BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF INFORMATION TECHNOLOGY
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

USE OF DIFFUSION MODELS IN DEEPFAKES

VYUŽITIE DIFFUSION MODELOV V OBLASTI DEEPFAKES

BACHELOR'S THESIS
BAKALÁŘSKÁ PRÁCE

AUTHOR **DOMINIK TRÚCHLY**
AUTOR PRÁCE

SUPERVISOR **Ing. TOMÁŠ LAPŠANSKÝ,**
VEDOUCÍ PRÁCE

BRNO 2024

BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Bachelor's Thesis Assignment

156464

Institut:	Department of Intelligent Systems (DITS)
Student:	**Trúchly Dominik**
Programme:	Information Technology
Title:	**Use of Diffusion Models in Deepfakes**
Category:	Security
Academic year:	2023/24

Assignment:

1. Learn about neural network models for creating synthetic content, such as deepfakes and the tools that use these models.
2. Learn about the technology of diffusion models and the tools that these models use for creating or improving the plausibility of deepfakes.
3. Identify the tools used to detect deepfakes and run at least two.
4. Design experiments that use diffusion models, such as DifFace or others, in deepfake generation that verify the capability of detection methods to detect this synthetic media.
5. Perform the proposed experiments for each deepfakes generation tool.
6. Evaluate the potential security implications of diffusion models.

Literature:

- RATHGEB, Christian, Ruben TOLOSANA, Ruben VERA-RODRIGUEZ a Christoph BUSCH, ed. *Handbook of Digital Face Manipulation and Detection: From DeepFakes to Morphing Attacks*. Cham: Springer, 2022, 487 s. Advances in Computer Vision and Pattern Recognition. ISBN 978-3030876630. ISSN 2191-6586.
- Koo, H., & Kim, T. E. (2023). A Comprehensive Survey on Generative Diffusion Models for Structured Data. *ArXiv*. /abs/2306.04139
- Ivanovska, M., & Štruc, V. (2023). On the Vulnerability of DeepFake Detectors to Attacks Generated by Denoising Diffusion Models. *ArXiv*./abs/2307.05397
- Kim, M., Liu, F., Jain, A., & Liu, X. (2023). DCFace: Synthetic Face Generation with Dual Condition Diffusion Model. *ArXiv*. /abs/2304.07060

Requirements for the semestral defence:
Items 1. to 4. of the assignment.

Detailed formal requirements can be found at https://www.fit.vut.cz/study/theses/

Supervisor:	**Lapšanský Tomáš, Ing.**
Head of Department:	Hanáček Petr, doc. Dr. Ing.
Beginning of work:	1.11.2023
Submission deadline:	9.5.2024
Approval date:	6.11.2023

Abstract

A deepfake is a type of synthetic media created through sophisticated machine learning algorithms, particularly deep neural networks. As an example Generative adversarial neural networks (GANs), that are capable of generating images that are almost impossible for ordinary individuals to differentiate from genuine reality. Consequently, deepfake detection algorithms have been developed to address this growing concern. Leveraging advanced machine learning techniques, these algorithms analyze various features within images and videos to identify inconsistencies or anomalies indicative of manipulation. This thesis investigates the application of diffusion models, commonly utilized in digital image processing to enhance image quality by reducing noise and blurring, in bolstering the realism of deepfakes. By using these models, we test their effect on detecting deepfakes images using deepfake detectors.

Abstrakt

Deepfake je typ syntetického média vytvoreného pomocou sofistikovaných algoritmov strojového učenia, najmä hlbokých neurónových sietí. Ako príklad možno uviesť generatívne adverzné neurónové siete (GAN), ktoré sú schopné generovať obrázky, ktoré sú pre bežných jednotlivcov takmer nemožné odlíšiť od skutočnej reality. V dôsledku toho boli vyvinuté algoritmy detekcie hlbokých falošných správ, ktoré riešia tento rastúci problém. Tieto algoritmy využívajú pokročilé techniky strojového učenia a analyzujú rôzne funkcie v rámci obrázkov a videí, aby identifikovali nezrovnalosti alebo anomálie svedčiace o manipulácii. Táto práca skúma aplikáciu difúznych modelov, bežne používaných v digitálnom spracovaní obrazu na zvýšenie kvality obrazu znížením šumu a rozmazania, pre posilňovanie realizmu deepfakes. Využitím týchto modelov testujeme ich efekt na odhaľovanie deepfakes obrázkov pomocou deepfake detektorov.

Keywords

deepfake, neural networks, deepfake detection, diffusion models, biometrics systems

Klíčová slova

deepfake, neurónové siete, deepfake detekcia, difúzne modely, biometrické systémy

Reference

TRÚCHLY, Dominik. *Use of Diffusion Models in Deepfakes.* Brno, 2024. Bachelor's thesis. Brno University of Technology, Faculty of Information Technology. Supervisor Ing. TOMÁŠ LAPŠANSKÝ,

Rozšířený abstrakt

Deepfake je typ syntetického médiá, ktoré vzniká pomocou sofistikovaných algoritmov strojového učenia, hlavne pomocou hlbokých neurónových sietí. Príkladom sú generatívne adverzné neurónové siete (GAN), ktoré dokážu vytvoriť obrázky, ktoré sú pre bežných jednotlivcov takmer nemožné odlíšiť od skutočnej reality. Týmto spôsobom sa dá využiť táto technológia na realizáciu rôznych typov útokov v rôznych oblastiach populácie, či už ide o online obťažovanie, politické zasahovanie a manipuláciu s verejným názorom šírením nepravdivých informácií, alebo na vytvorenie falošných obvinení, ovplyvňovanie súdnictva či online šikana.

Vzhľadom na túto výzvu sa vyvinuli algoritmy detekcie deepfakes, ktoré majú za úlohu riešiť tento narastajúci problém. Tieto algoritmy využívajú pokročilé techniky strojového učenia na analýzu rôznych vlastností obrázkov a videí s cieľom identifikovať jemné nezrovnalosti alebo anomálie, ktoré by mohli naznačovať manipuláciu. Medzičasom však neustále pokroky v oblasti generácie deepfake obsahu komplikujú úlohu detekcie. Algoritmy sa musia neustále zdokonalovať, aby udržali krok s neustále sa meniacimi technikami vytvárania deepfakes a ich stále väčšou realistickosťou.

Difúzne modely predstavujú mocný nástroj v oblasti digitálneho spracovania obrazu a majú potenciál výrazne zlepšiť kvalitu obrázkov a videí. Tieto modely sa často využívajú na znižovanie šumu a rozmazania v obraze, čím sa dosiahne výrazné zlepšenie jeho vizuálnej kvality. Táto práca skúma aplikáciu difúznych modelov, bežne používaných v digitálnom spracovaní obrazu na zvýšenie kvality obrazu znížením šumu a rozmazania, pre podporu realizmu deepfakes. Využitím týchto modelov testujeme ich efekt na odhaľovanie deepfakes obrázkov pomocou deepfake detektorov.

Pri výbere detektorov deepfake sme sa rozhodli pre použitie dvoch open source riešení: detektor GAN obrázkov na architektúre ResNet50 a detektor FaceOnLive. ResNet50 je hlboká konvolučná neurónová sieť, ktorá je široko používaná v oblasti počítačového videnia a rozpoznávania obrazu. Je známa svojou schopnosťou efektívne rozlíšiť vzory a vlastnosti v obrázkoch a je často využívaná na riešenie problémov s klasifikáciou a detekciou objektov. FaceOnLive je ďalším open source detektorom, ktorý sme zahrnuli do našej analýzy. Tento detektor sa špecializuje na identifikáciu manipulovaných tvárí a deepfake obrázkov. Jeho ľahkosť použitia a schopnosť pracovať v reálnom čase je veľmi užitočná pri monitorovaní obsahu na internete a sociálnych médiách. Výber týchto dvoch detektorov nám umožňuje porovnať ich výkonnosť, presnosť a schopnosť detekovať rôzne typy deepfake obsahu, čo nám poskytne ucelený pohľad na ich použiteľnosť a efektivitu v boji proti šíreniu manipulatívnych médií. Okrem toho sme sa rozhodli do našej práce integrovať dva biometrické systémy: Megamatcher a FaceOnLive Face Recognition. Tieto systémy sú známe svojou schopnosťou rozpoznávať tváre a môžu prispieť k našej analýze a overovaniu autenticity obrázkov.

V prvom experimente sme využili technológie StyleGAN2 a StyleGAN3 na generovanie obrázkov pre našu analýzu. Tieto pokročilé generatívne modely sú široko používané v oblasti syntetickej grafiky a majú schopnosť vytvárať pôsobivé a realistické obrázky tváre ľudí. Po vygenerovaní obrázkov sme ich podrobili detekcii pomocou GAN image detektora a FaceOnLive, aby sme zhodnotili ich detekčnú schopnosť. Následne sme sa rozhodli vylepšiť vygenerované obrázky pomocou nástroja difúzneho modelu s názvom DifFace. Tento nástroj je navrhnutý na zlepšenie kvality obrázkov a odstránenie šumu. Po aplikácii DifFace sme znovu podrobili vylepšené obrázky detekcii, aby sme posúdili či ich kvalita a autenticita boli zlepšené a či sa stali ťažšie odhaliteľnými detekčnými algoritmami.

V druhom experimente sme sa zamerali na overenie toho, či samotná technológia difúznych modelov na reálnych obrázkoch nebude nesprávne identifikovaná detektormi ako deepfake. Na tento účel sme použili súbor reálnych obrázkov z FFHQ datasetu, ktoré boli podrobené aplikácii difúznych modelov na zlepšenie ich kvality a odstránenie nežiaducich artefaktov. Tento experiment nám umožnil zistiť, či aplikácia difúznych modelov na reálnych obrázkoch môže viesť k nesprávnej identifikácii týchto obrázkov ako manipulovaných a poskytol nám lepší pohľad na ich vplyv pri výsledkoch detekcie deepfake. V tomto experimente sme ďalej využili technológie biometrických systémov, aby sme zabezpečili, že obrázky po použití difúznych modelov budú čo najpodobnejšie tým, čo boli pred ich aplikáciou. Tieto biometrické systémy boli použité na analýzu tvárových rysov a charakteristík v obrázkoch pred a po aplikácii difúznych modelov. Ich úlohou bolo identifikovať a zachovať základné rysy tváre a overiť, aby transformácia obrázkov pomocou difúznych modelov nezmenila významné aspekty tváre, ktoré by mohli ovplyvniť ich identifikáciu a podobnosť s originálom.

V treťom experimente sme sa zamerali na testovanie samotnej generatívnej činnosti difúznych modelov. Pomocou nástroja Arc2Face sme vytvorili niekoľko deepfakov pre každú z našich reálnych testovacích fotiek. Po vytvorení deepfakov sme každý z týchto vygenerovaných obrázkov podrobili analýze deepfake detekcie a následnému biometrickému porovnávaniu s originálnou fotografiou. Týmto spôsobom sme mohli preskúmať efektívnosť difúznych modelov pri generovaní deepfake obsahu a ich schopnosť napodobniť skutočné tváre.

Tieto experimenty nám poskytli cenné poznatky o využití difúznych modelov v kontexte deepfake technológií a podľa ich výsledkov sme nakoniec vyhodnotili potenciálne bezpečnostné dopady difúznych modelov.

Use of Diffusion Models in Deepfakes

Declaration

I hereby declare that this Bachelor's thesis was prepared as an original work by the author under the supervision of Ing. Tomáš Lapšanský I have listed all the literary sources, publications and other sources, which were used during the preparation of this thesis.

.....
Dominik Trúchly
May 5, 2024

Acknowledgements

I extend my gratitude to my supervisor, Ing. Tomáš Lapšanský, for his guidance and regular discussions throughout this thesis. Additionally, I want to thank to my friend, Ing. Milan Šalko, who offered me support and provided valuable advice that helped me complete this work. Thank you also go to my entire family, particularly my parents, for their unwavering support during my academic journey.

Contents

Chapter 1

Introduction

In an age defined by rapid technological progress, the digital realm has seen the rise of a trend blurring the lines between what's real and what's not, known as deepfakes. Stemming from combining „deep learning“ and „fake,“ deepfakes refer to the artificial intelligence-driven synthesis of hyper-realistic images, videos, or audio recordings that convincingly simulate the appearance and behaviour of real individuals. As the accessibility of powerful computational tools and sophisticated machine learning algorithms has burgeoned, so too has the capacity for creating deceptive content with unmatched authenticity. The proliferation of deepfake technology introduces countless ethical, social, and security challenges. While it offers innovative avenues for entertainment and creative expression, it simultaneously poses a grave threat to the veracity of digital content and the trust we place in the media. The potential misuse of deepfakes for malicious purposes, such as misinformation, identity theft, and political manipulation, underscores the critical need for robust solutions in both the generation and detection domains [26].

This bachelor's thesis delves into the intricate landscape of deepfake technology, exploring the mechanisms behind their generation and the evolving methodologies employed for their detection and potential usage of diffusion models technology for generating deepfakes. Readers will gain a comprehensive understanding of the current state and future implications of deepfake technology and detection methods, empowering them to navigate its ethical and societal ramifications. In addition, they will understand the potential of diffusion models in deepfake generation and will acquire a critical perspective on the future of media manipulation.

Chapter 2 delves into the intricacies surrounding neural networks and the essential models utilized both in the creation and detection of deepfakes. Expanding upon this comprehension, in Chapter 3 we describe popular deepfake attacks and open source model StyleGAN used for creating face images. In Chapter 4, we will describe in more detail the methods of identifying deepfake images and two specific detectors that we will test. Next, we will describe the biometric systems we use in our experiments, which verify the identity of the person in the picture. In Chapter 5, we will describe the technology of diffusion models, DifFace, a tool we use for enhancing deepfake and genuine images and Arc2Face, a tool for generating facial deepfake images from scratch. Chapter 6 contains a comprehensive summary of our experiments including the design, execution, and results. Chapter 7 contains a summary of the entire work and an evaluation of the potential impacts of diffusion models for the creation of deepfakes.

Chapter 2

Neural networks

Neural networks play a role in the realm of intelligence and machine learning. These computational models draw inspiration from the organization and functionality of the brain. Neural networks play a crucial role in both the creation and identification of synthetic media.

2.1 Technical information

Neurons are the building blocks of networks. They receive inputs, perform operations on them and produce outputs. These neurons work together in layers, such as the input layer, hidden layers (if any) and output layer. Hidden layers are intermediary layers between the input and output layers. They perform the complex computations and transformations necessary to learn meaningful representations of the input data. Weights act as parameters that determine the strength of connections between neurons. These weights are adjusted dynamically during the learning process. When a neuron receives input from multiple sources, it multiplies each input by a corresponding weight and then sums up these weighted inputs. Additionally, bias is a parameter for each neuron that compensates for any offset in input data and helps enhance the network's adaptability. For example, it could allow the neuron to have some inherent activation even when all input values are zero. Activation functions are components that introduce non-linearity to networks. They enable the network to identify relationships within data. Popular activation functions include Rectified Linear Unit (ReLU) Sigmoid function and Hyperbolic Tangent (Tanh). Each of these functions contributes to the network's ability to capture patterns [22].

Mathematically the output of neuron can be calculated as follows:

$$y = \text{activation}\left(\sum_{i=1}^{n}(w_i \cdot x_i) + b\right) \tag{2.1}$$

where y represents the output of neuron, $x_1, x_2, ..., x_n$ represents inputs, $w_1, w_2, ..., w_n$ are weights and b is bias.

Forward propagation is a step, in networks where input data is passed through the network to generate an output. In each layer, the neurons perform a calculation by considering the input values, applying an activation function and then passing the result to the next layer.

During training, the loss function takes the stage as it measures the difference between predicted and actual outputs. It uses optimization algorithms, such as gradient descent, which indicates the direction of the steepest decrease of this loss function. The main goal

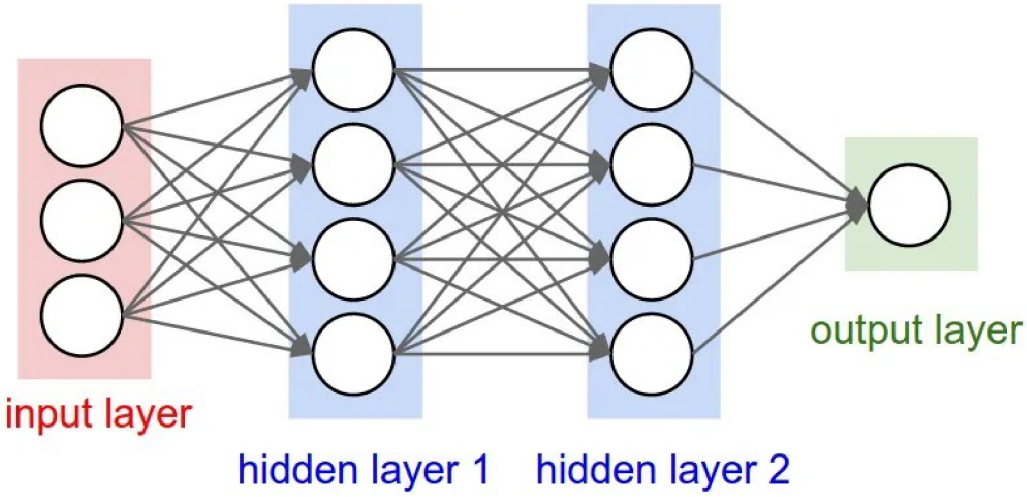

Figure 2.1: Neural network layers [7].

is to tune the model's weights and biases in order to minimize the loss. This process of improving the model is called backpropagation. Backpropagation systematically calculates how each model parameter affects the loss functions gradient starting from the output layer and moving backwards through the network. This information is then used to adjust weights and biases, gradually improving the network's accuracy with each iteration [5].

This is the way Mirsky and Lee mathematically defined neural networks [23]. Let $l^{(i)}$ denote the i-th layer in the network M, and let $||l^{(i)}||$ denote the number of neurons in $l^{(i)}$. Finally, let the total number of layers in M be denoted as L. The weights which connect $l^{(i)}$ to $l^{(i+1)}$ are denoted as the $||l^{(i)}||$-by-$||l^{(i+1)}||$ matrix $W^{(i)}$ and $||l^{(i+1)}||$-dimensional bias vector $\tilde{\mathbf{b}}^{(i)}$. Finally, we denote the collection of all parameters θ as the tuple $\theta \equiv (W, b)$, where W and b are the weights of each layer, respectively. Let $a^{(i+1)}$ denote the output (activation) of layer $l^{(i)}$ obtained by computing $f\left(W^{(i)} \cdot \tilde{\mathbf{a}}^{(i)} + \tilde{\mathbf{b}}^{(i)}\right)$, where f is often the Sigmoid or ReLU function. To execute a network on an n-dimensional input $\mathbf{x}$, a process known as forward-propagation is performed, where $\mathbf{x}$ is used to activate $l^{(1)}$ which activates $l^{(2)}$ and so on until the activation of $l^{(L)}$ produces the m-dimensional output $\mathbf{y}$.

2.2 Convolutional Neural Networks

Convolutional Neural Networks [28](CNNs) are designed with an architecture to effectively capture patterns and extract intricate features from visual data. The unique design of CNNs makes them particularly effective in computer vision tasks, such as image recognition, object detection, and image segmentation. This network is made out of several different layers: input layer (consists of neurons corresponding to the pixels of the input image), convolution layer, pooling layer, fully-connected layer and output layer (produces the final results or predictions based on the learned representations). Each type of layer performs specific operations and plays a crucial role in the feature extraction and representation learning process. The convolutional layer will determine the output of neurons which are connected to local regions of the input through the calculation of the scalar product between their weights and the region connected to the input volume. It involves a set of learnable filters or kernels that slide across the input data. Each filter is a small field that captures

local patterns or features, such as edges, textures or shapes. The filter's weights are learned during the training process, enabling the network to automatically discover relevant features for the given task.

As the filters move across the input data they generate feature maps that highlight specific patterns. Multiple filters are typically used in a single convolutional layer, enabling the network to learn a diverse set of features simultaneously. The depth of the output volume corresponds to the number of filters used and each element in a feature map represents how a specific filter is activated at a location. One of the advantages of convolutional layers is their ability to capture translational invariance, meaning that the network can recognize patterns regardless of their position in the input space. This characteristic is crucial for tasks like image recognition, where the position of an object in an image should not impact the model's ability to classify it [28, 14].

Pooling layers, frequently realized through techniques such as max pooling or average pooling 2.2, play a significant role in diminishing spatial dimensions. This downsizing not only improves computational efficiency but also augments the network's capacity to concentrate on critical features while discarding superfluous information. Max pooling involves selecting the maximum value from a local neighbourhood within the input feature map, commonly used when the network needs to focus on the most salient features and when translation invariance is desirable (for example, image classification tasks). On the other hand average pooling calculates the average value, providing a more smoothed representation of the features which can be suitable for tasks where exact localization is crucial [38].

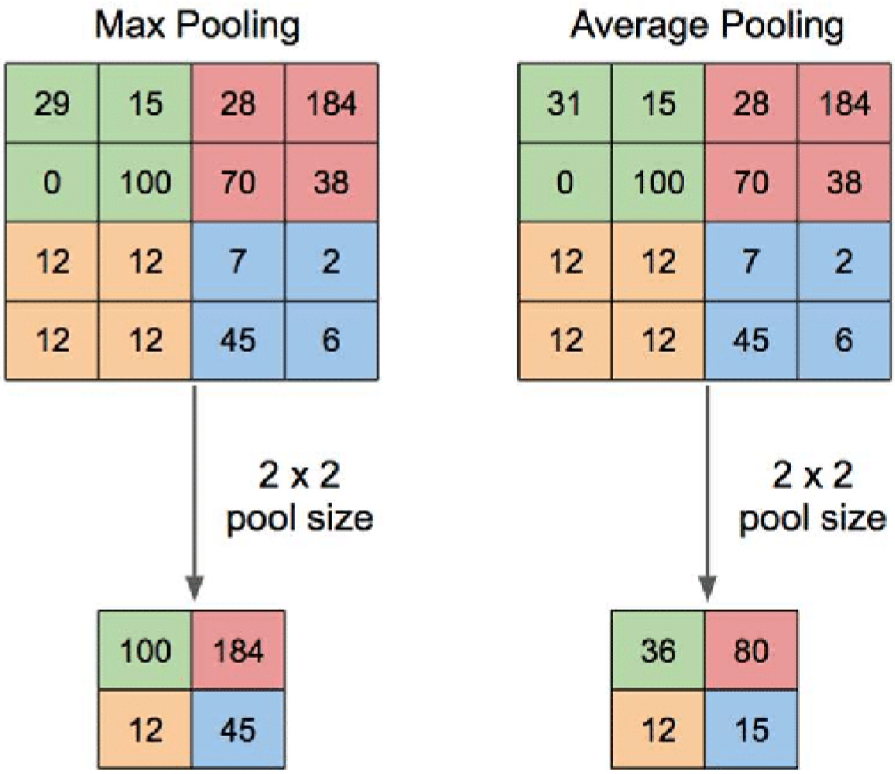

Figure 2.2: Illustration of Max Pooling and Average Pooling [38].

Towards the end of the network, fully connected layers bring together the extracted features and enable high-level predictions. This architecture allows CNNs 2.3 to adapt easily for visual recognition tasks such, as image classification, object detection and segmentation.

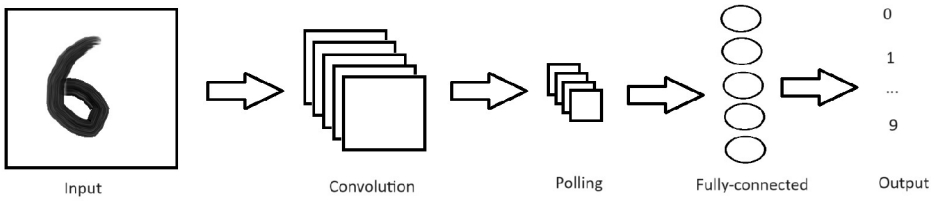

Figure 2.3: A simple CNN architecture [28].

2.3 Generative adversarial networks

Generative Adversarial Networks (GANs) were introduced by Ian Goodfellow and colleagues in 2014 [10]. They are renowned for their ability to generate realistic synthetic data. GANs consist of two neural networks: a generator and a discriminator – engaged in a cooperative and competitive learning process.

The generator creates data instances by transforming random noise into samples, initially far from authentic data. Through iterative training, it learns to produce increasingly realistic outputs. Simultaneously the discriminator distinguishes between genuine and synthetic data, continuously improving its discrimination abilities. Throughout the training process, the model undergoes a series of fine-tuning adjustments using backpropagation and gradient descent. The generator minimizes the probability of the discriminator correctly classifying its samples as fake, while the discriminator maximizes accuracy in distinguishing between real and synthetic data. This iterative process continues until a stable balance is achieved, indicated by the high quality of generated samples and the discriminator facing difficulties in distinguishing them from actual ones [1].

2.4 U-net

Convolutional networks are usually employed for classification tasks, but in scenarios where localization is crucial, such as assigning a class label to individual pixels, U-Net emerges as a powerful solution. Its unique architecture, with a focus on both feature extraction and spatial preservation, makes U-Net especially adept at tasks demanding pixel-level classification and intricate pattern recognition. Moreover, the network's capacity to learn and reproduce complex patterns contributes to the generation of realistic synthetic content in diffusion models [34, 27].

The U-Net architecture comprises a contracting path on the left and an expansive path on the right. The contracting path follows a standard convolutional network structure, employing repeated 3x3 convolutions, ReLU activation, and 2x2 max pooling for downsampling, with a doubling of feature channels at each step. The expansive path involves

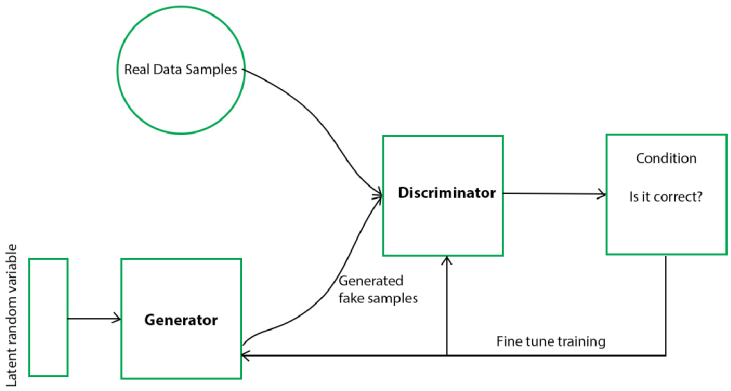

Figure 2.4: Generative Adversarial Networks architecture [1].

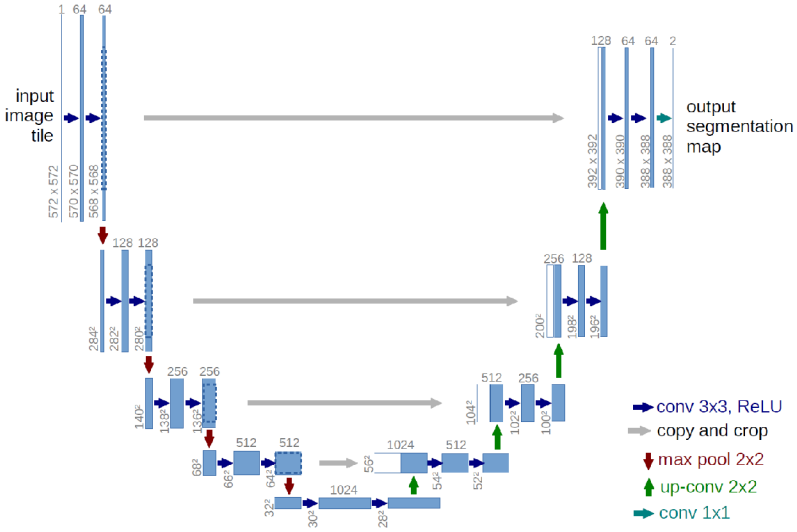

Figure 2.5: U-net architecture [34].

upsampling, a 2x2 convolution for channel reduction, concatenation with the cropped feature map from the contracting path, and two 3x3 convolutions with ReLU activation. A 1x1 convolution at the final layer maps each 64-component feature vector to the desired class count. With 23 convolutional layers, the network ensures seamless tiling of the output segmentation map, requiring careful consideration of input tile size to maintain even x- and y-size during 2x2 max-pooling operations [34].

Chapter 3

Deepfakes creation

The term „deepfake“ is derived from the combination of „deep learning“ and „fake“ emphasizing the use of deep neural networks to manipulate or generate visual and audio content that can look authentic in the human‘s eyes. This technological advancement has raised significant concerns due to its potential for misuse and the challenges it poses to traditional methods of content verification. At the core of deepfake technology is the utilization of deep neural networks, specifically generative adversarial networks (GANs) and autoencoders.

3.1 Deepfake attacks

Deepfakes can be used in many different areas and among the most used attacks are different types of deepfakes of human faces. In order to gain a better understanding of deepfakes and the potential harm that can be caused by their abuse, we will delve into a variety of attacks in this section. The examples below show some of the most frequently encountered ones, but there can be many more beyond those mentioned [9, 4].

1. The spread of Misinformation and Fake News: Deepfakes have the potential to generate convincing content that depicts political or other important figures being in places where they have never been or doing something inappropriate. This could have an impact on elections or public opinion by manipulating people's perceptions. Moreover, there is a concern that individuals could be portrayed as endorsing products, ideologies, or causes without their knowledge or consent.

For example, just a few days leading up to a crucial election in Slovakia, which would determine the country's leadership, a fake audio recording began circulating online. In the recording, one of the leading candidates appeared to brag about having manipulated the election [6].

2. Damage to one's reputation: Deepfakes can be employed to create fabricated content that shows individuals engaging in criminal activities leading to harm to their reputation. Innocent people might find themselves wrongly accused of actions they did not commit due to manipulated content.

3. Extortion and blackmail: Perpetrators could utilize deepfake videos that portray individuals in compromising situations as a means of extorting or blackmailing them.

4. Unauthorized access through identity theft: Deepfakes may be utilized as a way of bypassing voice or facial recognition systems allowing unauthorized individuals to gain access to information.

In this article, an employee in the finance department of a multinational corporation was deceived into transferring $25 million to scammers who employed deepfake technology to impersonate the company's chief financial officer during a video conference [13].

5. Exploitation and harassment: Deepfakes can be used to create explicit content featuring individuals without their consent, contributing to online harassment.

6. Fake Profiles and catfishing: Deepfakes can contribute to the creation of fake social media profiles with realistic face photos. This tactic is often used for catfishing, where individuals engage in deceptive online relationships.

This article describes how Facebook and Twitter must massively delete fake profiles that have started to appear on these platforms. They are using GAN technology to generate fake profile pictures and make them look trustworthy [29].

3.2 StyleGAN

In this section we will describe StyleGAN, a powerful deepfake creation tool used to generate faces from scratch. Later we will use the pictures generated by this tool in our experiments.

StyleGAN, developed by NVIDIA, serves as a tool for the creation of high-quality and lifelike images. The StyleGAN generator architecture undergoes a significant redesign, introducing innovative mechanisms for precise control over the image synthesis process. It begins with a learned constant input, dynamically adjusting the image 'style' at each convolutional layer based on the latent code. This means it can change things like the pose, lighting, or facial features of the generated image at each layer. Additionally, the method incorporates noise directly into the network, contributing to the automatic and unsupervised separation of high-level attributes (e.g., pose, identity) from stochastic variations (e.g., freckles, hair) in the generated images. These adjustments facilitate intuitive scale-specific mixing and interpolation operations. Importantly, these modifications are confined to the generator, leaving the discriminator and loss function unaltered. This approach stands independently from current discussions on GAN loss functions. The generator embeds the input latent code into an intermediate latent space, significantly influencing how factors of variation are represented within the network [16].

StyleGAN2[1]is still the most used and widespread version of StyleGAN. One notable application is showcased in websites like „thispersondoesnotexists.com“[2]. Using StyleGAN2 implementation, it generates convincing, entirely fictional faces with each page refresh. StyleGAN2 introduces modifications to the generator architecture, including skip connections, which help in better preserving the high-frequency details in generated images. This results in sharper and more realistic outputs. It also offers faster training times compared to its predecessor, training up to 40% faster at 61 images per second [17].

The newest version, StyleGAN3[3]focuses on the successful elimination of all sources of positional references that can cause aliasing. Aliasing becomes especially apparent when

[1] https://github.com/NVlabs/stylegan2
[2] https://thispersondoesnotexist.com/
[3] https://github.com/NVlabs/stylegan3

Figure 3.1: High-quality human faces generated by StyleGAN3.

rotating an image. When pixels appear fixed to particular positions within the image, their rotation appears unnatural. By substituting the input constant in StyleGAN 2 with Fourier features offers the added benefit of inherently defining an infinitely spatial map. This adjustment not only enhances the outcomes but also aids in computational processes[15, 19]. Overall, StyleGAN3 is one of the most modern and high-quality deepfake generators that is available for free.

3.3 FFHQ dataset

The FFHQ (Flickr-Faces-HQ) dataset released by NVIDIA in 2019 is a high-quality and diverse collection of facial images designed for training and evaluating generative adversarial networks (GANs), particularly for the generation of realistic human faces. The dataset is curated from images available on the Flickr platform, a popular online photo-sharing service and only images under permissive licenses were collected. It consists of 70,000 high-quality PNG images at 1024×1024 resolution and contains considerable variation in terms of age, ethnicity and image background. It also has good coverage of accessories such as eyeglasses, sunglasses, hats, etc [3].

Images from this dataset were used for a pre-trained version of StyleGAN2 and StyleGAN3 and also for the pre-trained version of DifFace (one of our tested diffusion models, explained in chapter5). For specific segments of our experiments requiring authentic human faces, will these pictures be selected from the FFHQ dataset[4].

[4] https://github.com/NVlabs/ffhq-dataset

Chapter 4

Deepfake detection

Detecting deepfakes is a part of the effort to combat the spread of manipulated multimedia content. With advancements in deepfake technology, it's becoming more important than ever to have reliable detection methods in place. Different strategies and techniques have been devised to recognize and minimize the damage caused by deepfake content. One prevalent method for detecting deepfakes involves leveraging artificial intelligence (AI) and machine learning algorithms. These algorithms are trained on vast datasets containing both authentic and manipulated media to learn the subtle differences between genuine and fake content.

In the pursuit of identifying manipulated content within images, Convolutional Neural Networks (CNNs) emerge as a cornerstone in contemporary deepfake detection methodologies. CNNs are particularly well-suited for this task due to their inherent ability to discern complex visual patterns and dependencies. These neural networks operate by employing convolutional layers, allowing them to effectively capture hierarchical features within the image data. The core of CNN-based deepfake detection lies in the training process. An extensive dataset encompassing both authentic and manipulated images is essential to enable the model to learn the nuanced distinctions between the two. During training, the CNN refines its internal parameters to recognize subtle artifacts, inconsistencies, and irregularities that may be indicative of image manipulation. A pivotal element in fake image detection involves scrutinizing facial features and expressions [31, 36].

Given that many deepfakes primarily target facial attributes, the CNN is configured to focus on intricate details such as facial landmarks, skin texture, and expression dynamics. Detecting fake images involves examining subtle errors that may occur during the generation process. Despite advancements in deepfake technology, certain common and noticeable anomalies can serve as indicators of manipulation. Some of the most common are inconsistent lighting and shadows pose, blurry edges and artifacts along the boundaries of the face, inconsistencies in reflections (especially in the eyes, glasses, or other reflective surfaces) or simply misalignment of facial features such as eyes, nose, mouth or eyebrows.

4.1 Face liveness detector FaceOnLive

FaceOnLive is a biometrics authentication and ID verification solution provider offering robust security solutions with advanced facial recognition, liveness detection, and ID document recognition technologies [8]. They offer several free and easy-to-use APIs and mobile application. Their Face liveness tool is available on the internet where the user after reg-

istering is able to test up to 1000 images per month for free. The API[1] is very intuitive and all you have to do is upload a picture from your computer and send it as a request. After that, you obtain a response containing information about the face from the picture, the result and detected liveness score in the form of JSON. The most common results are deepfake with a score set to -600, genuine with a score from 0 up to 20 and spoof with low negative number values. Spoof pictures are usually created with basic editing tools like Photoshop, or editing existing images with simple modifications, whereas deepfake pictures are generated using advanced AI algorithms to synthesize entirely new content. The only drawback is the absolute value of -600 for all deepfake results instead of some value from which we could get more information about the real credibility of the image. In Figure 4.1, we can see the result of the evaluation returned by this detector in the form of JSON.

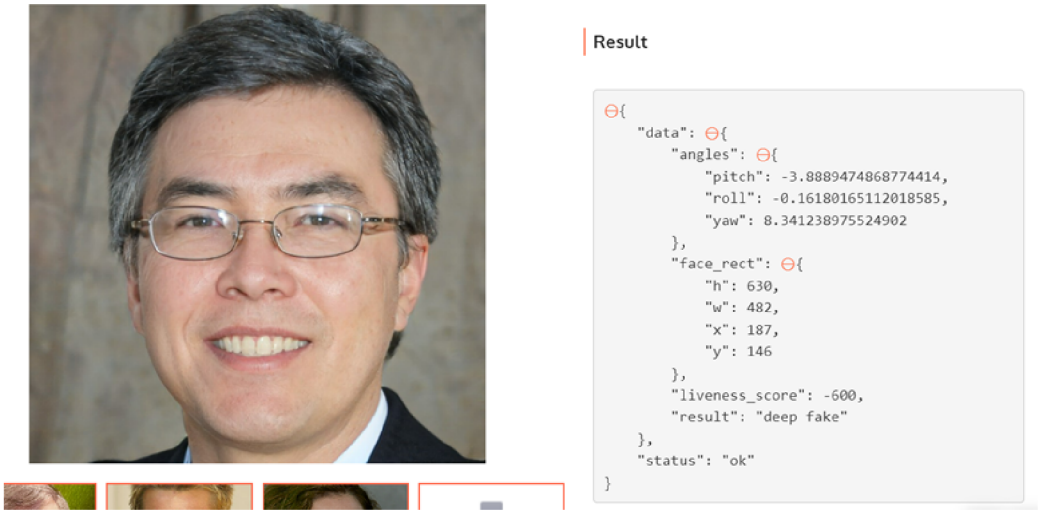

Figure 4.1: Face liveness detection for deepfake picture.

4.2 GAN image detector (ResNet50 NoDown)

ResNet50 is a convolutional neural network (CNN) architecture that was introduced by researchers at Microsoft Research in their paper titled „Deep Residual Learning for Image Recognition“ in 2015. The „50“ in ResNet50 refers to the number of layers in the network. Specifically, ResNet50 consists of 48 convolutional layers, one max pool layer, and one average pool layer. ResNet50 was designed as part of the ResNet family of models, which introduced the concept of residual learning to address the vanishing gradient problem in very deep neural networks. Traditional deep networks aim to directly learn the underlying mapping functions between input and output. As networks deepen, gradients tend to diminish during backpropagation, making it difficult to update parameters effectively. When this occurs, the weights of the early layers in the network are updated very slowly or not at all and as a result, the network fails to learn meaningful representations of the data, leading to poor performance. To address this, ResNet introduces the concept of residual learning. Instead of directly fitting underlying mapping functions, ResNet learns residual functions with reference to the layer inputs. Mathematically, if x represents the input to a

[1] https://rapidapi.com/organization/faceonlive-inc

given layer, and $H(x)$ is the desired underlying mapping, ResNet aims to learn the residual function $F(x) = H(x) - x$ [12].

The fundamental unit of ResNet50 is the residual block. Each residual block consists of two convolutional layers, each followed by batch normalization and ReLU activation. The input to the block is passed through the first convolutional layer, undergoes normalization and activation, and then enters the second convolutional layer. Importantly, ResNet50 introduces shortcut connections (or skip connections) that directly feed the input of a block to its output. This shortcut connection allows the network to learn residuals with respect to the identity function, facilitating gradient flow and easing the optimization of deep networks. ResNet50 NoDown architecture does not perform down-sampling in the first layer. This allows to retain more spatial information in the early stages of the network. Overall accuracy for GAN generated images is always above 90% irrespective of the type of architecture and further improves (above 97%) if training is carried out on StyleGAN2 [11, 12].

This detector can be freely tested at the University of Buffalo web site[2]. They also provide a code with pre-trained models for ProGAN and StyleGAN2 (trained on 720000 fake images) available at github[3]. The script tests the network on the image folder and collects the results in a CSV file with „logit“ values. A logit value determines how certain the detector is of its result, the higher the value from zero, the more certain it is that it is a deepfake, and the smaller the value than 0, the more certain it considers the photo to be genuine.

4.3 Face recogition FaceOnLive

Face Recognition API[4] is by the same manufacturer as FaceOnLive Face liveness detector. It is also available for free up to 1000 pictures. All you have to do is upload 2 images, and after a few seconds, you get the JSON result, whether the faces match or not. In addition you get a percentage similarity rate of uploaded pictures. If the similarity rate is above or equals 70%, faces are evaluated as the same person.

4.4 Biometric system Megamatcher

MegaMatcher Automated Biometric Identification System [24] is a complete system for the deployment of large-scale multi-biometric projects that is intended for biometric voter registration with records deduplication, passport issuing, border control, as well as other civil or criminal AFIS/ABIS.

The effectiveness of MegaMatcher face recognition heavily depends on the quality of images during enrollment. It is essential to maintain a minimum distance of 32 pixels between the eyes for reliable template extraction, with a recommended distance of 64 pixels for superior results. The face recognition engine accommodates various facial postures, allowing a head roll range of up to ±180 degrees (default ±15 degrees), head pitch within ±15 degrees (expandable to ±25 degrees), and head yaw within ±90 degrees. It is advisable to avoid smaller yaw tolerances unless constrained by system limitations. Enrolling multiple perspectives of the same face comprehensively covers the entire ±90 degrees yaw range [25].

[2] https://zinc.cse.buffalo.edu/ubmdfl/deep-o-meter/landing_page

[3] https://github.com/grip-unina/GANimageDetection

[4] https://rapidapi.com/faceonlive-inc-faceonlive-inc-default/api/face-recognition26/

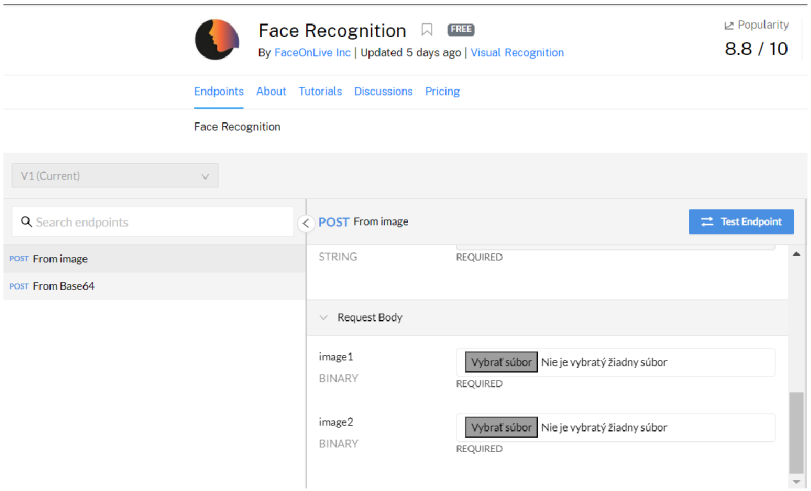

Figure 4.2: Faces uploading at Face recognition API.

The application is available online [5] where after registration and activation can be launched a free trial for one month. To confirm a person's identity, the person has to be enrolled with his biometric data and photos. After that, in the Identify window, you can add pictures of faces you want to compare with, and in a few seconds of evaluation, you get the results of whether the faces match or not. In the result, there is also a score, but even when comparing the same photo of a person, it is almost a random value from 0 to 200, and according to the information from developers, we could not find out what exactly the score means, so we will not take it into account. They also provide a downloadable demo application for easier testing of entire directories.

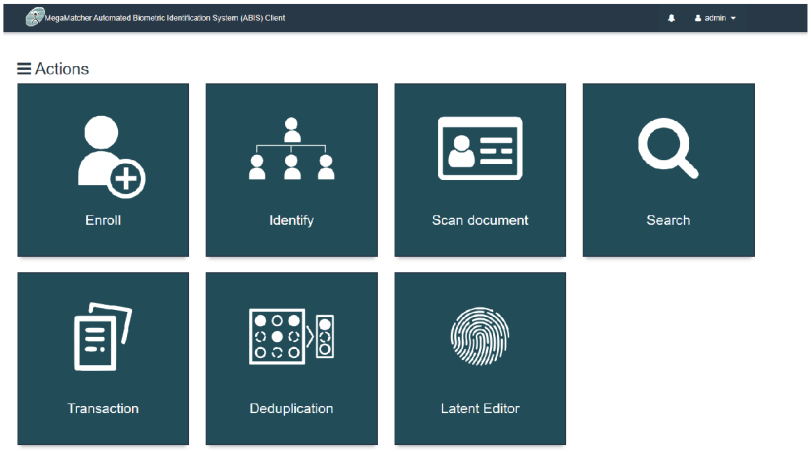

Figure 4.3: Megamatcher ABIS user interface.

[5] https://www.megamatcher.online/

Chapter 5

Diffusion models

Diffusion models belong to a category of denoising generative models designed to generate an image from random noise by progressively removing noise over time. They serve as generative models, tasked with generating data resembling their training data, but unlike traditional generative models that directly model the data distribution, diffusion models approach the problem by considering a sequence of transformations applied to a simple initial distribution. This sequence of transformations, known as the diffusion process, gradually transforms the initial distribution into the desired target distribution. They operate by introducing Gaussian noise incrementally to the training data, subsequently learning to reverse this process. Notably, Diffusion Models offer advantages such as eliminating the need for adversarial training and addressing documented challenges associated with it. Furthermore, they exhibit scalability and parallelizability, enhancing training efficiency [27].

Diffusion models involve a forward process where controlled Gaussian noise is progressively added to an input image. The noise is added at each time step t, according to the equation:

$$q(x_t|x_{t-1}) := \mathcal{N}\left(x_t; (1-\beta_t)x_{t-1}, \beta_t I\right), \quad \forall t \in \{1, \ldots, T\} \tag{5.1}$$

Here, x_t represents the image at time step t, β_t is a variance schedule controlling the amount of noise added at each step, and I is the identity matrix with the same dimension as the input data x_0 [21].

In the reverse process, a model aims to predict the initial noise from the corrupted image. The loss function $\mathcal{L}$ quantifies the discrepancy between the predicted noise and the initial noise:

$$\mathcal{L} = \mathbb{E}_{t, X_0, \epsilon}\Big[\big\|\epsilon_\theta\big(\underbrace{\sqrt{\alpha_t}X_0 + \sqrt{1-\alpha_t}\epsilon}_{X_t}, t\big) - \epsilon\big\|_2^2\Big] \tag{5.2}$$

Here, $\epsilon_\theta(X_t, t)$ represents the model used for predicting the noise, ϵ is the initial noise, and $\alpha_t := 1 - \beta$ is a scheduling scalar controlling the balance between the original image and the added noise. [18, 21].

The only requirement for the model is that its input and output dimensionality must be identical. Because of that, diffusion models are often implemented with the U-Net architecture, which allows them to effectively capture and reconstruct intricate details in complex data distributions.

One of the key advantages of diffusion models in image generation lies in their ability to capture complex dependencies and intricate details present in natural images. By iteratively denoising the input noise, diffusion models effectively learn the underlying structure of the data distribution, allowing them to generate images with realistic textures, shapes, and patterns. This capability is particularly valuable in applications where generating visually coherent and lifelike images is essential, such as in computer graphics, artistic rendering, and virtual reality. Furthermore, diffusion models can be tailored to specific image generation tasks by adjusting parameters such as the level of noise added at each step and the number of steps in the diffusion process. This flexibility allows practitioners to fine-tune the model's behaviour and generate images with desired characteristics, such as varying levels of noise, resolution, or style. Additionally, diffusion models can be conditioned on additional input information, such as class labels or semantic maps, enabling controlled generation of images belonging to specific categories or exhibiting certain attributes [20, 35].

Beyond image generation and denoising, diffusion models have demonstrated versatility in various other domains. For instance, researchers have explored their applications in audio generation, where they have shown promise in synthesizing realistic sounds and music. Additionally, diffusion models have been adapted for text generation tasks, including language modeling and text completion. These extensions highlight the flexibility of diffusion models and their potential to be applied across a wide range of signal processing and generative modeling tasks [32, 37].

5.1 DifFace

DifFace is a diffusion model framework that we decided to use for our deepfake-generated images. Its main purpose is blind face restoration. This refers to the process of reconstructing an accurate and clear representation of a person's face from an input image that is degraded or of low quality, such as containing noise or blur. The creators Zongsheng Yue and Chen Change Loy stated [39] that difFace is capable of gracefully handling unseen and complex degradations without the need for complicated loss designs. The key lies in the posterior distribution from the input low-quality (LQ) image to its high-quality (HQ) counterpart. Specifically, DifFace utilizes a transition distribution from the LQ image to the intermediate state of a pre-trained diffusion model. This transitional information is then gradually transmitted from the intermediate state to the HQ target by recursively applying the pre-trained diffusion model.

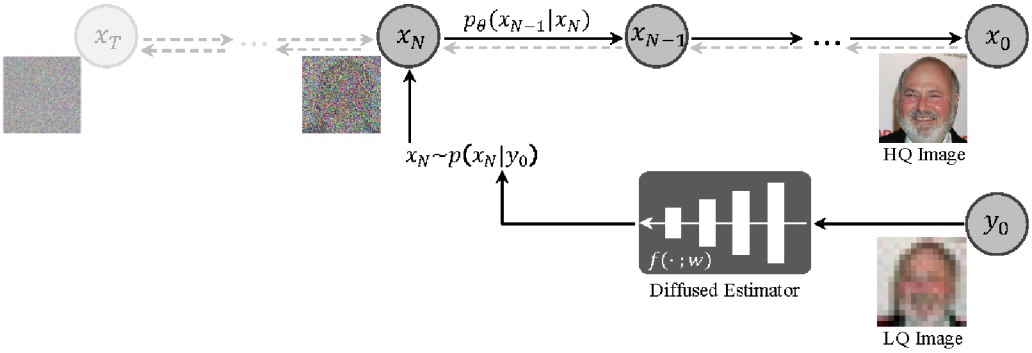

Figure 5.1: Overview of the DifFace method [39].

The inference process involves deriving an intermediate diffused variable x_N (where $N < T$) from the low-quality image y_0 using a specialized neural network, referred to as a diffused estimator. This network is designed to predict the diffusion step x_N based on the input image y_0. Subsequently, the desired x_0 is determined from this intermediate state. This approach offers efficiency gains compared to the full reverse diffusion process from x_T to x_0 by leveraging a pre-trained diffusion model from x_N to x_0. Moreover, there's no need to retrain the diffusion model from the beginning. Notably, this method handles unknown and complex degradations without requiring multiple constraints during training [2].

Figure 5.2: Examples of old photos restoration [39].

The 2 most important parameters that we can change when using DifFace are the number of initial diffusion steps (default set to 100) and the hyper-parameter ETA (default set to 0.5) that controls the fidelity-realness trade-off and can be adjusted between 0.0 and 1.0. The hyper-parameter ETA is used for regulating the randomness of the inference process and offers a lever by which we can control the capability of fidelity preservation of DifFace [39].

It's important to note that DifFace is not specifically designed or optimized for attacking deepfake detection systems. The primary objective is to assess the usage of open-source, available diffusion models based systems. The DifFace code is publicly available on github[1] with a pre-trained version on the FFHQ dataset containing 70,000 HQ photos.

5.2 Arc2Face

Arc2Face is a face foundation model that operates based on identity conditions. It utilizes the ArcFace embedding of an individual to produce a wide range of photo-realistic images. Compared to existing models, Arc2Face achieves a remarkable level of facial similarity in its generated images, all while preserving diversity in the output. Arc2Face can benefit various industrial and research applications, including media, entertainment, and data generation for face analysis and synthesis [30].

[1] https://github.com/zsyOAOA/DifFace

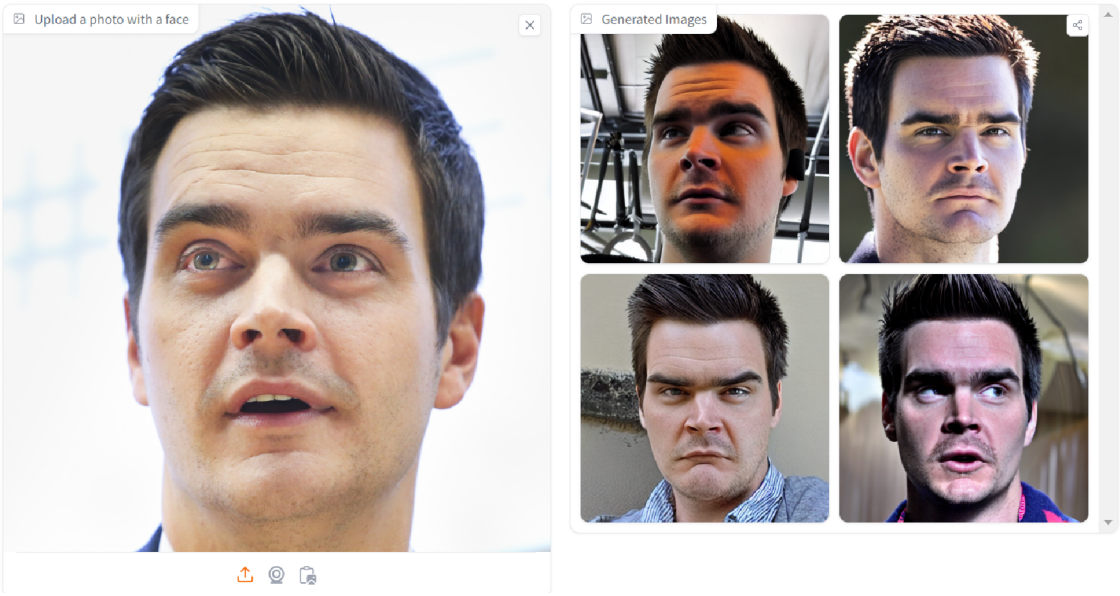

Figure 5.3: Arc2Face male generated photos examples.

The Arc2Face model builds upon a robust Stable Diffusion framework, facilitating the effective generation of top-notch images [33]. The process of generating facial images involves utilizing pre-trained face recognition networks and latent diffusion models. The Latent Diffusion model compresses images for efficient training. It also uses a universal conditioning mechanism to enhance image generation. The model comprises a perceptual autoencoder and a U-net denoiser. The autoencoder encodes facial images into a simpler latent space and then reconstructs them, effectively capturing crucial visual details. Meanwhile, the U-net denoiser eliminates noise from input images, enhancing the quality of the generated images. To maintain key facial features, the model employs cross-attention layers, focusing on relevant aspects of the input images and embeddings during both the encoding and decoding phases. Throughout the training process, standard loss functions are fine-tuned to minimize disparities between the reconstructed and genuine images, progressively improving the model's capability to produce high-quality facial images. The dataset used for training is primarily WebFace42M. Additionally the model is fine-tuned on FFHQ and CelebA-HQ datasets for enhanced quality and less constrained face images. As a result, Arc2Face generates FFHQ-aligned images at 512×512 pixels, providing high-quality facial image synthesis [30].

Arc2Face code is publicly available at their github[2], and they provide easy-to-test API built with Gradio on hugging face website[3]. All you have to do is to upload a photo, and with a few simple settings, you can generate pictures like in Figure 5.3 in just a few seconds.

[2] https://github.com/foivospar/Arc2Face
[3] https://huggingface.co/spaces/FoivosPar/Arc2Face

Chapter 6

Experiments

In our preceding discussions, we underscored the pivotal role of GAN technology in the generation of deepfake images and delved into the rapidly evolving domain of diffusion models. Yet, the unexplored potential lies in the synergy between these technologies, presenting novel opportunities and prospective benefits.

Diffusion models demonstrate exceptional skill in capturing subtle details and relationships within data, which makes them a powerful tool for image denoising or generation. Integrating them with GANs or other existing approaches for making deepfakes could make them look smoother and more natural. This integration poses a challenge for observers, as the subtle distinctions make it increasingly more difficult to detect anomalies. Furthermore, the adeptness of diffusion models in capturing intricate details and correlations within data suggests that their incorporation into the GAN framework could yield deepfakes characterized by enhanced facial expressions, realistic movements, and intricate visual details. This consequently contributes to an overall augmentation of credibility in the synthetic content produced.

This endeavour not only strives to expand the frontiers of deepfake technology but also introduces new challenges for detection mechanisms, thereby advancing the sophistication of synthetic media generation. The primary objective of our work is to rigorously assess whether diffusion models can indeed enhance the quality of generated deepfake images or create entirely new ones. We will evaluate the success rates of deepfake detectors on deepfake images generated or enhanced by diffusion models to assess their effectiveness in detecting this unique form of manipulated media. Furthermore, we also plan to test these deepfakes on biometric face verification systems, which should also be able to detect any alterations made to the facial features of the individual. Our expectation is that diffusion models hold the potential to rectify the classic flaws inherent in deepfake generation, that are often exploited by existing detection mechanisms. By mitigating these imperfections, smoothing out discernible traces, and improving overall image quality, diffusion models have the potential to render deepfakes more convincing and, consequently, impact the outcomes of deepfake detectors.

To evaluate our results, we will mainly focus on absolute changes in the count of individual results of deepfake detectors.

The subsequent section outlines a series of meticulously designed experiments aimed at unravelling the practical implications of this integration and its consequential impact on the deepfake landscape. These experiments will serve as a critical exploration of the potential benefits and challenges associated with diffusion models and based on the results, we should find out whether it makes sense to use them for creating or enhancing deepfakes.

6.1 Experiment 1: Enhancing human GAN generated faces with integration of diffusion models

In this experiment, we select exclusively deepfake images of human faces generated by StyleGAN and subject them to deepfake detectors. Testing on genuine photos will only take place in the second experiment. For this experiment, we will be testing deepfake pictures generated by both StyleGAN2 and StyleGAN3 versions. We used official repositories that provide pre-trained models trained on the Flickr-Faces-HQ (FFHQ) dataset. Subsequently, we enhance these images using a diffusion model tool DifFace and re-evaluate those pictures with the detectors, examining any changes in the detectors success rates based on the number of correctly identified images. *We aim to assess how the incorporation of diffusion models influences the realism and credibility of the generated content and, subsequently, how these enhancements affect the performance of deepfake detectors.* This experiment will be divided into 2 parts. In the first part of the experiment, we will focus on how diffusion models change the results of the most modern types of deepfakes and measure the detector's ability to detect deepfakes generated by StyleGAN3, which they are not trained for. This will be used as something like a blackbox testing of detectors.

On the other hand, both of our tested detectors should be able to detect images generated by StyleGAN2. Therefore, in the second part of the experiment, we will use older types of deepfakes and we will see if the diffusion model can remove all the artifacts according to which the detectors mark them as deepfakes.

6.1.1 StyleGAN3 results

We initiated the dataset acquisition process by employing the pre-trained StyleGANv3 tool to synthesize a wide array of facial images. Thanks to the fact that it is the most modern image generation tool available, we expect very good and realistic-looking image results with various facial expressions, lighting conditions, and backgrounds.

First of all, we test the artificially created images using FaceOnLive and GAN detector to set a starting point for how well the detectors can identify them. For our testing we pick up only faces that both detectors returned a valid result. Common problems were that on some pictures, there was no face detected, the face was too small, multiple faces were detected, or the face was not fronted, and therefore, the detector could not return a valid result. Those pictures were returned with an error message instead of an evaluation result. Because of this, we had to keep only 712 of the original 867 StyleGAN3-tested images that we planned to test.

Table 6.1: FaceOnLive initial results

Face Status	**Count**	**Average Liveness Score**
Genuine	393	3.845
Deepfake	242	-600
Spoof	77	-1.738

Table 6.2: GAN detector initial results

Face Status	**Count**	**Average logit**
Genuine	659	-4.652
Deepfake	53	1.098

In Table 6.1 and Table 6.2, we can see the results of 712 tested StyleGAN3-generated images. We can see that if we calculate the ratio of correctly rated images, we get about 45% for the FaceOnLive detector and only 7% for the GAN detector. StyleGAN3 deepfakes results are already impressive, with pretty high average liveness scores for genuine evaluated pictures and detectors that are not strictly trained for it have big problems finding deepfakes artifacts. GAN detector with the pre-trained version on StyleGAN2 recorded very little success. This proves how much better the newer versions of deepfakes are and how difficult it is to detect deepfakes, which detectors are not trained for.

Nevertheless, the FaceOnLive deepfake detector was able to detect some very high-quality generated deepfakes. In Figure 6.1, we see examples of very realistically created pictures on which it would be very difficult to find any anomalies or signs of deepfake technology for the human eye. However, all of them were correctly marked as fake by the FaceOnLive detector.

Figure 6.1: Very realistic pictures detected as deepfakes.

First of all, for the correct execution of our experiment, we tried to find ideal DifFace parameters that would have the highest chance of improving our test results. We picked 100 randomly selected pictures from our dataset and tried to find out how the change of initial diffusion steps and ETA parameter would affect the test results for the FaceOnLive deepfake detector. Because of already very low success rate of the GAN detector, we decided to find ideal parameters only based on the FaceOnlive detector results. Later when we find out the best parameters, we fully execute our experiment for both face liveness detectors on all 712 pictures.

Figure 6.2: Different diffusion steps.

In Figure 6.2, we can see how increasing the number of diffusion steps slowly changes the individual details of the image, such as wrinkles, hair, or scars on the face. At 120 initial diffusion steps there are visible changes that differentiate from the original picture and DifFace starts to fail when restoring our photos, that's why we didn't test higher values. The change of ETA parameter does not cause any significantly visible changes for the human eye.

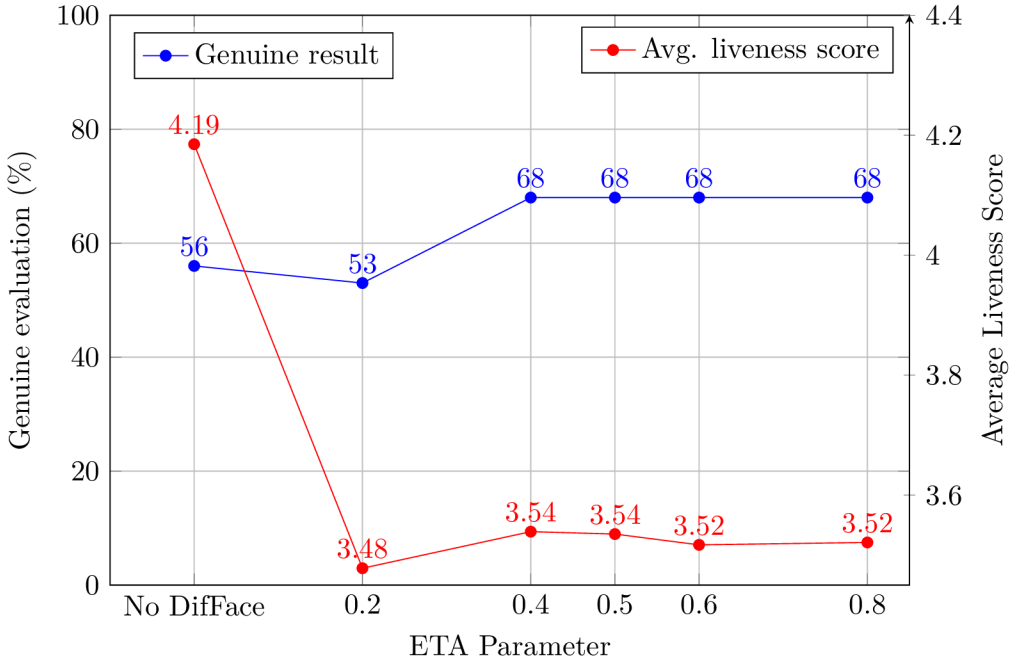

Figure 6.3: Percentage of tested images with genuine result and average Liveness Score for different ETA parameter

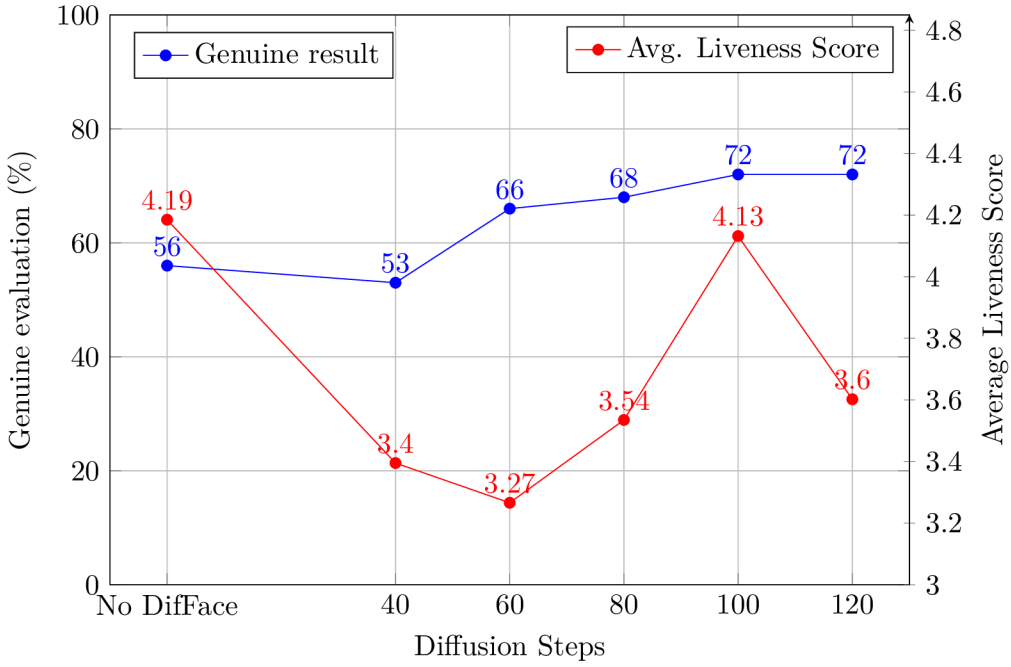

Figure 6.4: Percentage of tested images with genuine result and average liveness score for different diffusion steps.

In Figure 6.3 and 6.4, we can see the statistics results of the detector for different ETA and diffusion steps values. The best overall results were obtained for 100 diffusion

steps, where the number of genuine pictures changed from 56 to total of 72. The average liveness score for genuine pictures is slightly lower, but that could be caused because 16 pictures that were previously marked as fake are also included in the calculation. The ETA parameter did not make any change to the picture for the human eye and did not cause significant changes in the overall results, but it did slightly affect the average liveness score. Thanks to these results, we were able to determine the ideal parameters of diffusion steps at 100 and ETA at 0.5. These results already indicate that the improvement of the diffusion model really increased the reliability of the tested deepfakes, but for better results we need to perform the experiment on more samples and both detectors. After these findings, we could fully execute our experiment for both tested detectors.

Table 6.3: FaceOnLive results for 100 diffusion steps and 0.5 ETA

Face Status	**Count**	**Average Liveness Score**
Genuine	512	3.705
Deepfake	93	-600
Spoof	87	-1.746
Face could not be tested	20	x

Table 6.4: GAN detector results for 100 diffusion steps and 0.5 ETA

Face Status	**Count**	**Average logit**
Genuine	712	-5.421
Deepfake	0	x

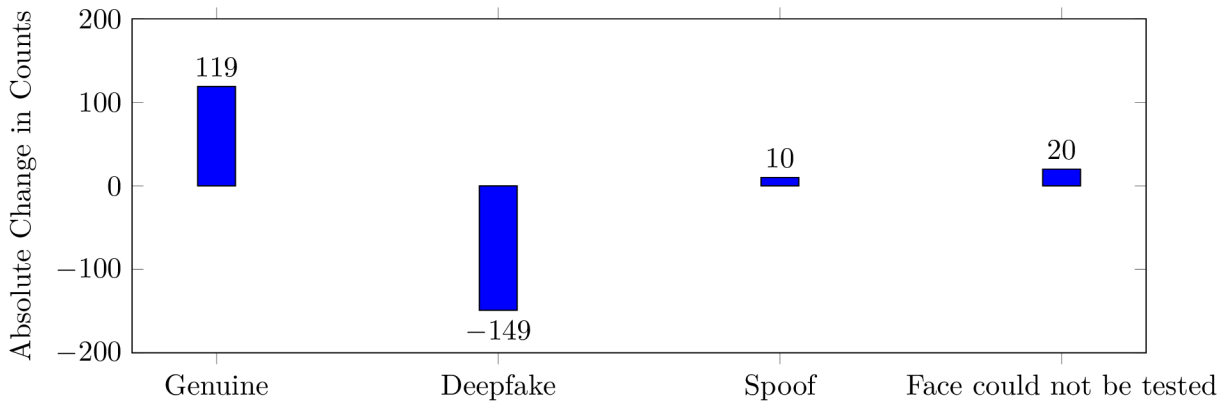

Figure 6.5: Absolute changes of 712 pictures after DifFace for FaceOnLive detector.

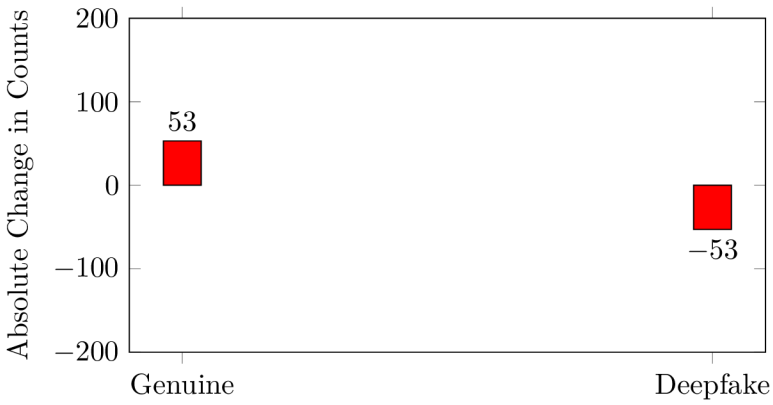

Figure 6.6: Absolute changes of 712 pictures after DifFace for GAN detector.

In Table 6.3 and Table 6.4, we can see the final results of both detectors. Figures 6.5 and 6.6 show how the absolute count has changed for individual results categories. The GAN detector has completely lost its ability to recognize these deepfakes. The success rate with StyleGAN3 photos was already low, but not a single photo was detected as a deepfake after the enhancement. Artifacts left by StyleGAN3 generation were removed, and even if some other types of artifacts could appear by using DifFace, the detector was not trained for them and did not consider them as evidence of deepfake. In addition, the average value of the logit for the images decreased, which means that the images are more trustworthy and the detector considers them to be real.

Also, when testing on the FaceOnLive detector, the quality of deepfakes increased. The most important part is that after enhancements, the count of images classified as strictly deepfakes significantly decreased to 93. This indicates a substantial improvement in the perceived authenticity of the images, as fewer were detected as synthetic by the deepfake detector. Initially, 393 images were classified as genuine, with an average liveness score of 3.845. Post-enhancement, the number of images classified as genuine increased to 512, even though with a slightly lower average liveness score of 3.705. That is a pretty impressive improvement of more than **30%** images. For some reason, 20 images that have been previously in one of 3 unambiguous results (genuine, spoof, deepfake) could not be tested. This indicates a potential area where diffusion models might introduce artifacts or changes that are not typical of genuine images or failure of the diffusion model. In Table 6.5, we have also processed the specific occurrences of the changes in the evaluation of this detector.

Table 6.5: FaceOnLive changes occurances

Change	Occurrences
genuine remained the same	326
genuine changed to spoof	32
genuine changed to deep fake	27
genuine changed to face could not be tested!	8
deep fake remained the same	58
deep fake changed to genuine	157
deep fake changed to spoof	22
deep fake changed to face could not be tested!	5
spoof remained the same	33
spoof changed to deep fake	8
spoof changed to genuine	29
spoof changed to face could not be tested!	7

6.1.2 StyleGAN2 results

In the second part of our first experiment, we will use an older version of StyleGAN2, for which our detectors should be better prepared. We collected 500 images generated by StyleGAN2 which were initially tested by our detectors with valid testing results. The results of this testing are shown in Table 6.6 and Table 6.7.

Table 6.6: FaceOnLive initial testing results

Face Status	Count	Average Liveness Score
Genuine	135	3.603
Deepfake	313	-600
Spoof	52	-2.967

Table 6.7: GAN detector initial testing results

Face Status	Count	Average logit
Genuine	0	X
Deepfake	500	14.257

We can see that especially the GAN detector recorded a significant improvement in detecting deepfakes compared to deepfakes generated by StyleGAN3. The pre-trained version for StyleGAN2 seems to be of really good quality and in our case, it recorded 100% success and correctly marked all images as deepfakes. Also, the FaceOnLive detector recorded an initial improvement and correctly identified almost 73% of deepfakes or modified images.

After that we enhanced all pictures with DifFace using 100 diffusion steps and 0.5 ETA value, just like for StyleGAN3 generated deepfakes.

Table 6.8: FaceOnLive results for 100 diffusion steps and 0.5 ETA

Face Status	Count	Average Liveness Score
Genuine	359	4.064
Deepfake	45	-600
Spoof	85	-1.846
Face could not be tested	11	x

Table 6.9: GAN detector results for 100 diffusion steps and 0.5 ETA

Face Status	Count	Average logit
Genuine	500	-4.554
Deepfake	0	X

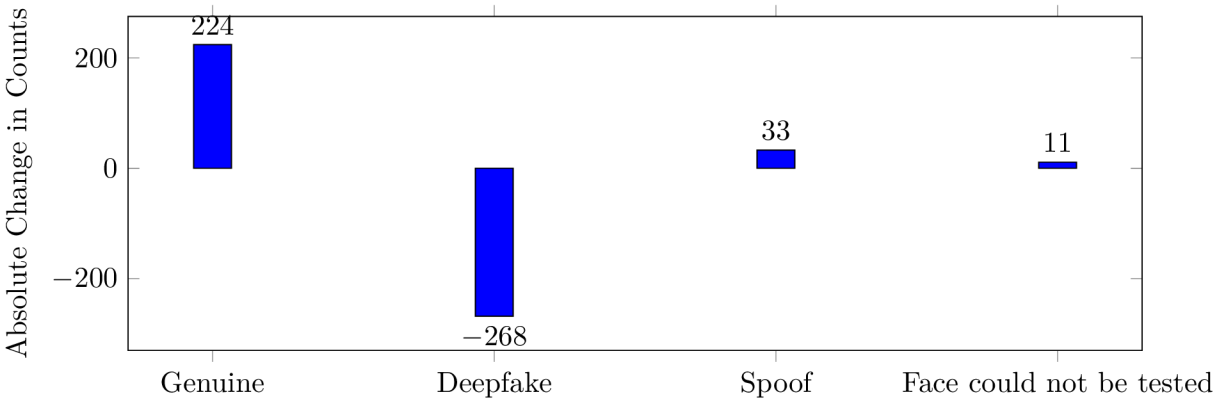

Figure 6.7: Absolute changes of results FaceOnLive detector.

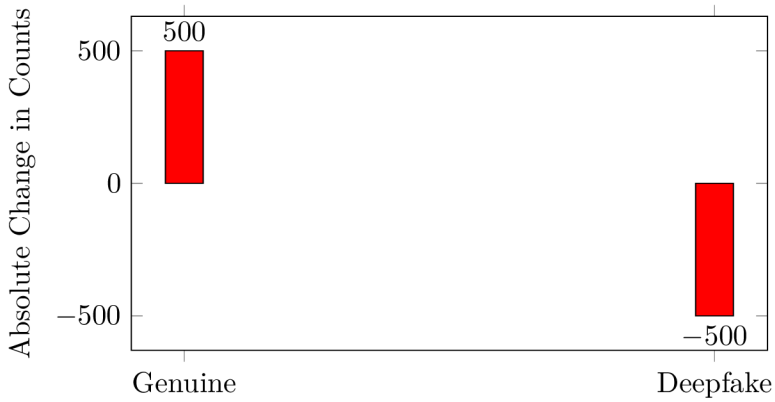

Figure 6.8: Absolute changes of results GAN detector.

In Table 6.8 and Table 6.9, we can see the final results of both detectors. In Figures 6.7 and 6.8 we see how the absolute count has changed for individual categories. The final results were really surprising. Although the GAN detector initially correctly evaluated all the images as deepfakes, there was not a single one after using the diffusion model

enhancements. The diffusion model really managed to remove all the artifacts that this detector considered as signs of deepfake, even in extreme cases when StyleGAN2 failed and generated some kind of glitched image. This type of images 6.9, would be easily recognizable even by a human eye. However, because of subtle changes made by DifFace, the detector could not detect them correctly.

Figure 6.9: Examples of easily detectable deepfakes.

DifFace also achieved success with the FaceOnLive detector. 72% of pictures were evaluated as genuine and the average liveness score rose to 4.064. Although this detector was able to detect deepfakes as in Figure 6.9, it also had problems with more credible-looking deepfakes.

Summarizing the results of this experiment, we found out that by enhancing deepfake images using diffusion models, we managed to improve their credibility for deepfake detectors and thus significantly increase their chances of being considered genuine.

6.2 Experiment 2: Testing deepfake detectors on real pictures enhanced by diffusion model

In this experiment, we will test the use of diffusion models on exclusively genuine images and find out whether deepfake detectors will not start to consider them as deepfakes. While diffusion models excel in enhancing visual quality, they might inadvertently introduce subtle changes that could be detected by the deepfake detection system. The challenge lies in striking a balance between visual enhancement and maintaining the authenticity cues crucial for accurate classification.

This experiment involves a comparative analysis between two sets of images: original, unaltered real images and their counterparts that have been enhanced or modified using diffusion models. Both sets will be subjected to deepfake detection to assess the impact of diffusion model processing on the detection outcomes. We will observe whether the result of the diffusion model does not leave noise on the photos, which could confuse deepfake detectors with noise generated by GAN tools and thus mistakenly mark the images as deepfakes. However, with real photos, we often want the photo to remain as close to the original as possible and thus with unchanged facial features. Because of that, we test them

on Megamatcher and FaceOnLive biometric systems and find out if they can still accurately match an enhanced image to its original, unaltered counterpart, thereby determining the impact of such image modifications on biometric verification processes.

The usage of the FFHQ dataset provides a wide range of subjects, lighting conditions, and backgrounds to ensure comprehensive testing. All images are standardized to a consistent 1024x1024 resolution and format, ensuring that any variations in detection outcomes are attributable to the application of the diffusion model rather than unexpected external variables. We randomly selected 500 pictures from this dataset and subjected them to initial testing to make sure all of them would be tested as genuine. We deleted all misaligned or badly-fronted pictures. Due to the capacity of one of our tested detectors, we will test all 500 images only for 100 diffusion steps, for other number of diffusion steps we will test only the first 100 images.

Table 6.10: FaceOnLive detector initial testing results

Face Status	**Count**	**Average Liveness Score**
Genuine	500	3.498

Table 6.11: GAN detector initial testing results

Face Status	**Count**	**Average logit**
Genuine	500	-16.793

It is interesting that the average liveness score of 500 random photos (3.498) from the FFHQ dataset is lower than in the initial testing of photos generated by StyleGAN v3 (3.845) in the previous experiment. This indicates how developed and trustworthy deepfake generators are and how difficult it is to recognize real photos. Artificially generated images exhibit face artifacts that appear even more convincing than those found in real photos.

After that, we tried to change initial diffusion steps and observe how the deepfake detection results will change. Ideally, diffusion models should refine and improve the visual appearance of genuine images without introducing characteristics that mimic the manipulations typically associated with deepfakes.

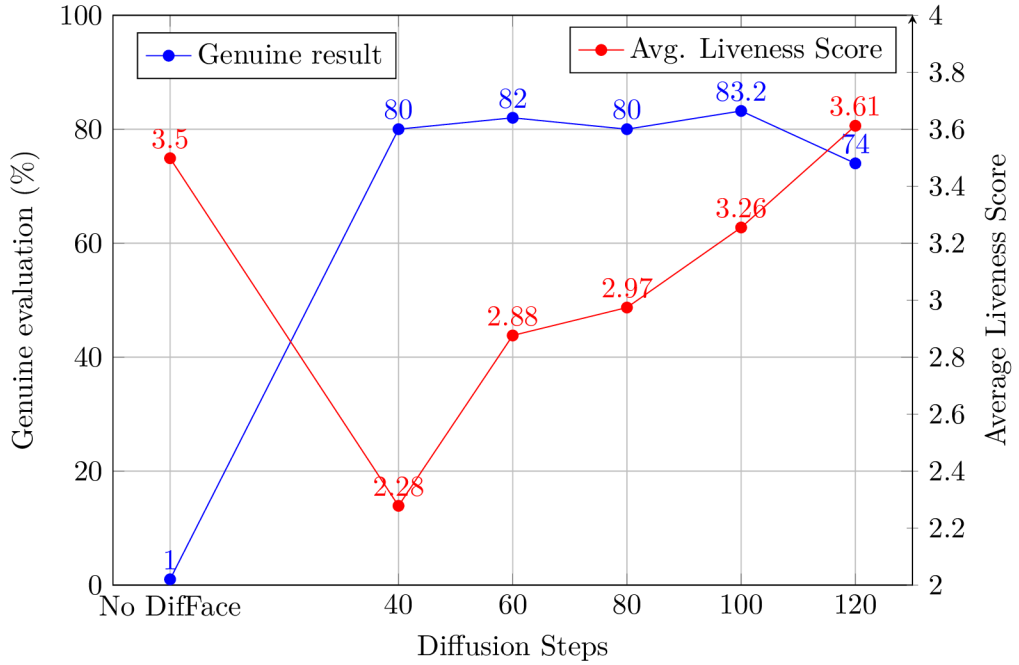

Figure 6.10: Percentage of tested images with genuine result and average liveness score by FaceOnLive detector.

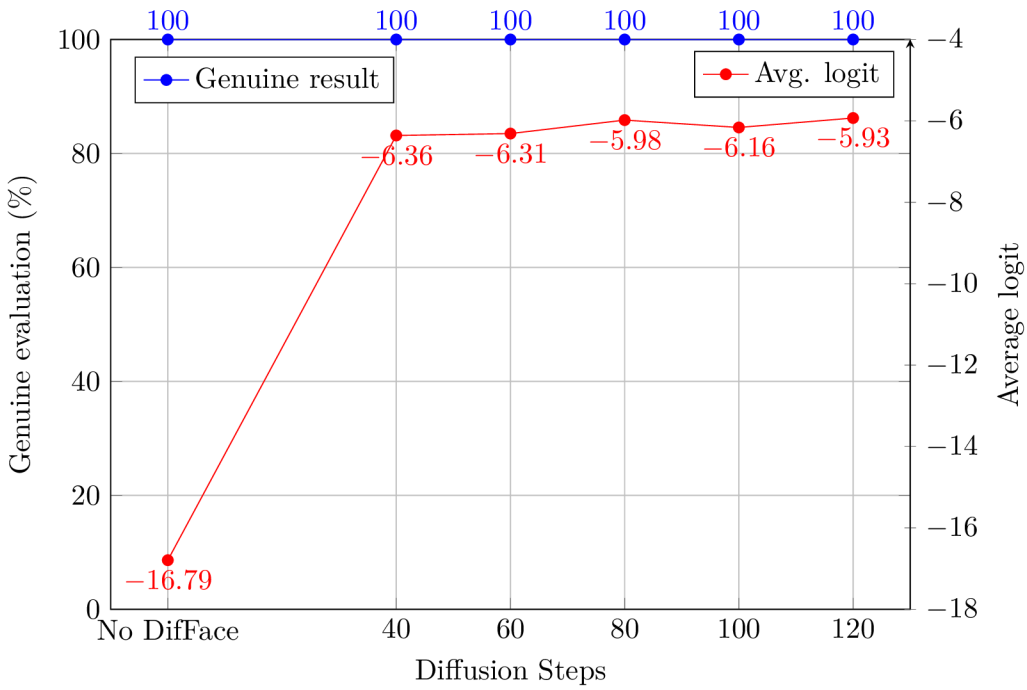

Figure 6.11: Percentage of tested images with genuine result and average logit by GAN detector.

In Figure 6.11 we can see the results GAN image detector testing for different diffusion steps. As we found out also in the previous experiment, this detector is pre-trained on

StyleGAN2 pictures, and it seems that it is not able to detect any other kind of deepfakes, so it was expected that even in this experiment, it had no problem considering all images, as genuine, regardless of the number of diffusion steps.

In Figure 6.10 we can see the results of FaceOnLive detector testing and how the average liveness score changed. Images are less visibly changed in fewer diffusion steps but tend to have lower quality for human observers. The average liveness score of tested pictures significantly decreased, this can be due to the quality of the image or the poor processing of certain parts of the face by the diffusion model (such as when rendering the eye), errors are introduced into the image. These errors sometimes occur, as shown in Figure 6.12. We are not sure why these errors appear, but most of the time, it is enough just to restart the DifFace with the original picture, and the result is without error.

With more diffusion steps, there is a possibility of over-enhancement. The diffusion process might unintentionally amplify certain patterns that trigger the deepfake detection system and evaluate face liveness as insufficient. Overall, we found that the detectors mostly do not consider the images modified by the diffusion model to be deepfakes. If such a situation did occur, it was mostly due to an error in the diffusion model as in Figure 6.12, which was detected by the FaceOnLive detector. The GAN detector, which is focused on detecting only GAN-generated deepfakes, was not able to detect even such failures and considered all photos to be genuine.

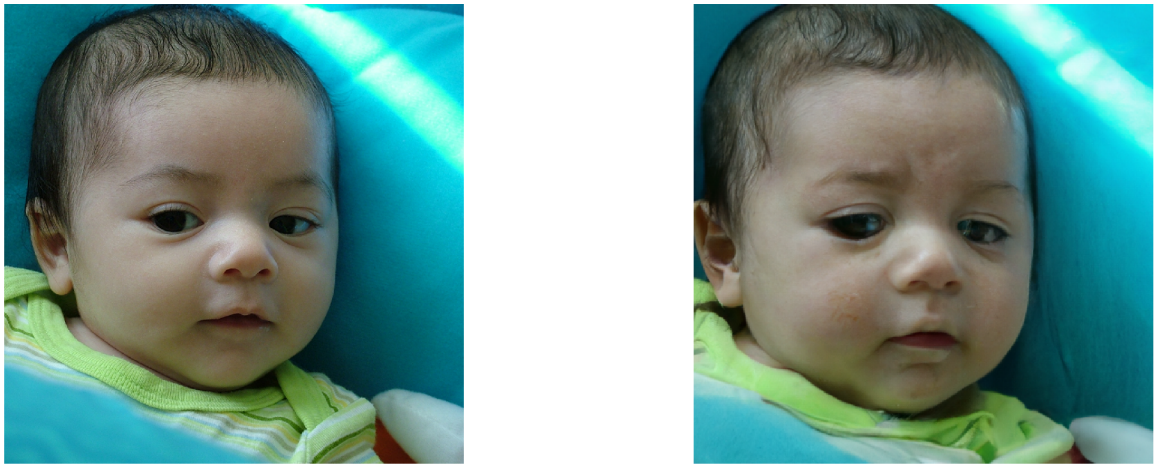

Figure 6.12: Incorrectly processed eyes by diffusion model. Original on the left and picture enhanced by DifFace with 100 diffusion steps on the right.

After that, we uploaded those upgraded pictures into our face-matching systems to find out how sensitive they are to changes made by diffusion models when identifying a person and whether, by any chance, they will not consider the enhanced image of a person to be false.

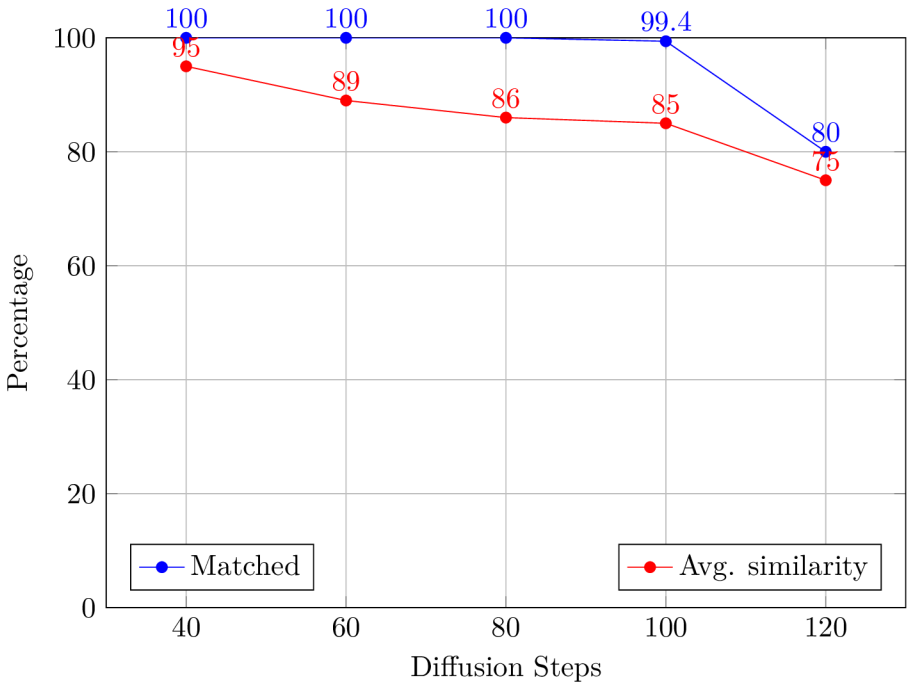

Figure 6.13: Face recognition FaceOnLive results.

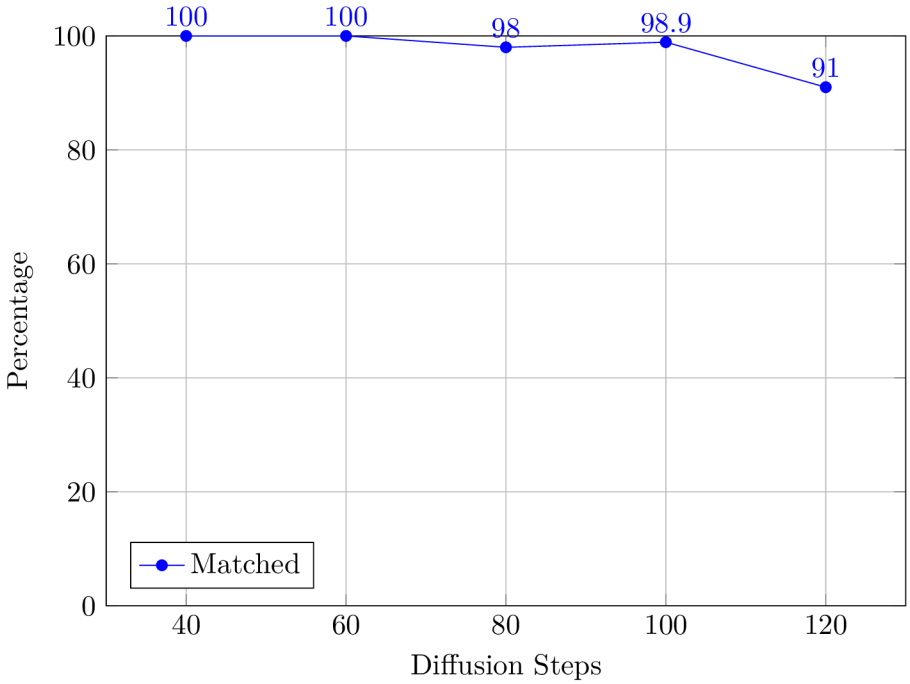

Figure 6.14: Megamatcher identification results.

In Figure 6.13 and Figure 6.14, we can see the results of our biometric identifiers. To summarise the results, we can see that up to 100 diffusion steps images are not really changed for the human eye or biometric system. In almost all cases, the pictures are

evaluated as the same person and no synthetic modifications of the face are detected to such a level that it could be considered a different person's picture or not match its original. However, at 120 starting diffusion steps, we observe changes in the evaluation as both face recognition systems reject some pictures. 100 initial diffusion steps still seem to be the optimal value for using our model. At this value, the improvement of the image is most noticeable, and even if fine details are changed in the image, it is still within the assessment limits of the same human face. Exceeding this value, however, leaves much larger changes in the image. At this value, the positive assessment by the biometric systems decreases, and changes are also visible to the human eye (for example, the colour and shape of eyes on Figure 6.15).

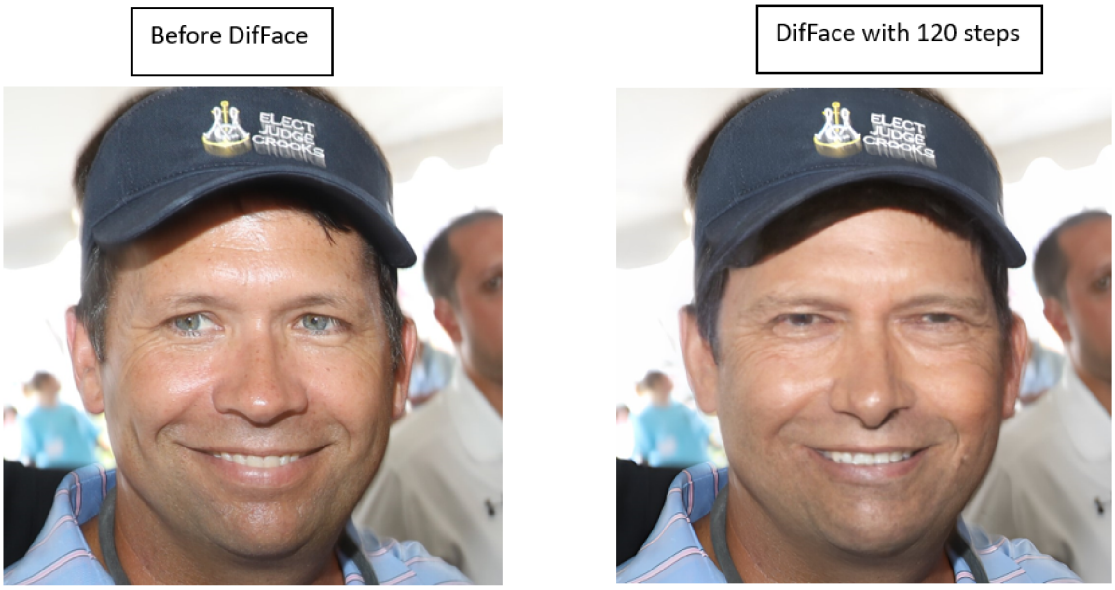

Figure 6.15: Visible changes in pictures at 120 initial diffusion steps.

From this experiment, we learned that using the DifFace tool on authentic images only slightly changed the test results, and the vast majority of images passed the deepfake detection testing as genuine. In addition, we found out that using this tool to improve the picture quality does not change the individual features of the human face to such an extent that the picture does not pass the biometric comparison with the original picture.

6.3 Experiment 3: Generating faces with diffusion model

This experiment aims to assess the performance of deepfake detection systems on images generated by the Arc2Face diffusion model. The process involves two main stages: image generation and deepfake detection evaluation. Later, we will also test the generated images with biometric systems, which will compare the similarities of the inserted images with the generated ones.

Firstly, 500 facial images will be generated using the Arc2Face model. To achieve this, we selected various photos of people from the FFHQ dataset and tried to generate 4 samples for each. Arc2Face produced high-quality 512x512 images with features based on input identity data. Then, we inserted these resulting images into the FaceOnLive and

GAN image deepfake detectors. The results of our testing can be seen in Table 6.12 and Table 6.13.

Table 6.12: FaceOnLive results for 500 Arc2Face deepfakes

Face Status	Count	Average Liveness Score
Spoof	230	-3.306
Genuine	202	2.508
Deepfake	11	-600
Face could not be tested	57	X

Table 6.13: GAN detector results for 500 Arc2Face deepfakes

Face Status	Count	Average logit
Genuine	500	-9.816
Deepfake	0	X

We can see that the success of genuine photos exceeded 40% with the FaceOnLive detector and 100% with the GAN detector. Although the average liveness score is not as high as with the photos enhanced by DifFace in experiments 1 and 2, this success rate is still very high. Only 11 of these photos are directly classified as deepfakes by the FaceOnLive detector. It evaluated most of the images as spoofs, probably considering them authentic pictures edited with Photoshop or another similar tool. 57 photos could not be detected, mostly due to bad face alignment or small face.

Next, we performed testing with biometric systems, the results of which can be seen in Table 6.14 and Table 6.15.

Table 6.14: Results of FestOnLive face recognition system

Diffusion Steps	Matched	Total Average Similarity (%)
same	477	**83.68**
different	23	

Table 6.15: Results of Megamatcher matched images

Result	Matched
Same	465
Different	35

The biometric system recognized the majority of the deepfake images as being the same as the original reference images. This suggests that the biometric systems struggled to effectively distinguish between genuine and deepfake images, resulting in a significant number of false positive identifications. These results underscore the remarkable ability of Arc2Face to produce deepfake images that closely mimic the characteristics of the original reference images and moreover, they highlight the challenges caused by diffusion-based deepfake generation methods in terms of detection and authentication by biometric systems.

Overall, in this experiment, we achieved the rapid generation of high-quality deepfake images of human faces, many of which were evaluated as genuine by face liveness deepfake detectors and, moreover, passed biometric identification tests.

Chapter 7

Conclusion

The proliferation of deepfake technology poses significant challenges to society, particularly in misinformation, privacy invasion, and the erosion of trust in digital media. Through the lens of this thesis, we've explored how diffusion models can be leveraged to create more convincing and sophisticated synthetic media, raising both the promise of innovation and the spectre of heightened societal risks.

It is indisputable that this technology has great potential use in areas such as entertainment, visual effects, and virtual reality. By refining the synthesis process, researchers and content creators can produce highly realistic simulations with unprecedented levels of fidelity, opening new avenues for creative expression and immersive storytelling. However, this very same technology also amplifies the dangers posed by deepfake manipulation. The ability to enhance or generate synthetic content indistinguishable from reality exacerbates existing concerns about deepfakes. As both deepfake generating software and diffusion models architectures become more accessible to the general public, the barrier to entry for producing sophisticated deepfakes diminishes, amplifying the potential for widespread abuse and manipulation.

In this thesis, we tested 2 different detectors that recognize face liveness in pictures and detect synthetic media and tested whether they would still be able to detect this content just as well, even after using diffusion models for enhancing the original image or generating new one. As a diffusion model tool for enhancing deepfakes we chose and used DifFace, which is intended for the face restoration of images. Our endeavours revealed the inherent difficulty in finding a one-size-fits-all detector for deepfakes. We tested the GAN detector with ResNet50 architecture that was specifically focused and pre-trained on pictures generated by the StyleGAN2 generator. Even though it was very capable of detecting these pictures, only a subtle adjustment of the diffusion model was enough for the detector to lose its ability to recognize these images and the same images that were previously considered to be deepfakes were evaluated to be genuine. Another detector was the FaceOnLive face liveness detector, which appeared to be more universal and pre-trained for various types of deepfakes. This detector was not so much focused on specific artifacts generated by one tool and thus had a slightly better chance of detecting deepfakes improved by the diffusion model. However, the success rate remained low, reaffirming our realization that diffusion models will challenge existing detection algorithms.

Traditional detection methods, reliant on static features or artifacts inherent in deepfake generation processes, struggled to discern subtle alterations introduced by diffusion-based enhancement. As a result, the efficacy of detectors was compromised, leading to increased false negative rates and diminished confidence in detection outcomes. The DifFace archi-

tecture was not specially modified to improve deepfakes, and it can be expected that if it was, the images would be even more believable and more difficult to recognize from the real ones.

In our next experiment, we tested the diffusion model on real photos. The main use of our test model, DifFace, was the restoration of mostly old or low-quality images. That's why we wanted to find out whether such images will still be considered genuine by deepfake detectors and will not be mistakenly evaluated as deepfakes. We also used 2 biometric systems with which we tested the similarity of the images before and after enhancement with our model. In most of the tested cases, these enhancements did not trigger our face liveness detection architectures, and for the classic use of DifFace with default values, the images were evaluated as the same by both biometric systems. Even though unexpected DifFace errors appeared when restoring the image in rare cases, we can evaluate that it is a high-quality architecture with very good results for face restoration.

In our third experiment, we focused on photos generated by the diffusion model technology from scratch, instead of enhancing already existing images. We used the Arc2Face model, which can generate high-quality deepfakes of a given subject using only his face embedding. We tried to test these generated pictures with our deepfake detectors and then with biometric systems, which were supposed to confirm their similarity with the original photo. The resulting images were very similar to the original images and passed biometric testing with high success. Face liveness was also high and none of our used detectors could consistently detect these deepfakes. The results were both remarkable and concerning, as the diffusion model demonstrated an unprecedented ability to rapidly and effortlessly generate highly realistic deepfake images of a given person with exceptional quality.

One possible approach could be to train a deepfake detector capable of detecting the usage of diffusion models technology on images. For example, a discriminator model trained specifically on artifacts generated through a fixed number of denoising iterations. This discriminator could potentially effectively identify manipulations associated with the level of diffusion noise corresponding to the specific number of diffusion steps used. While such a detector could excel in detecting diffusion model enhancements, it might struggle to generalize its detection capabilities to other types of deepfake attacks. To address this limitation, a more comprehensive deepfake detection system needs to be developed that would incorporate diverse training data and employ techniques that enhance the model's ability to detect various types of manipulations and artifacts commonly found in different deepfake generation methods.

Bibliography

[1] Agrawal, R. *Generative Adversarial Networks (GANs): End-to-End Introduction.* 2024. Available at: https://www.analyticsvidhya.com/blog/2021/10/an-end-to-end-introduction-to-generative-adversarial-networksgans/.

[2] Amanatullah, A. *Meet DifFace: A Novel Deep-Learning Diffused Model For Blind Face Restoration.* July 2023. Available at: https://medium.com/@amanatulla1606/meet-difface-a-novel-deep-learning-diffused-model-for-blind-face-restoration-ab4bda7143f4.

[3] Bai, H., Kang, D., Zhang, H., Pan, J. and Bao, L. *FFHQ-UV: Normalized Facial UV-Texture Dataset for 3D Face Reconstruction.* 2023. Available at: https://arxiv.org/abs/2211.13874.

[4] BasuMallick, C. *Deepfake Types, Examples, Prevention* [Spiceworks Inc.]. May 23 2022. Available at: https://www.spiceworks.com/it-security/cyber-risk-management/articles/what-is-deepfake/.

[5] Chamanth, M. No more confusion on Backpropagation. *AI Mind.* Nov 2022. Published in AI Mind, 11 min read. Available at: https://pub.aimind.so/no-more-confusion-on-backpropagation-7adfc271539f.

[6] Curt Devine, S. L. *A fake recording of a candidate saying he'd rigged the election went viral. Experts say it's only the beginning | CNN Politics — edition.cnn.com.* 2024. Available at: https://edition.cnn.com/2024/02/01/politics/election-deepfake-threats-invs/index.html.

[7] Dabbura, I. *Coding Neural Network — Forward Propagation and Backpropagation* [Towards Data Science]. April 2018. Available at: https://towardsdatascience.com/coding-neural-network-forward-propagation-and-backpropagtion-ccf8cf369f76.

[8] FaceOnLive. *Biometrics Authentication & ID Verification - FaceOnLive.* 2023. Available at: https://faceonlive.com/.

[9] Fortinet. *What is deepfake: AI endangering your cybersecurity?* Available at: https://www.fortinet.com/resources/cyberglossary/deepfake.

[10] Goodfellow, I. J., Pouget Abadie, J., Mirza, M., Xu, B., Warde Farley, D. et al. *Generative Adversarial Networks.* 2014. Available at: https://arxiv.org/abs/1406.2661.

[11] Gragnaniello, D., Cozzolino, D., Marra, F., Poggi, G. and Verdoliva, L. *Are GAN generated images easy to detect? A critical analysis of the state-of-the-art.* 2021. Available at: https://arxiv.org/abs/2104.02617.

[12] He, K., Zhang, X., Ren, S. and Sun, J. *Deep Residual Learning for Image Recognition*. 2015. Available at: https://arxiv.org/abs/1512.03385.

[13] Heather Chen, K. M. *Finance worker pays out $25 million after video call with deepfake 'chief financial officer' | CNN — edition.cnn.com*. 2024. Available at: https://edition.cnn.com/2024/02/04/asia/deepfake-cfo-scam-hong-kong-intl-hnk/index.html.

[14] Jiang, S., Qin, S., Pulsipher, J. L. and Zavala, V. M. *Convolutional Neural Networks: Basic Concepts and Applications in Manufacturing*. 2022. Available at: https://arxiv.org/abs/2210.07848.

[15] Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J. et al. *Alias-Free Generative Adversarial Networks*. 2021. Available at: https://arxiv.org/abs/2106.12423.

[16] Karras, T., Laine, S. and Aila, T. *A Style-Based Generator Architecture for Generative Adversarial Networks*. 2019. Available at: https://arxiv.org/abs/1812.04948.

[17] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J. et al. *Analyzing and Improving the Image Quality of StyleGAN*. 2020. Available at: https://arxiv.org/abs/1912.04958.

[18] Kim, M., Liu, F., Jain, A. and Liu, X. *DCFace: Synthetic Face Generation with Dual Condition Diffusion Model*. 2023. Available at: https://arxiv.org/abs/2304.07060.

[19] Klaudel, G. *StyleGAN 3 - Computer Vision and Image Manipulation*. Available at: https://www.appsilon.com/post/stylegan-3-image-manipulation.

[20] Ko, J., Kong, I., Park, D. and Kim, H. J. *Stochastic Conditional Diffusion Models for Semantic Image Synthesis*. 2024. Available at: https://arxiv.org/abs/2402.16506.

[21] Koo, H. and Kim, T. E. *A Comprehensive Survey on Generative Diffusion Models for Structured Data*. 2023. Available at: https://arxiv.org/abs/2306.04139.

[22] Mehlig, B. *Machine Learning with Neural Networks: An Introduction for Scientists and Engineers*. Cambridge University Press, october 2021. ISBN 9781108494939. Available at: http://dx.doi.org/10.1017/9781108860604.

[23] Mirsky, Y. and Lee, W. The Creation and Detection of Deepfakes: A Survey. *ACM Computing Surveys*. Association for Computing Machinery (ACM). january 2021, vol. 54, no. 1, p. 1–41. DOI: 10.1145/3425780. ISSN 1557-7341. Available at: http://dx.doi.org/10.1145/3425780.

[24] Neurotechnology. *Megamatcher SDK*. January 2023. Available at: https://www.neurotechnology.com/megamatcher.html.

[25] Neurotechnology. *Megamatcher SDK*. 2023. Available at: https://neurotechnology.co/megamatcher-especificaciones-tecnicas/.

[26] Nguyen, T. T., Nguyen, Q. V. H., Nguyen, D. T., Nguyen, D. T., Huynh The, T. et al. Deep learning for deepfakes creation and detection: A survey. *Computer Vision and Image Understanding*. Elsevier BV. october 2022, vol. 223, p. 103525. DOI: 10.1016/j.cviu.2022.103525. ISSN 1077-3142. Available at: http://dx.doi.org/10.1016/j.cviu.2022.103525.

[27] O'Connor, R. *Introduction to Diffusion Models for Machine Learning*. 2022. Available at: https://www.assemblyai.com/blog/diffusion-models-for-machine-learning-introduction/.

[28] O'Shea, K. and Nash, R. *An Introduction to Convolutional Neural Networks*. 2015. Available at: https://arxiv.org/abs/1511.08458.

[29] O'Sullivan, D. *Now fake Facebook accounts are using fake faces | CNN Business — edition.cnn.com*. 2024. Available at: https://edition.cnn.com/2019/12/20/tech/facebook-fake-faces/index.html.

[30] Papantoniou, F. P., Lattas, A., Moschoglou, S., Deng, J., Kainz, B. et al. *Arc2Face: A Foundation Model of Human Faces*. 2024. Available at: https://arxiv.org/abs/2403.11641.

[31] Prasad, D. V., M., H., Krishna, N. N., Sanjay, T. C. and C., D. K. Y. *Comparative study on Deepfake Detection Methods*. 2022. Available at: https://ijarcce.com/papers/comparative-study-on-deepfake-detection-methods/.

[32] Prenger, R., Valle, R. and Catanzaro, B. *WaveGlow: A Flow-based Generative Network for Speech Synthesis*. 2018. Available at: https://arxiv.org/abs/1811.00002.

[33] Rombach, R., Blattmann, A., Lorenz, D., Esser, P. and Ommer, B. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022. Available at: https://arxiv.org/abs/2112.10752.

[34] Ronneberger, O., Fischer, P. and Brox, T. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. 2015. Available at: https://arxiv.org/abs/1505.04597.

[35] Stan, G. B. M., Wofk, D., Aflalo, E., Tseng, S.-Y., Cai, Z. et al. *LDM3D-VR: Latent Diffusion Model for 3D VR*. 2023. Available at: https://arxiv.org/abs/2311.03226.

[36] Thing, V. L. L. *Deepfake Detection with Deep Learning: Convolutional Neural Networks versus Transformers*. 2023. Available at: https://arxiv.org/abs/2304.03698.

[37] Wu, T., Fan, Z., Liu, X., Gong, Y., Shen, Y. et al. *AR-Diffusion: Auto-Regressive Diffusion Model for Text Generation*. 2023. Available at: https://arxiv.org/abs/2305.09515.

[38] Yu, D., Wang, H., Chen, P. and Wei, Z. Mixed Pooling for Convolutional Neural Networks. In:. October 2014, p. 364–375. Available at: https://www.researchgate.net/publication/300020038_Mixed_Pooling_for_Convolutional_Neural_Networks.

[39] Yue, Z. and Loy, C. C. *DifFace: Blind Face Restoration with Diffused Error Contraction*. 2023. Available at: https://arxiv.org/abs/2212.06512.

Appendix A

Attached media

The Src folder contains subfolders of python scripts used for working with individual tools in this thesis:

- `Arc2Face`: A script for generating pictures with API
- `FaceOnLive_detector`: Script for detecting deepfakes with API and to process outputs
- `Face_recognition`: Scripts for comparing two pictures with API and to process outputs
- `GAN_image_detector`: A script to process outputs
- `Megamatcher`: A script to process outputs