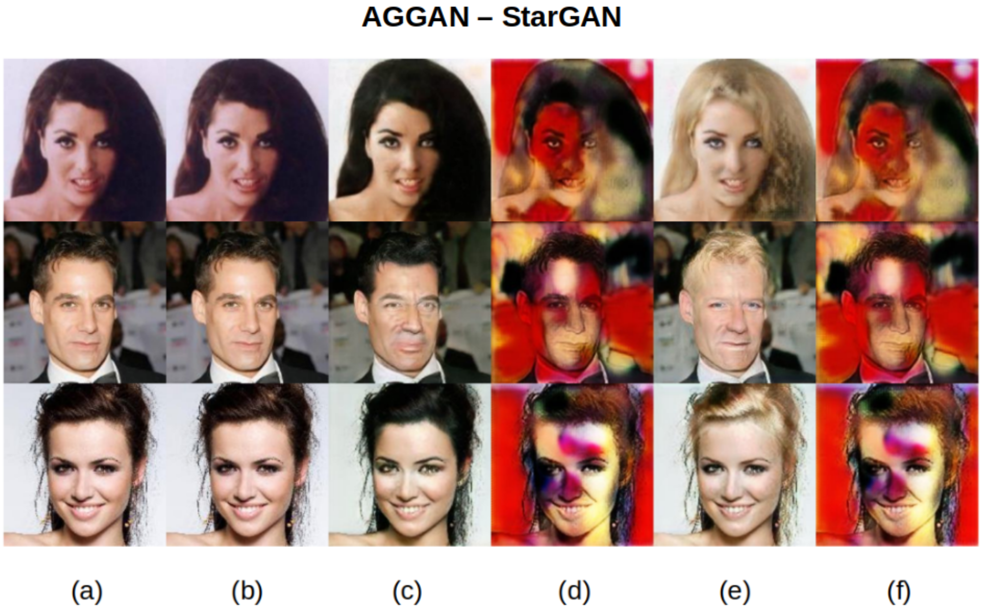


Figure A.17: Disruption results obtained for the method outlined in [17] using a AGGAN - StarGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

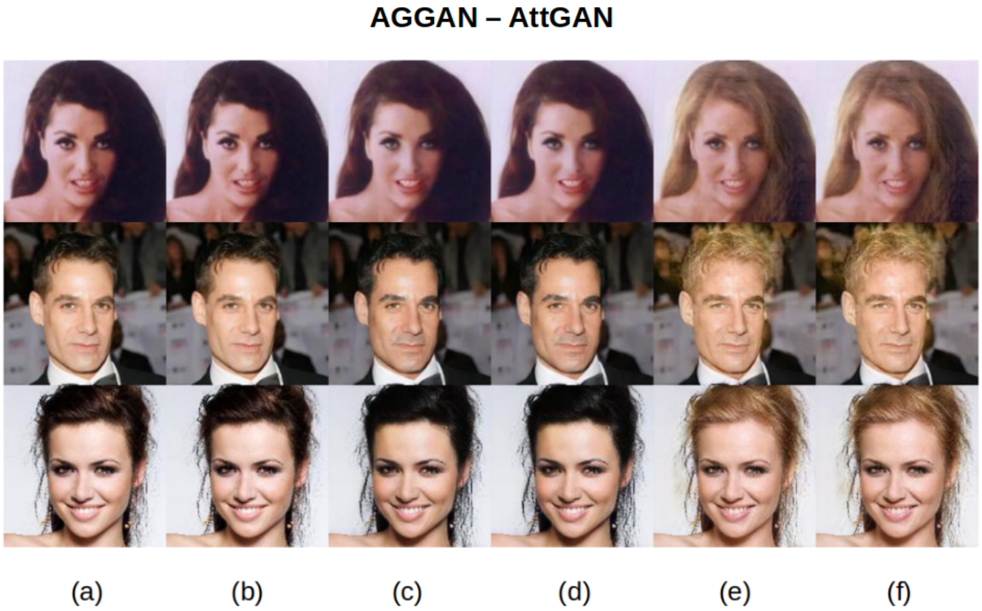


Figure A.18: Disruption results obtained for the method outlined in [17] using a AGGAN - AttGAN configuration. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

A.3 Black-Box Attack

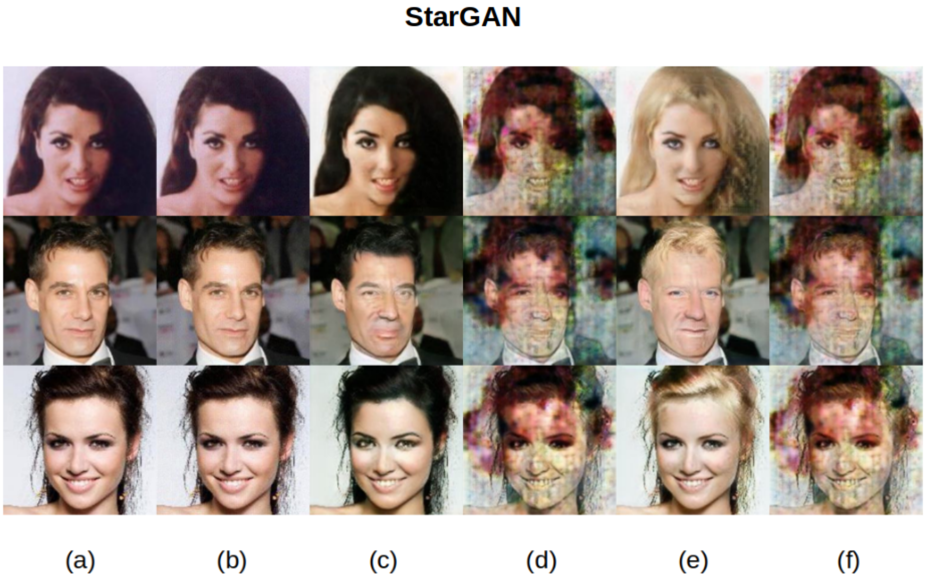


Figure A.19: Disruption results obtained for the method outlined in [18] using the StarGAN model. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

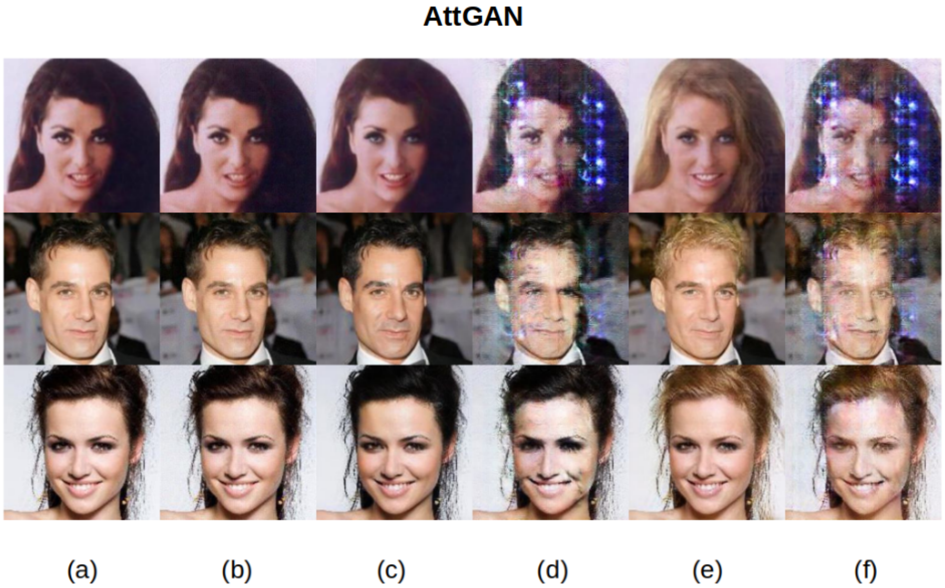


Figure A.20: Disruption results obtained for the method outlined in [18] using the AttGAN model. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

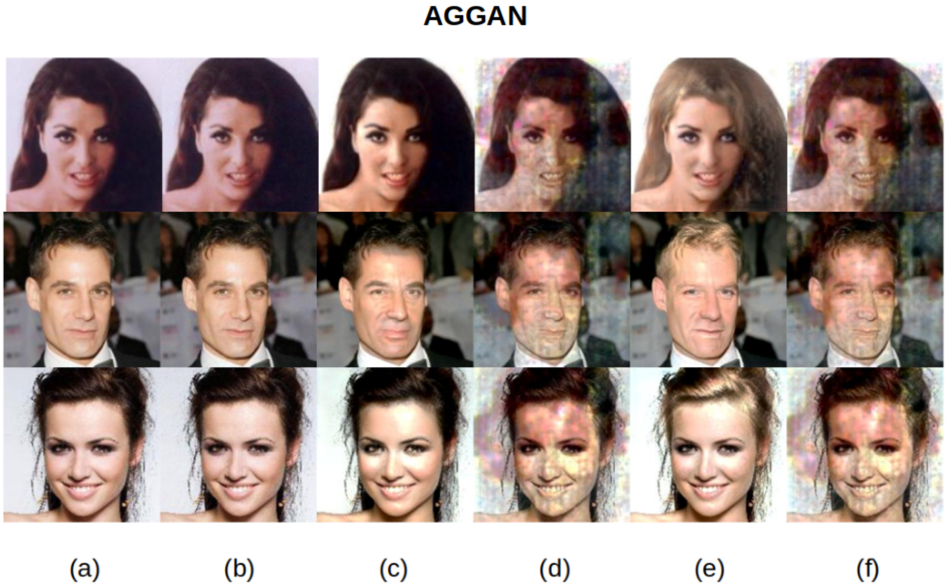


Figure A.21: Disruption results obtained for the method outlined in [18] using the AGGAN model. **(a)** - original image; **(b)** - adversarial image; **(c)** - DeepFake of the original image - black hair; **(d)** - DeepFake of the adversarial image - black hair; **(e)** - DeepFake of the original image - blonde hair; **(f)** - DeepFake of the adversarial image - blonde hair.

Appendix B

Image Reconstruction Results

B.1 Cross-Model Results

B.1.1 White-Box Attack

To further assess the robustness of the white-box attack against image reconstruction techniques, tests were conducted on the reconstruction efficacy using a DeepFake model distinct from the one on which the perturbation was originally trained. Table B.1 displays the reconstruction results for the perturbation trained on StarGAN and tested on AttGAN, table B.2 displays the reconstruction results for the perturbation trained on StarGAN and tested on AGGAN, table B.3 displays the reconstruction results for the perturbation trained on AttGAN and tested on StarGAN, table B.4 displays the reconstruction results for the perturbation trained on AttGAN and tested on AGGAN, table B.5 displays the reconstruction results for the perturbation trained on AGGAN and tested on StarGAN, and finally, table B.6 displays the reconstruction results for the perturbation trained on AGGAN and tested on AttGAN.

StarGAN - AttGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.00316	0.97264	33.14200
Reconstruction	0.00308	0.97145	32.56021
Blur	0.00360	0.96668	32.11135
Resize	0.00316	0.97062	32.61585

Table B.1: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on StarGAN and tested on AttGAN. Using the disruption method referenced in [2].

StarGAN - AGGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.20848	0.71938	14.88903
Reconstruction	0.05997	0.88886	21.92562
Blur	0.11653	0.81552	18.04553
Resize	0.16342	0.78302	16.17851

Table B.2: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on StarGAN and tested on AGGAN. Using the disruption method referenced in [2].

AttGAN - StarGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.02739	0.82244	22.33454
Reconstruction	0.00804	0.93807	27.78110
Blur	0.01204	0.90775	25.98808
Resize	0.00641	0.94796	29.05413

Table B.3: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on AttGAN and tested on StarGAN. Using the disruption method referenced in [2].

AttGAN - AGGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.02449	0.87820	26.07579
Reconstruction	0.00948	0.94510	29.92363
Blur	0.01207	0.93021	29.00493
Resize	0.00713	0.95199	31.28893

Table B.4: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on AttGAN and tested on AGGAN. Using the disruption method referenced in [2].

AGGAN - StarGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.84871	0.30009	7.01928
Reconstruction	0.21317	0.67918	14.20745
Blur	0.22949	0.58445	12.90069
Resize	0.73479	0.35034	7.67565

Table B.5: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on AGGAN and tested on StarGAN. Using the disruption method referenced in [2].

AGGAN - AttGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.00077	0.98919	38.96056
Reconstruction	0.00120	0.98551	36.19237
Blur	0.00116	0.98399	36.68326
Resize	0.00086	0.98775	37.96833

Table B.6: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on AGGAN and tested on AttGAN. Using the disruption method referenced in [2].

B.1.2 Gray-Box Attack

To further assess the robustness of the gray-box attack against image reconstruction techniques, tests were conducted on the reconstruction efficacy using a DeepFake model distinct from the one on which the perturbation was originally trained. Table B.1 displays the reconstruction results for the perturbation trained on StarGAN and tested on AttGAN, table B.2 displays the reconstruction results for the perturbation trained on StarGAN and tested on AGGAN, table B.3 displays the reconstruction results for the perturbation trained on AttGAN and tested on StarGAN, table B.4 displays the reconstruction results for the perturbation trained on AttGAN and tested on AGGAN, table B.5 displays the reconstruction results for the perturbation trained on AGGAN and tested on StarGAN, and finally, table B.6 displays the reconstruction results for the perturbation trained on AGGAN and tested on AttGAN.

StarGAN - AttGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	6.85041e-06	0.99983	59.76404
Reconstruction	0.00096	0.98978	37.00697
Blur	0.00061	0.99208	39.73656
Resize	0.00065	0.99224	39.07334

Table B.7: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on StarGAN and tested on AttGAN. Using the disruption method referenced in [17].

StarGAN - AGGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.83990	0.62672	10.25817
Reconstruction	0.00340	0.97761	33.37174
Blur	0.02049	0.93673	26.39990
Resize	0.03409	0.93322	24.74426

Table B.8: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on StarGAN and tested on AGGAN. Using the disruption method referenced in [17].

AttGAN - StarGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.07460	0.66883	17.68668
Reconstruction	0.05279	0.73523	19.42430
Blur	0.02722	0.82411	22.39516
Resize	0.01062	0.91696	26.71500

Table B.9: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on AttGAN and tested on StarGAN. Using the disruption method referenced in [17].

AttGAN - AGGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.05990	0.76858	21.87429
Reconstruction	0.04218	0.81796	23.38096
Blur	0.02415	0.87651	26.13271
Resize	0.01118	0.93921	29.84214

Table B.10: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on AttGAN and tested on AGGAN. Using the disruption method referenced in [17].

AGGAN - StarGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	0.64544	0.50673	8.47280
Reconstruction	0.00416	0.96949	30.69474
Blur	0.03316	0.87584	22.04795
Resize	0.03649	0.89947	21.44218

Table B.11: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on AGGAN and tested on StarGAN. Using the disruption method referenced in [17].

AGGAN - AttGAN			
	MSE ↑	**SSIM ↓**	**PSNR ↓**
Adversarial	8.78577e-06	0.99976	58.80888
Reconstruction	0.00096	0.98978	36.99853
Blur	0.00061	0.99203	39.69793
Resize	0.00065	0.99219	39.06546

Table B.12: Comparison between authentic DeepFake images and reconstructed adversarial DeepFake images. The adversarial perturbations were trained on AGGAN and tested on AttGAN. Using the disruption method referenced in [17].

B.2 Visual Results of Image Reconstruction

B.2.1 White-Box Attack

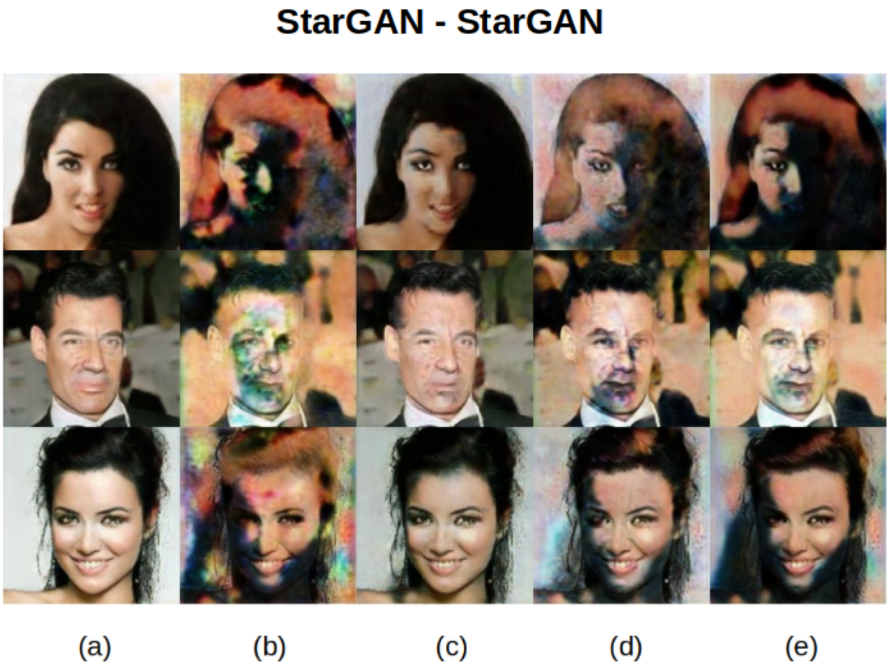


Figure B.1: Reconstruction results obtained for the method outlined in [2] using a StarGAN - StarGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

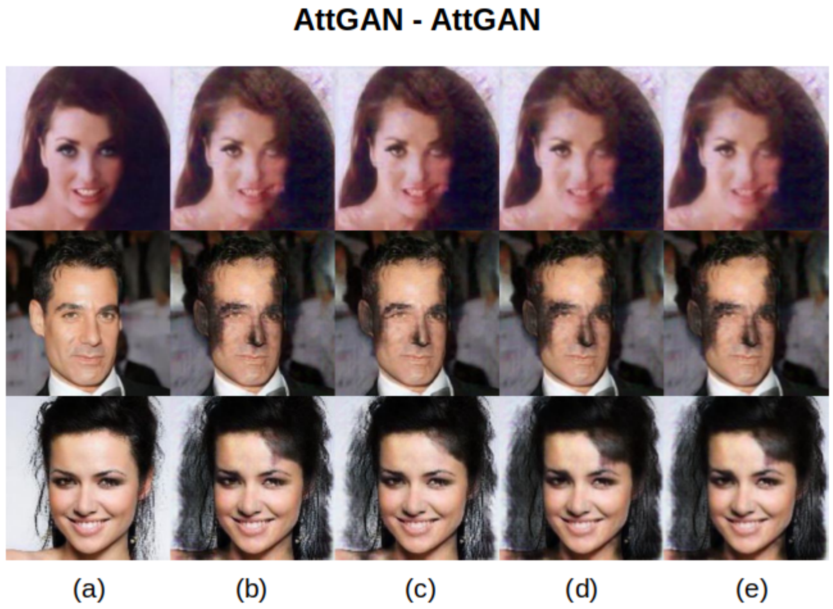


Figure B.2: Reconstruction results obtained for the method outlined in [2] using an AttGAN - AttGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

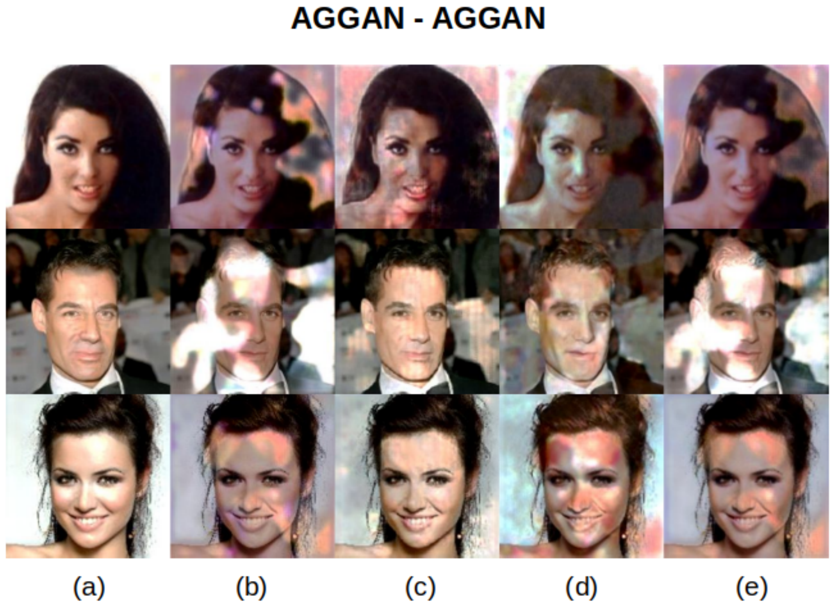


Figure B.3: Reconstruction results obtained for the method outlined in [2] using an AGGAN - AGGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

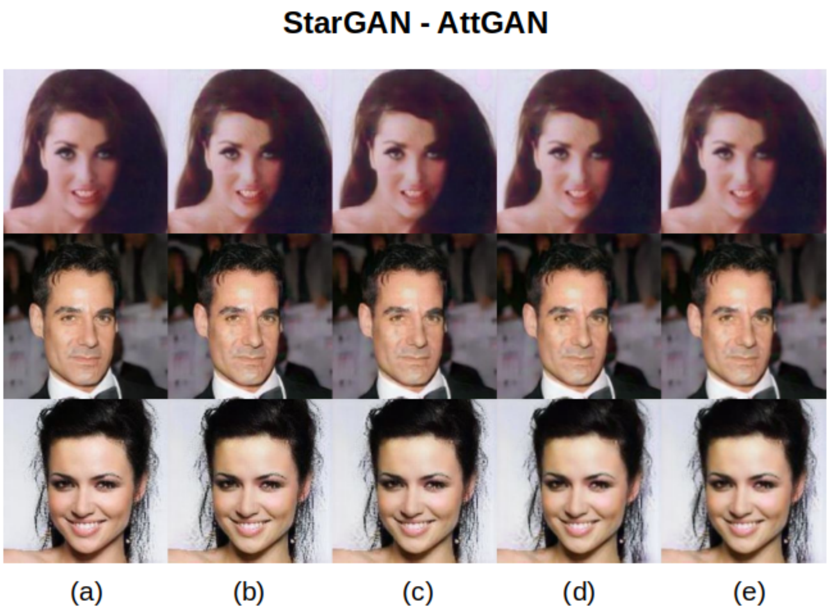


Figure B.4: Reconstruction results obtained for the method outlined in [2] using a StarGAN - AttGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

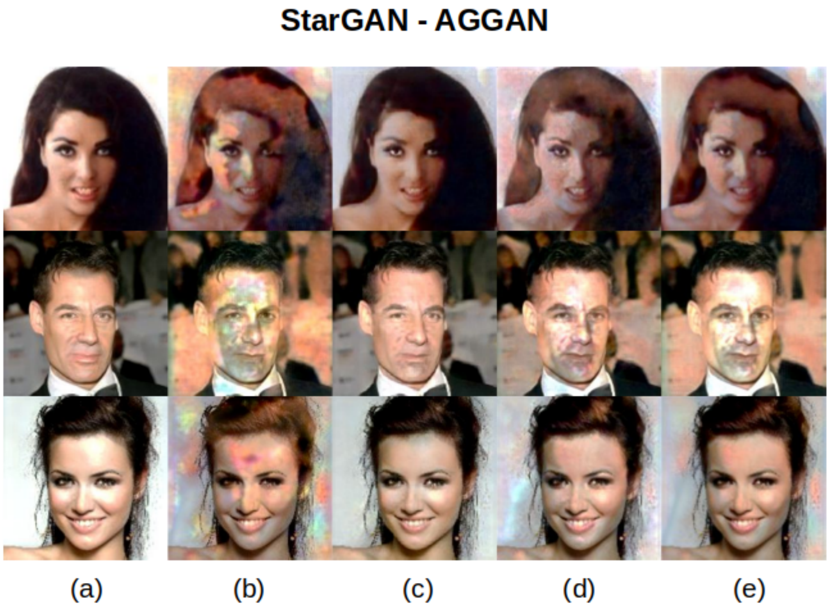


Figure B.5: Reconstruction results obtained for the method outlined in [2] using a StarGAN - AGGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

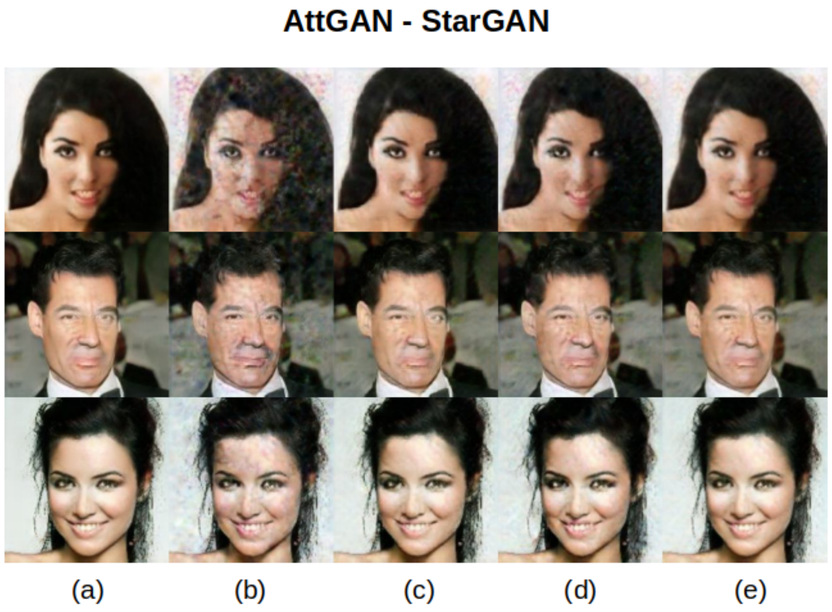


Figure B.6: Reconstruction results obtained for the method outlined in [2] using an AttGAN - StarGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

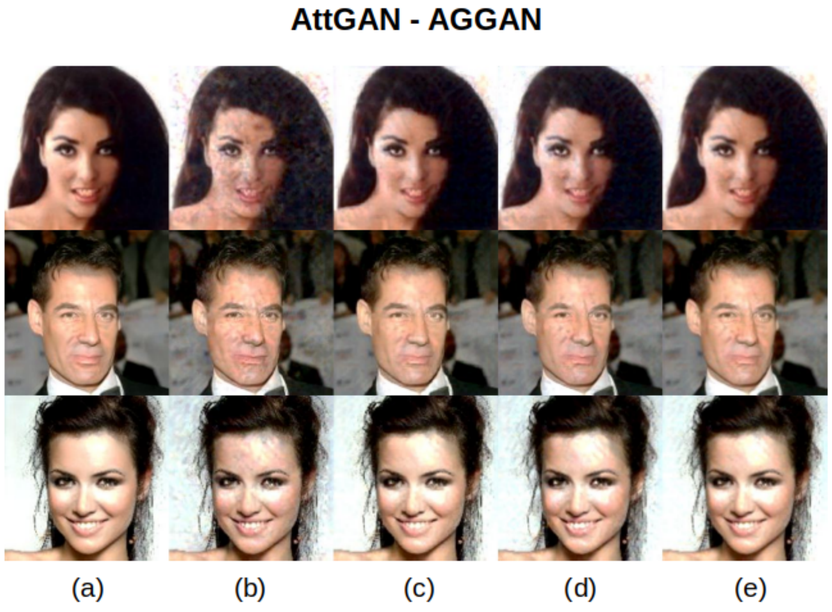


Figure B.7: Reconstruction results obtained for the method outlined in [2] using an AttGAN - AGGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

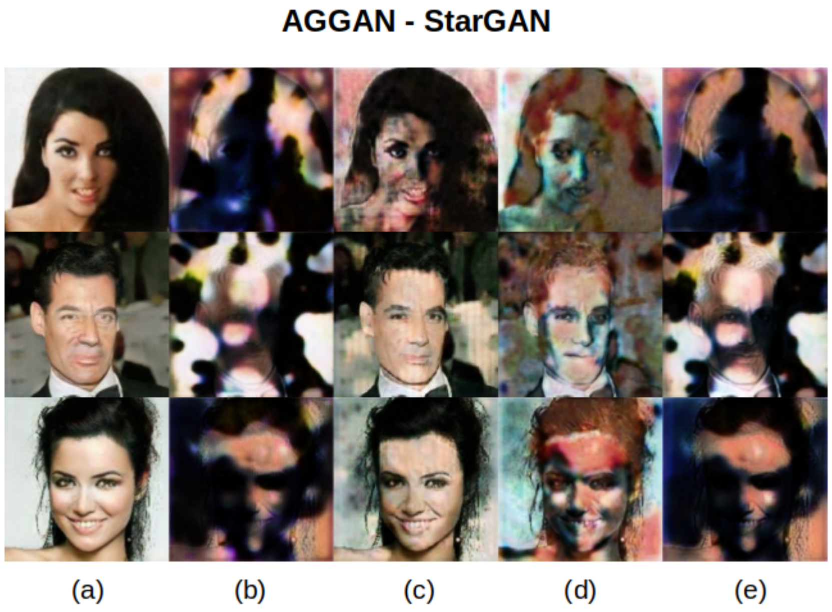


Figure B.8: Reconstruction results obtained for the method outlined in [2] using an AGGAN - StarGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

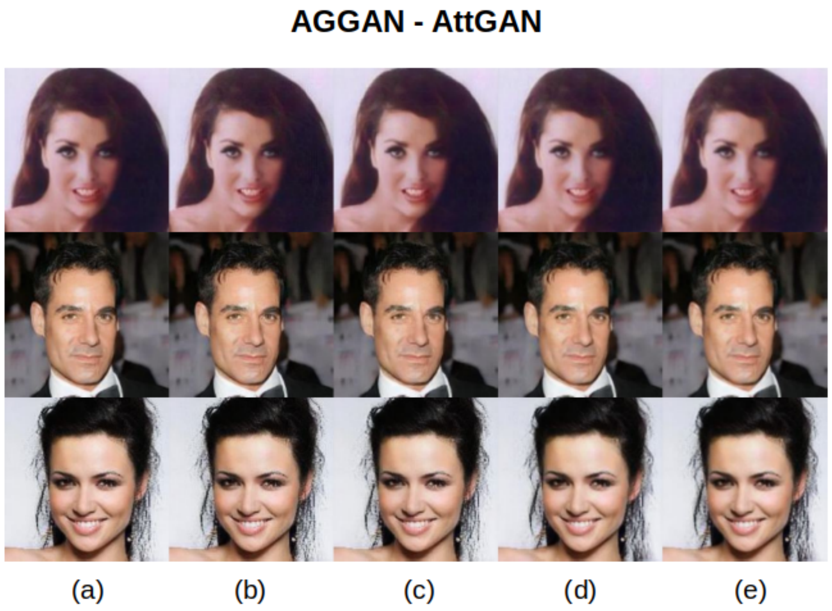


Figure B.9: Reconstruction results obtained for the method outlined in [2] using an AGGAN - AttGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

B.2.2 Gray-Box Attack

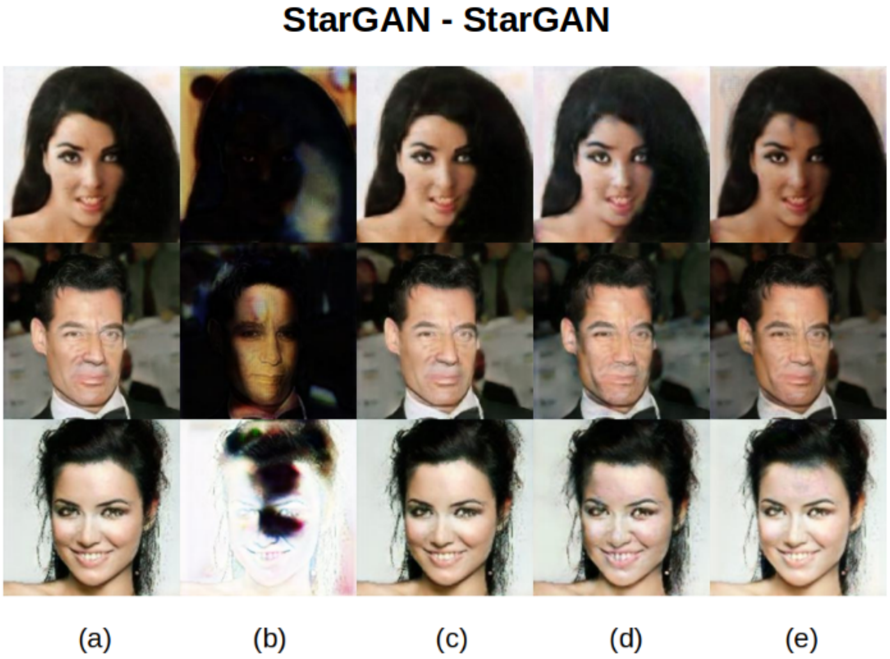


Figure B.10: Reconstruction results obtained for the method outlined in [17] using a StarGAN - StarGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

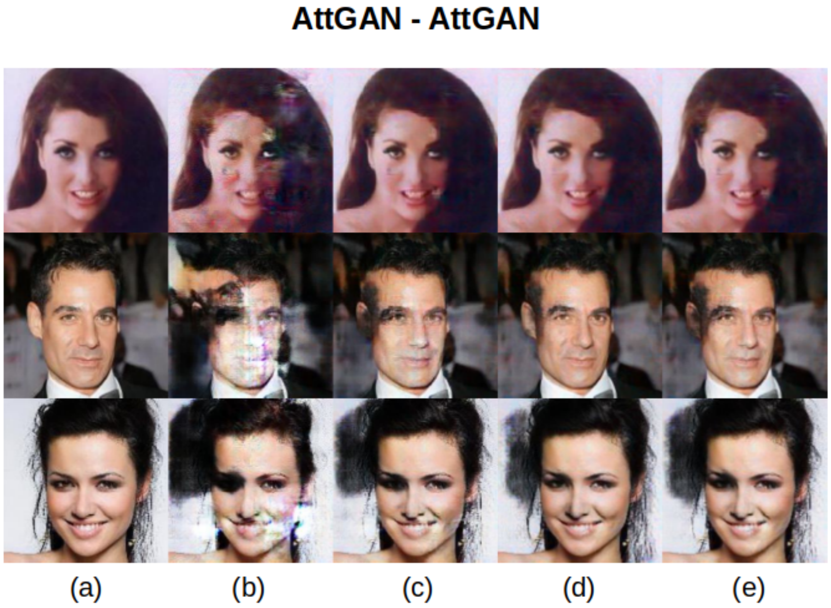


Figure B.11: Reconstruction results obtained for the method outlined in [17] using an AttGAN - AttGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

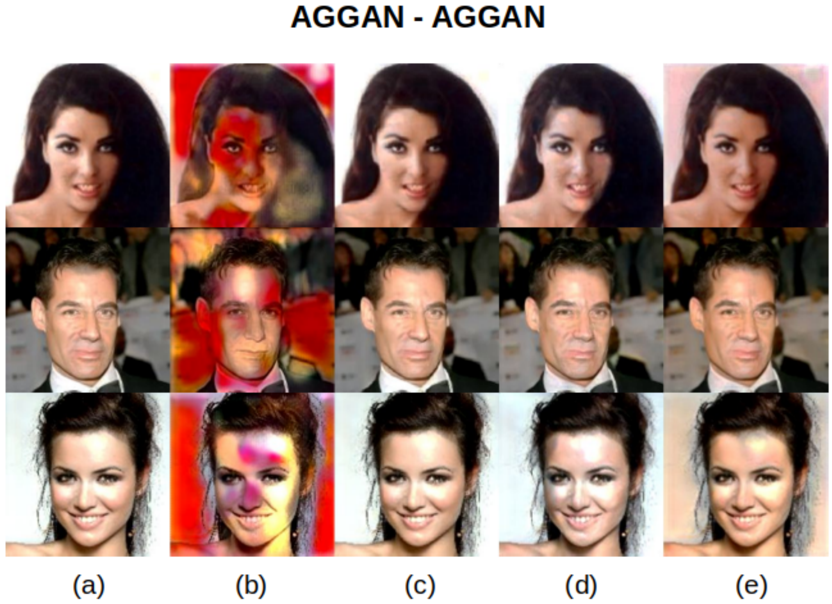


Figure B.12: Reconstruction results obtained for the method outlined in [17] using an AGGAN - AGGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

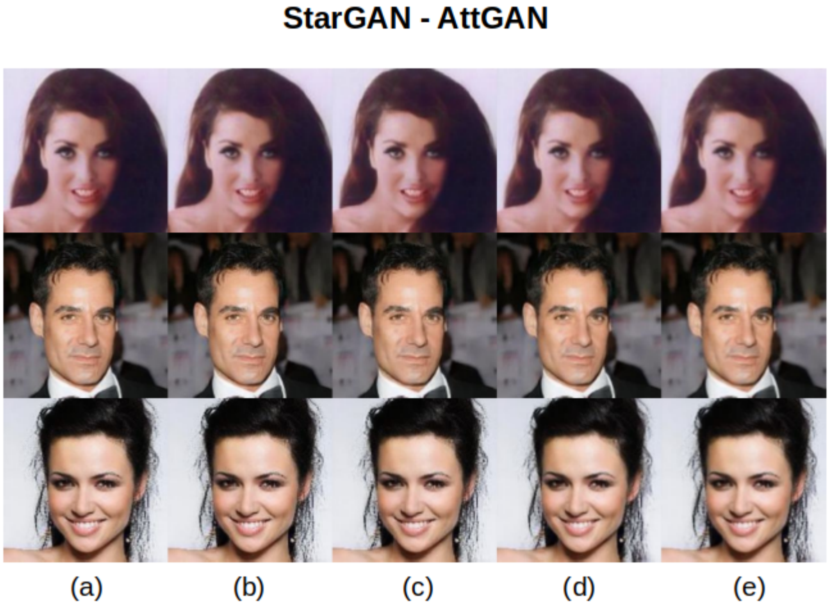


Figure B.13: Reconstruction results obtained for the method outlined in [17] using a StarGAN - AttGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

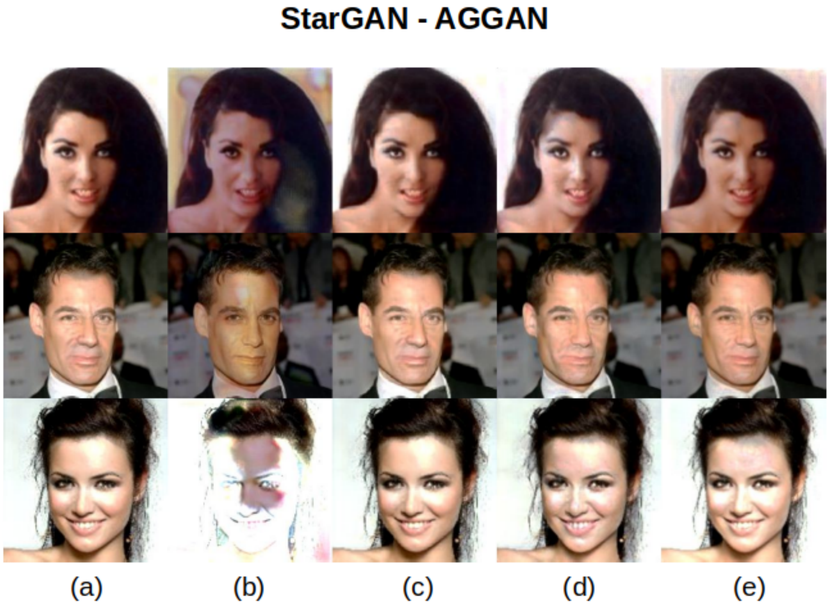


Figure B.14: Reconstruction results obtained for the method outlined in [17] using a StarGAN - AGGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

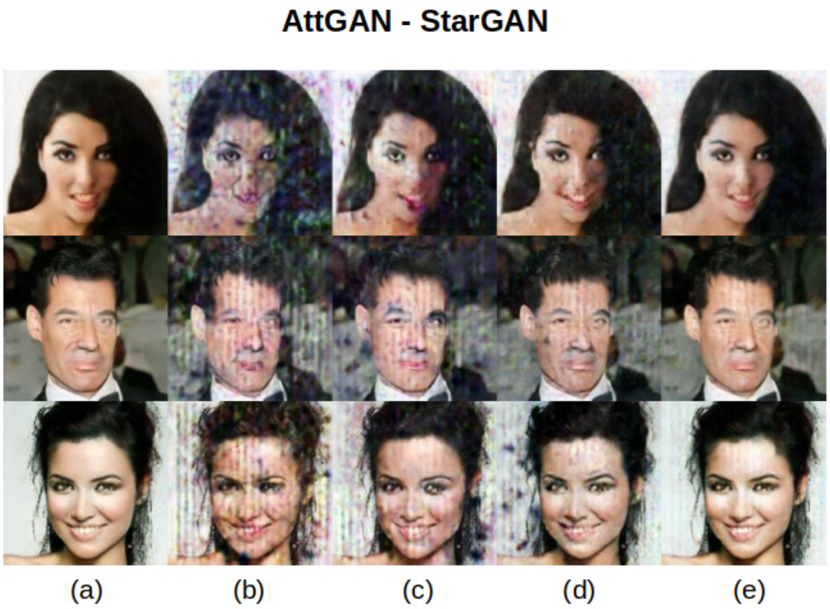


Figure B.15: Reconstruction results obtained for the method outlined in [17] using an AttGAN - StarGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

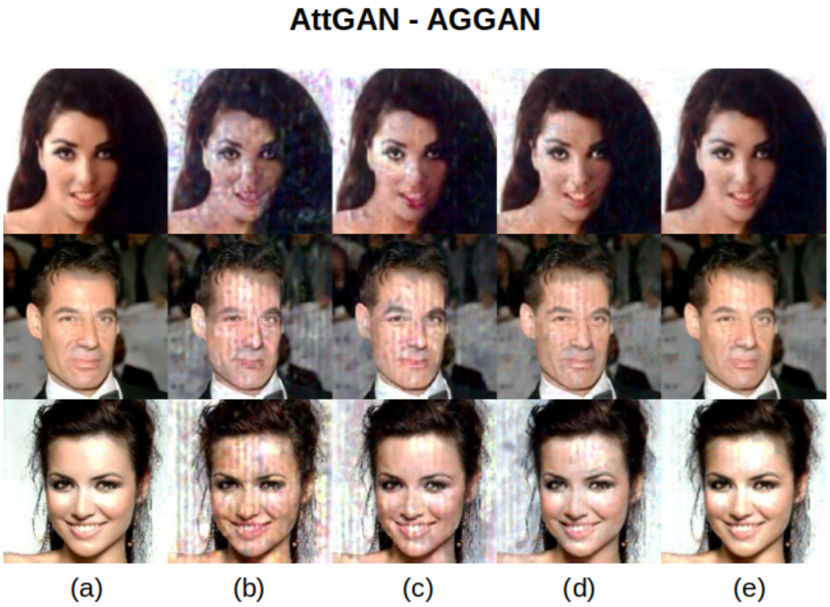


Figure B.16: Reconstruction results obtained for the method outlined in [17] using an AttGAN - AGGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

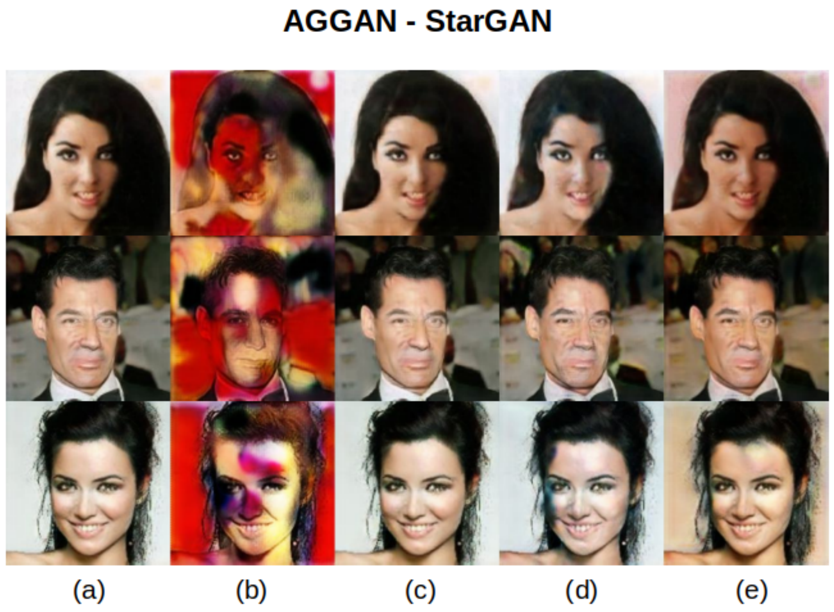


Figure B.17: Reconstruction results obtained for the method outlined in [17] using an AGGAN - StarGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

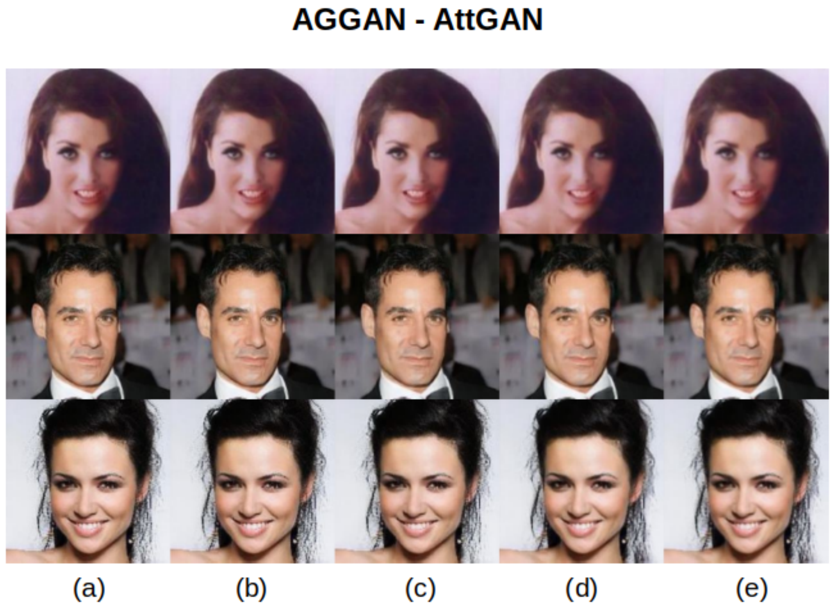


Figure B.18: Reconstruction results obtained for the method outlined in [17] using an AGGAN - AttGAN configuration and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

B.2.3 Black-Box Attack

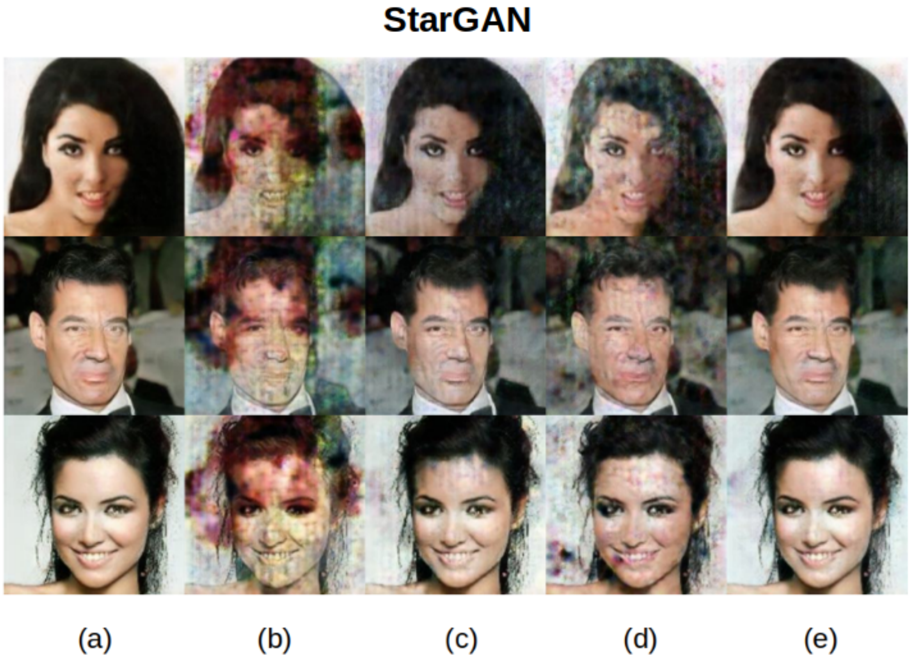


Figure B.19: Reconstruction results obtained for the method outlined in [18] using the StarGAN model and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

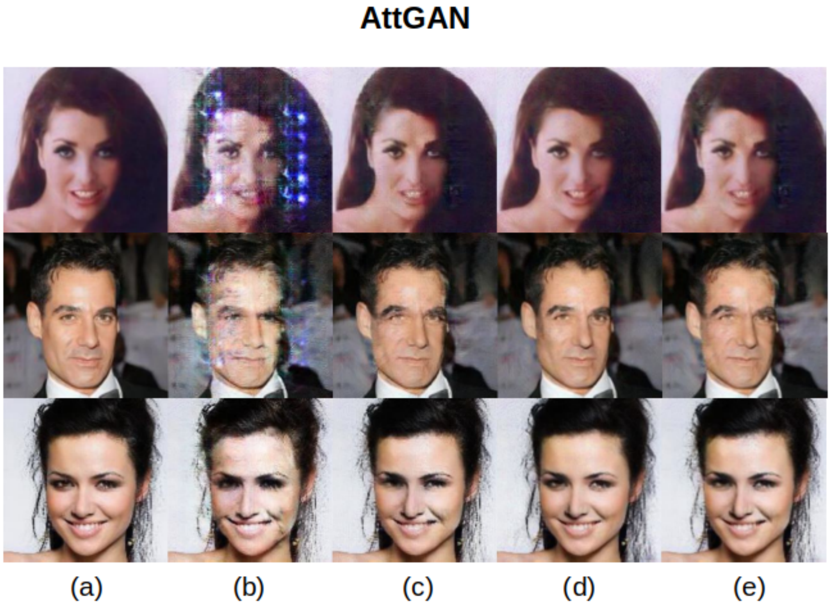


Figure B.20: Reconstruction results obtained for the method outlined in [18] using the AttGAN model and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.

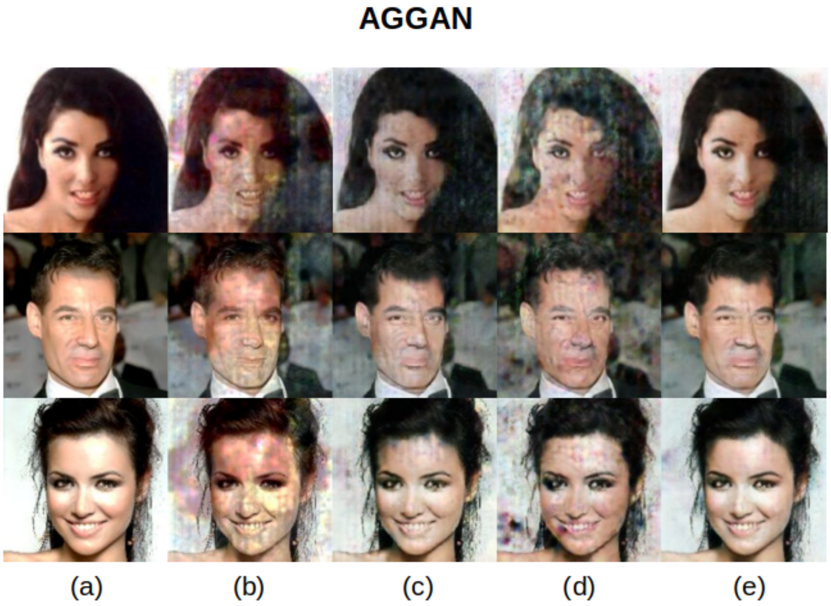


Figure B.21: Reconstruction results obtained for the method outlined in [18] using the AGGAN model and a black-hair attribute. **(a)** - DeepFake of the original image; **(b)** - DeepFake of the adversarial image; **(c)** - DeepFake of the reconstructed image using the technique outlined in [19]; **(d)** - DeepFake of the reconstructed image using the median filter; **(e)** - DeepFake of the reconstructed image using the upsampling and downsampling technique.