الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

جامعة سعيدة د. مولاي الطاهر

كلية التكنولوجيا

قسم: الإعلام الآلي

Master's thesis

Specialty : computer security and cryptography

Theme

DeepFake for DeepPrivacy

Presented by :

ALI CHERIF Nora

TERCHI Naima

Led by:

Mr. Hadj Ahmed Bouarara

Class of 2021 - 2022

Dedicaces

To my dear parents,

I can't with these few lines to express my love and gratitude to you. However, I thank you very much for your continued support and great sacrifices in order to help me overcome all difficulties during my years of studies. I present this success with great respect and appreciation and hope to make you happy and proud of me more one day. ***«God bless you»***.

To my dear sisters and brothers,

Thank you for staying with me and your sincere encouragement to me throughout my studies and during difficult times when I needed you most.

Thank you for staying by my side.

To my whole family,

For their continued support, great confidence in my abilities, love and great appreciation for me, I present them this work and I hope to repay them one day.

To all my friends, and to all my relatives,

For their help and moral support during the development of my graduation project.

To Naima,

Because you are a hardworking and persevering colleague, but you have also proven that you are a great friend

To my dear grandmother, May God bless her and place her in his widest heavens, to my dear grandfather, may God prolong his life, and to all those whose names I have forgotten.

"From the union of 'if' with 'but' was born a child named 'never' "
"There is no 'if' or 'but', you have to succeed"

NORA

Dedicaces

To my dear parents,

There are no words to express what I owe them, for their charity, their friendliness and their support... Treasures of kindness, generosity and tenderness, as a testimony to my deep love and deep respect, I offer them this humble work. « May God protect you » .

To my dear sisters and brothers,

I am grateful for their efforts to help me complete my studies. I dedicate this humble work to them as a testimony to my great love and infinite gratitude.

To all my friends,

For their help and moral support during the development of this work.

To my whole Family

For their continued support, their great confidence in my abilities, their love and their great appreciation for me.

To Nora

Thank you for being always a great friend full of kindness and respect, and thank you for all the efforts you've made in this work.

"From the union of 'if' with 'but' was born a child named 'never' "
"There is no 'if' or 'but', you have to succeed"

NAIMA

Thanks

It is a great pleasure to express through these few lines my great gratitude and deep love to all those I know from near and far and to all those who have contributed to my success.

At first, I would really like to thank ***Mr. Hadj Ahmed Bouarara*** *for his support, supervision, seriousness and kindness, especially for his invaluable assistance throughout the development of this work. I have always considered him an example of a wonderful and respectable professor who, no matter how long it takes, no student will forget it and will remain etched in our memory.*

I would also like to express the honour given to me by the members of the jury by agreeing to judge my work.

Finally, I am happy to perform the duty of gratitude and thanks to all my teachers for the quality of teaching they gave me during my studies in order to provide me with effective training and I want to thank ***Saida university****, especially since it was like my second house where I spent five years full of beautiful memories that I will not forget no matter how long it takes.*

Abstract

Because of the huge technological development that our world is witnessing, access to information of all kinds has become easy and does not require any effort. This has caused great harm to many people and prompted many researchers to search for solutions to limit access to confidential information and use it in the wrong way. In this project, we will try to develop a way in which we can protect people's privacy and prevent the spread of their photos, especially after the large presence of surveillance cameras in public places. Our project relies on automatically masking the identities of faces in photos while preserving the original data distribution. We ensure the complete anonymity of all faces in an image by generating photorealistic images with a conditional generative adversarial network. Our model can remove facial recognition properties while producing high-quality images and videos.

Key-Words : DeepFake, DeepPrivacy, Conditional Identity Anonymization Generative

Résumé

En raison de l'énorme développement technologique que connaît notre monde, l'accès aux informations de toutes sortes est devenu facile et ne nécessite aucun effort. Cela a causé beaucoup de tort à de nombreuses personnes et a incité de nombreux chercheurs à rechercher des solutions pour limiter l'accès aux informations confidentielles et les utiliser à mauvais escient. Dans ce projet, nous essaierons de développer un moyen de protéger la vie privée des personnes et d'empêcher la diffusion de leurs photos, en particulier après la présence massive de caméras de surveillance dans les lieux publics. Notre projet repose sur le masquage automatique de l'identité des visages sur les photos tout en préservant la distribution originale des données. Nous assurons l'anonymat complet de tous les visages d'une image en générant des images photoréalistes avec un réseau contradictoire génératif conditionnel. Notre modèle peut supprimer les propriétés de reconnaissance faciale tout en produisant des images et des vidéos de haute qualité.

Mots clés : DeepFake, DeepPrivacy, Conditional Identity Anonymization Generative

ملخص

بسبب التطور التكنولوجي الهائل الذي يشهده عالمنا، أصبح الوصول إلى المعلومات بجميع أنواعها سهلاً ولا يتطلب أي جهد. وقد تسبب هذا في ضرر كبير لكثير من الناس ودفع العديد من الباحثين للبحث عن حلول للحد من الوصول إلى المعلومات السرية واستخدامها بطريقة خاطئة. سنحاول في هذا المشروع تطوير طريقة يمكننا من خلالها حماية خصوصية الناس ومنع انتشار صورهم، خاصة بعد التواجد الكبير لكاميرات المراقبة في الأماكن العامة. يعتمد مشروعنا على إخفاء هويات الوجوه تلقائيًا في الصور مع الحفاظ على توزيع البيانات الأصلي. نحن نضمن عدم الكشف عن هويته بالكامل لجميع الوجوه في صورة ما من خلال إنشاء صور واقعية باستخدام شبكة خصومة توليدية شرطية. يمكن لنموذجنا إزالة خصائص التعرف على الوجه أثناء إنتاج صور ومقاطع فيديو عالية الجودة.

الكلمات الدالة: التزييف العميق، الخصوصية العميقة، التوليد الشرطي لإخفاء الهوية

Abbreviation list

AES	Advanced Encryption Standard
APN	Almost Perfect Nonlinear
CCPA	California Consumer Privacy Act
CelebA	CelebFaces Attributes
CGAN	Conditional Generative Adversarial Network
CoGAN	Coupled Generative Adversarial Networks
CPRA	California Privacy Protection Agency
CIAGAN	Conditional Identity Anonymization Generative Adversarial Network
DCGAN	Deep Convolutional Generative Adversarial Network
DES	Digital Encryption Standard
DNN	Deep Neural Network
FID	Frechet Inception Distance
GDPR	General Data Protection Regulation
HOG	Histogram of Oriented Gradients
HIPAA	Health Insurance Portability and Accountability Act
JPEG	Joint Photographic Experts Group
LFW	Labeled Faces in the Wild
LRC	Loss Rank Correlation
MAC	Medium Access Control
MIA	Membership inference attacks
MPC	Multi-Party Computation
MTMC	Multi-Target, Multi-Camera
PII	Personally Identifiable Information
PN	Perfect nonlinear
PHI	Protected health information
PNG	Portable Network Graphic
RNN	Recurrent neural network
RGB-D	Red Green Blue-Depth
RGB	Red Green Blue
SFE	Secure Function Evaluation
SIFT	Scale Invariant, and Feature Transform
SSH	Single Stage Headless Face Detector
SVM	Support vector machine
TTS	Text-to-speech
URL	Uniform Resource Locator
VAE	Variational Autoencoder
VC	Voice conversion

Table of contents

Figures table

list of tables

1.1 Context

Because of the great technological advances that the world is currently witnessing and because of the high rates of crime, violence and theft in society, we now see that there is no place without surveillance cameras, especially in areas where there is a large gathering of people, such as schools, roads and shopping malls. The extensive use of closed-circuit cameras and monitoring everywhere and their increasing prevalence have become major concerns for most people's privacy and data protection. Even if these cameras don't use facial recognition technology, the current reality is that there are people looking at these images wherever there are security cameras in large cities, buildings, or other locations. Watching this footage invades the privacy of its subjects, and the information obtained is protected only at the observer's discretion. The anonymity of these images would improve compliance with data protection regulations while increasing public confidence in these monitoring systems.

1.2 Motivation

To protect the privacy of individuals, it has become necessary to remove important personal information represented by facial information, which can be done through simple techniques such as blur and obfuscation. Still, these techniques remain unsecured and can fail and reveal some parts of the face, especially when a person moves.

1.3 Problem of the thesis

One of the major challenges that scientists are now facing is how to protect the privacy of individuals by removing important personal information represented by facial information and can be done through simple techniques such as blur and obfuscation, but these techniques remain uncertain and can fail and reveal some information.

1.4 Objective

It was, therefore, necessary to look for other ways and alternatives that would ensure that the identity of the person remained hidden. Parts of the face, especially when a person moves. In our research, we will talk about the technique of replacing real faces with artificial faces that are created in a good way based on falsification to appear more realistically. In this way, we have protected people's privacy and prevented anyone from identifying people. In the case of any suspicious activity, the face will be immediately revealed and returned to its original form to find out the identity of this person.

1.5 Organization of the thesis

This thesis is divided into 6 chapters:

Chapter 2: Preservation of life privacy

Chapter 3: Generative Adversarial Network

Chapter 4: Proposed Approach

Chapter 5: Results, experimentation and comparison

Chapter 6: General Conclusion

2.1 Introduction

Suppose the principle of the protection of private life aims to protect everyone against any form of interference in their personal life. In that case, it is mainly against revelations made by the press that the legal texts have been put in place. This privacy protection has constantly been threatened and is even more so today, with the increase in the influence of social networks, the decline in moral constraints and the development of new investigative techniques.
At a time when the world is facing many challenges linked to the development of new technologies, social networks, and the increase in increasingly specialized but also intrusive surveillance measures, it is essential that the balance between the right to security and the right to privacy is maintained and that the protection of privacy remains at the heart of the policies put in place. The right to respect for private life is mentioned very regularly in the opinions of the CNCDH, which recalls its attachment to Article 8 of the European Convention on Human Rights. The right to respect for private life is affirmed by Article 12 of the Universal Declaration of Human Rights of 1948 and codified in French law in article 9 of the Civil Code, which provides that "Everyone has the right to respect for his private life. »
The state must protect the privacy of its citizens and ensure that invasions of privacy if they do occur, are necessary and proportionate. The CNCDH integrates broad themes regarding private and family life because it is a right that many measures can violate.

2.2 Access to private data

The increased power and interconnectivity of computer systems available today allow the ability to store and process large amounts of data, resulting in networked information accessible from anywhere. Indeed, thanks to the availability of post-third generation mobile networks, user transactions are no longer bound to the traditional office-centered environment but can be started virtually anywhere and at any hour. Resources may then be accessed in various contexts, and users requesting access may be required to disclose a rich set of distributed information about themselves, including dynamic properties such as their location or communication device and conventional, identity-related user attributes. The vast amounts of personal details thus available have led to growing concerns about their users' privacy. Therefore, personal information privacy is an issue that most people are concerned about, mainly because the possibilities of information distribution, combination, and reuse have been increased [1] .

In such a scenario, information privacy is about the collection, processing, use, and protection of personal information and should also be addressed by developing privacy-aware languages and policies that encompass two notions [1] **:**

- Guaranteeing the desired level of privacy of information exchanged between different parties and controlling access to services/resources based on this information.
- Managing and storing personal information given to small parties responsibly. A privacy-aware solution should combine these two notions and be expressive and straightforward enough to support the following functionality.

2.3 Differential privacy

Differential privacy is the technology that enables researchers and database analysts to obtain useful information from the databases containing people's personal information without divulging their personal identification. We can describe it as a promise made by a data holder, or curator, to a data subject (owner). The promise is like this: “You will not be affected adversely or otherwise by allowing your data to be used in any study or analysis, no matter what other studies, datasets or information sources are available” [2] .

This can be achieved by introducing a minimum distraction in the information given by the database. The introduced distraction is immense enough to protect the privacy and, at the same time, limited sufficient so that they provide information to analysts is still valid [2] .

Differential privacy can be applied to everything from social networks to location-based services. For example:

- Apple employs differential privacy to accumulate anonymous usage insights from devices like iPhones, iPads and Mac.
- Amazon uses differential privacy to access users’ personalized shopping preferences while covering sensitive information regarding their past purchases.
- Facebook uses it to gather behavioral data for target advertising campaigns without defying any nation’s privacy policies.

For example, consider an algorithm that analyzes a dataset and computes its statistics such as mean, median, mode, etc. Now, this algorithm can be regarded as differentially private only if via examining the output if a person cannot state whether any individual’s data was included in the actual dataset or not [2] .

What does it guarantee?

- Differential privacy guarantees mathematically that a person observing the outcome of the private differential analysis will likely produce the same inference about an individual’s private information, whether or not that individual’s private information is combined in input for the analysis.

- It also specifies verified mathematical assurance of privacy protection counter to a considerable range of privacy attracts such as a differencing attack, linkage attacks, etc.

2.4 Secure multiparty computation

Secure multi-party computation (also known as secure computation, multi-party computation (MPC) or privacy-preserving computation) provides a cryptographic protocol where no individual can see the other parties data while distributing the data across multi parties. It enables the data scientists and analysts to compute privately on the distributed data without exposing it.

Secure multiparty computation started early in the 1970s. It was known as multiparty computation at that time. It did not gain popularity at that time as it was not implemented practically. In the 1982's, it was introduced as secure two-party multiparty computation. It is used to solve a lot of computation problems without revealing the inputs to other parties. Finally, it came with a name as secure multiparty computation in which the functions of different types are computed; that is why it is sometimes called **SFE- Secure Function Evaluation.**

- The secure multiparty computation is used to utilize data without compromising privacy.
- It is the cryptographic subfield that helps preserve the privacy of the data.
- Emerging technologies like blockchain, mobile computing, IoT, and cloud computing have resulted in the rebirth of secure multiparty computation.
- Secure multiparty computation has become the hot area of research in the last decade due to the rise of blockchain technology.

2.4.1 Secure multiparty computation Protocol

In an MPC, a given number of participants, p1, p2, ..., pN, each have private data, respectively d1, d2, ..., dN. Participants want to compute the value of a public function on that private data: F(d1, d2, ..., dN) while keeping their own inputs secret.

For example, suppose we have three parties Alice, Bob and Charlie, with respective inputs x, y and z denoting their salaries. They want to find out the highest of the three salaries, without revealing to each other how much each of them makes. Mathematically, this translates to them computing: F(x, y, z) = max(x, y, z).

If there were some trusted outside party (say, they had a mutual friend Tony who they knew could keep a secret), they could each tell their salary to Tony, and he could compute the maximum and say that number to all of them. The goal of MPC is to design a protocol where, by exchanging messages only with each other, Alice, Bob, and Charlie can still learn F(x, y, z) without revealing who makes what and without having to rely on Tony. They should know no more by engaging in their protocol than they would learn by interacting with an incorruptible, perfectly trustworthy Tony.

In particular, all that the parties can learn is what they can learn from the output and their own input. So, in the above example, if the output is z, then Charlie learns that his z is the maximum value, whereas Alice and Bob learn (if x, y and z are distinct) that their input is not equal to the maximum, and that the maximum held is equal to z. The basic scenario can be easily generalized to where the parties have several inputs and outputs, and the function outputs different values to different parties.

Informally speaking, the most fundamental properties that a multi-party computation protocol aims to ensure are:

- ✓ **Input Privacy:** No information about the private data held by the parties can be inferred from the messages sent during the execution of the protocol. The only information that can be inferred about the private data is what could be inferred from seeing the function's output alone.

- ✓ **Correctness:** Any proper subset of opposing colluding parties willing to share information or deviate from the instructions during the protocol execution should not be able to force honest parties to output an incorrect result. This correctness goal comes in two flavors: either the honest parties are guaranteed to compute the correct output (a “robust” protocol), or they abort if they find an error (an MPC protocol “with abort”).

2.5 Data Anonymization

Data anonymization is the process of protecting private or sensitive information by erasing or encrypting identifiers that connect an individual to stored data. For example, you can run Personally Identifiable Information (PII) such as names, social security numbers, and addresses through a data anonymization process that retains the data but keeps the source anonymous [3] .

Data anonymization techniques alter data across systems so it can't be traced back to a specific individual while preserving the data's format and referential integrity. It is one of several approaches organizations can use to comply with stringent data privacy laws that require the protection of personally identifiable information (PII) such as contact information, health records, or financial details [3] .

2.5.1 Data Anonymization Techniques

Data masking: hiding data with altered values. You can create a mirror version of a database and apply modification techniques such as character shuffling, encryption, and word or character substitution. For example, you can replace a value character with a symbol like "*" or "x." Data masking makes reverse engineering or detection impossible [3] .

Pseudonymization: a data management and de-identification method that replaces private identifiers with fake identifiers or pseudonyms, for example, replacing the identifier "John Smith" with "Mark Spencer." Pseudonymization preserves statistical accuracy and data integrity, allowing the modified data to be used for training, development, testing, and analytics while protecting data privacy [3] .

Generalization: deliberately removes some of the data to make it less identifiable. Data can be modified into a set of ranges or a broad area with appropriate boundaries. You can remove the house number in an address, but make sure you don't remove the road name. The purpose is to eliminate some of the identifiers while retaining a measure of data accuracy [3] .

Data swapping: also known as shuffling and permutation, is a technique used to rearrange the dataset attribute values, so they don't correspond with the original records. Swapping attributes (columns) that contain identifier values such as date of birth, for example, may have more impact on anonymization than membership type values [3] .

Data perturbation: modifies the original dataset slightly by applying techniques that round numbers and add random noise. The range of values needs to be in proportion to the perturbation. A small base may lead to weak anonymization, while a large base can reduce the dataset's utility. For example, you can use a base of 5 for rounding values like age or house number because it's proportional to the original value. You can multiply a house number by 15, and the value may retain its credence. However, using higher bases like 15 can make the age values seem fake [3] .

Synthetic data: algorithmically manufactured information that has no connection to real events. Synthetic data is used to create artificial datasets instead of altering the original dataset or using it as is and risking privacy and security. The process involves creating statistical models based on patterns found in the original dataset. You can use standard deviations, medians, linear regression or other statistical techniques to generate the synthetic data [3] .

2.6 Example of data anonymization in surveillance video

Concerning the protection of observed people's privacy, smart video surveillance implies both risk and chance. The systems' increasing capabilities can be misused if we do not deploy suitable counter-measures. On the other hand, when designed with care, such systems can be both privacy-aware and powerful. Smart video surveillance systems aim to assist operators by pointing their attention to certain events that have been detected by intelligent algorithms. From a privacy-protection perspective, video data should only be released in an anonymized fashion, at least until a human operator evaluates whether an event is critical: First-level situation assessment is a matter of what people are doing and not of who people are. The rationale behind privacy-aware smart video surveillance designs is to prevent groundless and unjustified intrusions into people's privacy. In their legal analysis of smart video surveillance according to European data protection law, Bretthauer and Krempel emphasize that people must not be subjected to automated individual decisions made by a system processing personal information. Thus, assessing whether a situation is critical has to be in the hands of a human operator [4] .

FIGURE 2.1: An image Picture taken from a surveillance camera

2.6.1 Detecting and Obfuscating RoIs

Our method for anonymizing videos is as follows. We employ a background estimator for learning the longterm background of the scene as well as for detecting

foreground objects, i.e., our regions of interest (RoI) for anonymization. We extract the RoIs from each frame, apply obfuscating image filters or pixel operations, and finally put the anonymized RoIs onto the long-term background model of the scene. With this approach we reduce the risk of privacy breaches due to missing a RoI in some frames. In terms of obfuscating image filters and pixel operations we use pixelization, edge detection, Gaussian blurring, which we apply either on color or on gray-scale RoIs, and reducing the RoI to its silhouette (cf. Fig. 2.2). It also seems natural to combine such operations, e.g., by applying the silhouette technique first and then employ Gaussian blurring afterwards in order to erase smaller structures at the border of the silhouette, which otherwise may reveal visual information about clothes and/or body shapes of persons. However, the focus of our first study is to investigates the effects of each anonymization technique separately [4] .

FIGURE 2.2: Anonymization techniques: original pixelation ,gray blurring ,color gray blurring, edge detection.

2.7 Privacy preserving encrypted

Data security is critical not only for businesses but also for home computer users. From client to payment information, personal files, or bank account details, all this information can be hard to replace and potentially dangerous if it falls into the wrong hands. Indeed, data lost due to disasters, such as a flood or fire, is crushing, but losing it because of hackers or malware infection can have much greater consequences. Therefore, data security is the science that studies the methods to protect data in computer and communication systems. It embodies cryptographic controls to face security threats. Cryptography is the science of using mathematics to encrypt and decrypt data to keep messages secured by transforming intelligible data form (plaintext) into an unintelligible form (ciphertext). Every cryptosystem consists of five elements: plaintext, encryption algorithm, decryption algorithm, ciphertext, and Key. The plaintext is messages or data that are in their normal, readable (not encrypted) form. Encryption is the process of converting plaintext to ciphertext by using a key. Ciphertext results from the encryption performed on plaintext using an algorithm called a cipher. Decryption is the process of retrieving the plaintext back from the ciphertext. Finally, the Key uses the information to control the cryptosystem (cipher system), and the sender and the receiver only know it [5] .

2.7.1 Cryptography in data security

Lawrence Lessig (1999) wrote, “Encryption technologies are the most important technological breakthrough in the last one thousand years.” 5 It might seem a slight exaggeration, but it emphasizes the importance of encryption technologies in today’s digital world. Indeed, encrypted data play a significant role in protecting data subjects’ privacy. Cryptography is the science of using mathematics to encrypt and decrypt data. While cryptography is the science of securing data, cryptanalysis is the science of analyzing and breaking secure communication. Classical cryptography provided secrecy for information sent over channels where eavesdropping and message interception was possible. The sender selected a cipher and an encryption key and either gave it directly to the receiver or else sent it indirectly over a slow but secure channel - typically a trusted courier. Modern cryptography protects data transmitted over high-speed electronic lines or stored in computer systems. Figure 2.3 below shows how a classical information channel works [5].

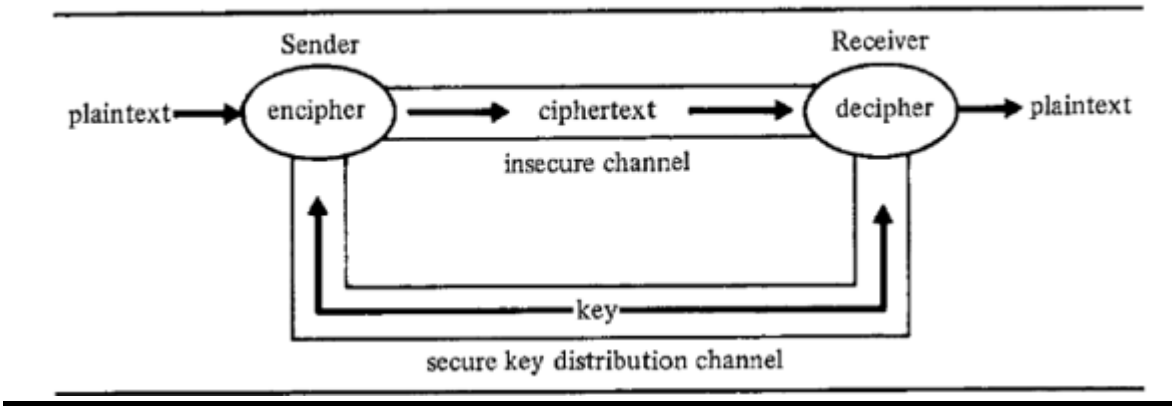

FIGURE 2.3: Classical information channel

2.8 Encryption and nonlinear functions

2.8.1 Nonlinear functions

In mathematics and science, a nonlinear system is a system in which the output change is not proportional to the change of the input. Nonlinear problems are of interest to cryptographers, engineers, biologists, physicists, mathematicians, and many other scientists because most systems are inherently nonlinear in nature [5] .

2.8.2 Nonlinear functions in cryptography

In the late 1980s, the importance of highly nonlinear functions in cryptography was first discovered by Meier and Staffelbach from the point of view of correlation attacks on stream ciphers and later by Nyberg in the early 1990s after the introduction of the differential cryptanalysis method. Perfect nonlinear (PN) and almost perfect nonlinear (APN) functions, which have the optimal properties for offering resistance against differential cryptanalysis, have since then been an object of intensive study by many mathematicians. In this paper, we survey some of the theoretical results obtained on these functions in the last 25 years. We recall how the links with other mathematical concepts have accelerated the search for PN and APN functions. To illustrate the use of PN and APN functions in practice, we discuss examples of ciphers and their resistance to differential attacks. In particular, we recall that in cryptographic applications, suboptimal functions are often used [5] .

2.8.3 How Linear Function Makes Encryption strong

Nonlinearity is crucial since most linear systems are easily breakable. As a cipher that explicitly follows the principle of confusion and diffusion, we mention DES. Likewise, this concept applies to other cryptosystems, block ciphers, and stream ciphers [5] .

The theory of perfect nonlinear (or bent) functions has interesting implications for the design of block ciphers as well as stream ciphers. Nonlinearity may not be compatible with other cryptographic design criteria. For example, perfect nonlinearity cannot be achieved in conjunction with balance or the highest nonlinear order. However, a reasonable strategy will be to find nearly perfect nonlinear functions which satisfy additional design criteria. This is illustrated by the following example of finding almost perfect nonlinear functions which are balanced [5] .

If you combine functions linear of the same field you get again a function linear over that field, and linear equations are easy to solve. So, you do need something non-linear to make solving difficult [5] .

2.8.4 Example of nonlinear function in AES :

The non-linear operation is **AES's S-box**, which is a finite-field inverse $S(x) = x^{-1}$. You can see that the S-box is non-linear because it is not necessarily true that $(x+y)^{-1} = x^{-1}+y^{-1}$.

2.9 Data Augmentation and privacy

Data augmentation is a strategy that enables practitioners to significantly increase the diversity of data available for training models, without actually collecting new data. Data augmentation techniques such as cropping, padding, and horizontal flipping are commonly used to train large neural networks [6] .

Data augmentation is often necessary when confronted with a massive domain of use like an open world as it improves the robustness of a neural network. Data augmentation is useful but can be time-consuming thus slowing down your development. In order to scale down the cost of training, you need adequate data augmentation [6] .

2.9.1 Data augmentation in privacy

Deep learning models often raise privacy concerns as they leak information about their training data. This leakage enables membership inference attacks (MIA)
that can identify whether a data point was in a model's training set. Research shows that some 'data augmentation' mechanisms may reduce the risk by combatting a key factor increasing the leakage, overfitting. While many mechanisms exist, their effectiveness against MIAs and privacy properties have not been studied systematically. Employing two recent MIAs, we explore the lower bound on the risk in the absence of formal upper bounds. First, we evaluate 7 mechanisms and differential privacy, on three image classification tasks. We find that applying augmentation to increase the model's utility does not mitigate the risk and protection comes with a utility penalty. Further, we also investigate why popular label smoothing mechanism consistently amplifies the risk. Finally, we propose 'loss-rank-correlation' (LRC) metric to assess how similar the effects of different mechanisms are. This, for example, reveals the similarity of applying high-intensity augmentation against MIAs to simply reducing the training time. Our findings emphasize the utility-privacy trade-off and provide practical guidelines on using augmentation to manage the trade-off [6] .

2.9.2 The way it works

Computer vision applications use common data augmentation methods for training data. There are classic and advanced techniques in data augmentation for image recognition and natural language processing (NLP).

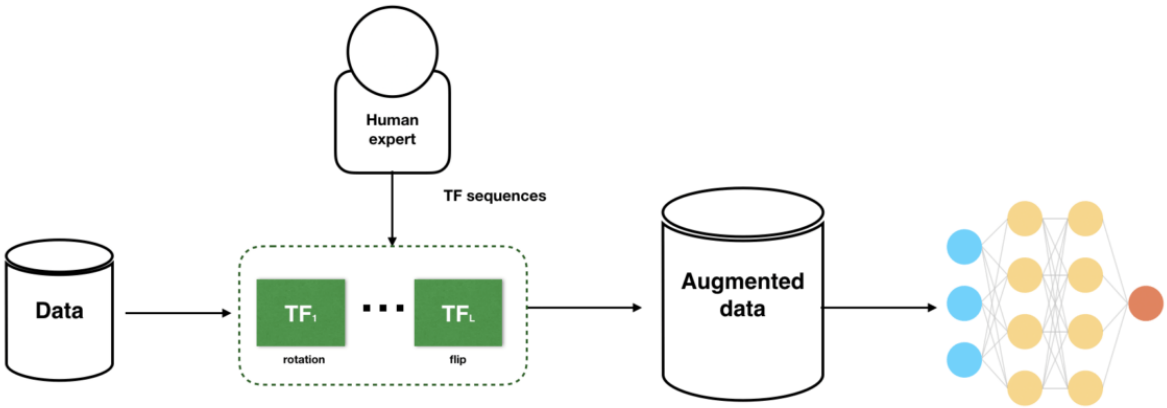

FIGURE 2.4: Data augmentation methods

2.10 Deidentification

Data de-identification refers to breaking the link between data and the individual with whom the data is initially associated. Essentially, this requires removing or transforming personal identifiers. Once personal identifiers are removed or transformed using the data de-identification process, it is much easier to reuse and share the data with third parties.

Data de-identification is expressly governed under HIPAA, which is why most people associate the data de-identification process with medical data. However, data de-identification is also important for businesses or agencies that want or need to mask identities under other frameworks, such as CCPA and CPRA, or even GDPR.

When applied to metadata or general data about identification, the process is also known as data anonymization. Common strategies include deleting or masking personal identifiers, such as personal name, and suppressing or generalizing quasi-identifiers, such as date of birth. The reverse process of using de-identified data to identify individuals is known as data re-identification. Successful re-identifications cast doubt on de-identification's effectiveness. A systematic review of fourteen distinct re-identification attacks found "a high re-identification rate […] dominated by small-scale studies on data that was not de-identified according to existing standards".

HIPAA: names two different methods of de-identifying data: Safe Harbor and Expert Determination.

1-Safe Harbor

The Safe Harbor method of de-identification requires removing 18 types of identifiers, like those listed below, so that residual information cannot be used for identification:

- Names
- Dates, except the year
- Telephone numbers
- Geographic data
- Fax numbers
- Social Security numbers
- Email addresses
- Medical record numbers
- Account numbers
- Health plan beneficiary numbers
- Certificate/license numbers
- Vehicle identifiers and serial numbers, including license plates
- Web URLs
- Device identifiers and serial numbers
- Internet protocol addresses
- Full face photos and comparable images
- Biometric identifiers
- Any unique identifying number, characteristic, or code

Any of these identifiers can classify health information as protected health information (PHI), which limits use and disclosure. Data de-identification tools with sensitive data discovery can detect and mask such information.

The Safe Harbor method, which is usually praised for its simplicity and low cost, is not well adapted for all use cases: It can either be overly restrictive, leaving too little utility within the data, or overly permissive, leaving too many indirect identifiers in the clear.

2- Expert Determination

Expert determination involves applying statistical and scientific principles to data to achieve a very small risk of re-identification. This method makes it possible to tailor the de-identification process to the use case at hand while also maximizing utility; it is therefore praised for its flexibility.

Expert determination is sometimes considered too costly to use because it requires the involvement of an expert in statistics, who can be expensive to source. However, the expert determination method enables the use of quantitative methods to lower the re-identification risk, which opens the door for leveraging generalization and automation.

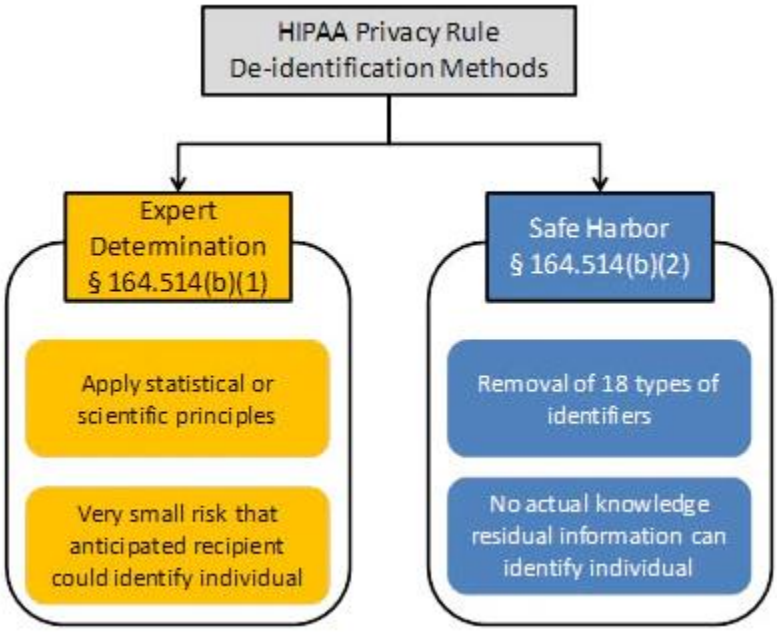

FIGURE 2.5: Two methods to achieve de-identification in accordance with the HIPAA Privacy Rule.

2.10.1 Techniques of de-identification

Common strategies of de-identification are masking personal identifiers and generalizing quasi-identifiers. Pseudonymization is the main technique used to mask personal identifiers from data records, and k-anonymization is usually adopted for generalizing quasi-identifiers:

✓ Pseudonymization

Pseudonymization is performed by replacing real names with a temporary ID. It deletes or masks personal identifiers to make individuals unidentified. This method makes it possible to track the individual's record over time even though the record will be updated. However, it cannot prevent the individual from being identified if some specific combinations of attributes in the data record indirectly identify them.

✓ Generalizing (k-anonymization)

K-anonymization is a data generalization technique that is implemented once direct identifiers have been masked. The k-anonymization process reduces re-identification risks by hiding individuals in groups and suppressing indirect identifiers for groups smaller than a predetermined number, *k*. This aims to mitigate identity and relational inference attacks. This de-identification technique can help reduce the need for data redaction in data sets, which helps increase its utility without compromising data privacy.

✓ Randomizing (differential privacy and randomized response)

Differential privacy is a randomization technique that is implemented once direct identifiers have been masked. There are two approaches to differential privacy: local and global.

Local differential privacy is a data randomization method that usually applies to sensitive attributes and offers a mathematical guarantee against attribute-based inference attacks. This is accomplished by randomizing attribute values in a way that limits the amount of personal information inferable by an attacker while still preserving some analytic utility, since gathering too much information on a specific record can undermine privacy. Individuals whose data is included in the queried data set are therefore able to deny the specific attributes attached to their records. Technology companies like Google and Apple, which collect a wide range and huge amount of personal data, have adopted local differential privacy.

Global differential privacy is a method that randomizes aggregate data. This approach constrains data users to only formulate aggregate queries (e.g., count, average, max, min, etc.), and offers a mathematical guarantee against identity-, attribute-, participation-, and relational-based inference attacks. Individuals whose data is included in the queried data set are therefore able to deny their participation in the data set as a whole. The US Census Bureau, for example, employs global differential privacy because aggregation on its own is insufficient to preserve privacy.

2.11 Fingerprint privacy

The impression made by the papillary ridges on the ends of the fingers and thumbs. Fingerprints afford an infallible means of personal identification because the ridge arrangement on every finger of every human being is unique and does not alter with growth or age. Fingerprints serve to reveal an individual's true identity despite personal denial, assumed names, or changes in personal appearance resulting from age, disease, plastic surgery, or accident. Utilizing fingerprints as a means of identification, referred to as dactyloscopy, is an indispensable aid to modern law enforcement.

Each ridge of the epidermis (outer skin) is dotted with sweat pores for its entire length and is anchored to the dermis (inner skin) by a double row of peglike protuberances, or papillae.

Any ridged area of the hand or foot may be used as identification. However, finger impressions are preferred to those from other parts of the body because they can be taken with a minimum of time and effort. The ridges in such impressions form patterns (distinctive outlines or shapes) that can be readily sorted into groups for ease in the filing.

Fingerprints are classified in a three-way process: by the shapes and contours of individual patterns, by noting the finger positions of the pattern types, and by relative size, determined by counting the ridges in loops and by tracing the ridges in whorls. The information obtained in this way is incorporated into a concise formula, which is known as the individual's fingerprint classification.

2.11.1 Fingerprint identification

A person's fingerprint is compared to stored fingerprint data in fingerprint identification. A fingerprint can, for example, be stored in an identification system database, a passport chip, or an access card's memory. Identification locations include a door side fingerprint reader, a reader connected to a computer, or a fingerprint reader integrated into a smartphone.

Fingerprint identification is nearly always combined as a part of another system, such as a locking system.

Fingerprint identification is used to verify a person's identity, after which the system can perform the required actions, such as opening a lock, allowing a person to use software, or enabling a machine to start.

2.11.2 Fingerprint identification implementation

Fingerprint identification is based on pattern recognition, where the arches, loops and whorls of the fingerprint ridges are compared with stored data. Identification is performed in three parts:

1. A picture is taken of the fingerprint. The image can be taken optically with a camera in the reader or electronically or combined with these two methods. The end result is a digital black and white photograph of the ridges in the fingerprint.

2. The fingerprint is then transformed into a numerical model which stores the fingerprint's unique characteristics, such as the arches and loops and their distance from each other, as a series of numbers.

3. A recognized numerical model is compared with a stored numerical model (or models) to find similarities.

2.12 Conclusion

In this section, we have tried to highlight the most important means and techniques like Privacy preserving encrypted, data anonymization and fingerprint used by researchers to preserve information of all kinds and prevent its spread illegally and how to preserve an individual's private life.

3.1 Introduction

One of the most prominent aspects of technology that has sparked widespread controversy in recent times and has become life-threatening for the public in general, celebrities and politicians in particular, is the technique of deep fake.

3.2 Deepfake

The term "deep fake" refers to the use of fake AI-based videos, allowing a person to manipulate a specific personality's face and apply their features to another character with common features in a video. The seriousness of this technique lies in fabricating videos of important figures in the country or fabricating a complete video that offends a particular person, and the problem is the difficulty of detecting the video because of its proximity to the truth. Hollywood directors spent millions of dollars on special effects to make a real scene. It became easy to achieve with a few dollars, one computer, and little time by some amateurs who could master this dangerous technique.

FIGURE 3.1 : Example of an image generated by Deepfake

3.3 The techniques of deep fake

A few years ago, making fake videos required graphic designer image processing and manipulation skills, which took a lot of time. Today, techniques have evolved, and new skills are needed. In particular, to train mathematical and algorithmic models and perfect models to generate false images.

The computing power of computers offered by cloud computing, combined with recent research in deep learning, has made it possible to make tricks more and more realistic. The abundance of data on social networks and on the internet offered enough material to train the models (i.e., the algorithms “learn” through “training” on datasets).
The techniques for creating deep fake videos are numerous**:**

3.3.1 Face changing (i.e., face-swapping):

Face-swapping is one of the techniques used to create deep fake videos. It is based on putting the face of a Specific Person on another person's body. This method is relatively simple and accessible on mobile applications of short videos like Snapchat, Face Swap Live, reface and face magic. It is based on the use of an auto-encoder, itself made up of an encoder and a decoder [7] .

The encoder describes what the person is doing in the video and extracts the most important information. For its part, the decoder will then try to reconstruct the image.

To simplify, think of a crime scene. The encoder would be the witness describing the scene, and the decoder would be the person sketching the suspect from the description. This requires large image databases (of the order of several 10ths of thousands as input to train the model) [7] .

Step 1
The image region showing the original face is extracted from an original frame of the video. This image acts as input to an Encoder or Deep Neural Network (DNN), a technique from the domain of machine learning and artificial intelligence. A latent Face image with dormant features is created, which acts as an input to the Decoder. A reconstructed Face image is produced for both the images (Face A and Face B)[7] .

Step 2
The DNN automatically generates a matching image showing the second face or the reconstructed Face [7] .

Step 3
The face that is generated is inserted into the original reference image to produce the Deepfake or FaceSwap. Using the concepts of Deep learning (a machine learning method used to train deep neural networks (DNNs), this technology efficiently replaces the faces from the images or videos, and imagining the change is practically impossible due to its accuracy. Exceptional results are derived using AI (Artificial Intelligence) and machine learning techniques [7] .

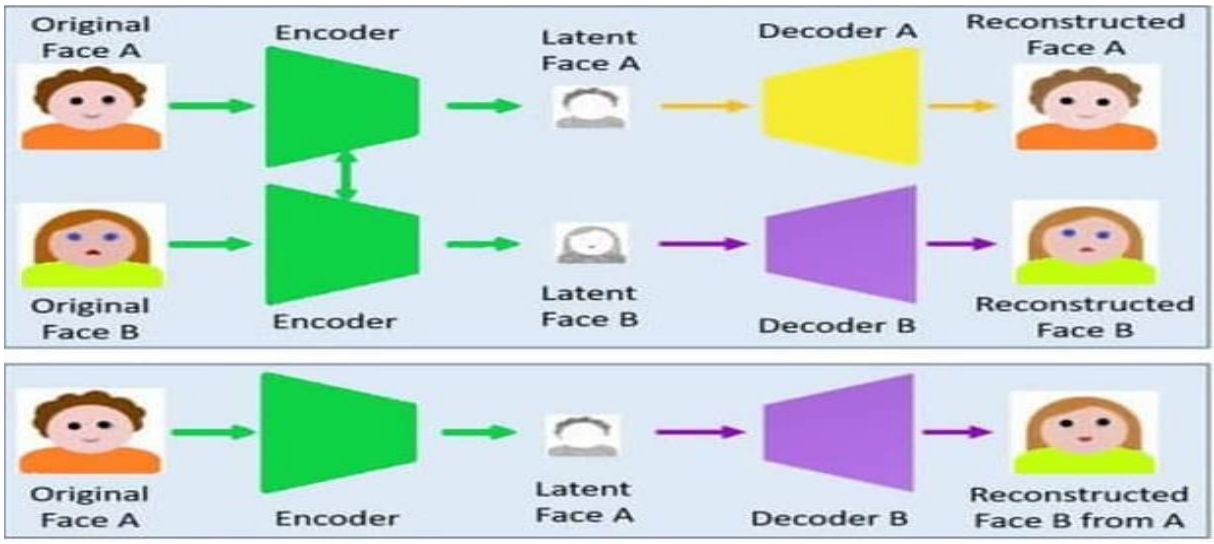

FIGURE 3.2 : Face-swapping technique

3.3.2 The puppet-master :

Puppet-master, also known as face reenactment, is another common variation of deep fakes that manipulates the facial expressions of a person, e.g., transferring the facial gestures, eye, and head movements to an output video that reflects those of the source actor. Puppet-mastery aims to deform the person's mouth movement to make fabricated content. Facial reenactment has various applications, i.e., altering the facial expression and mouth movement of a participant to a foreign language in an online multilingual video conference, dubbing or editing an actor's head and their facial expressions in film industry post-production systems, or creating photorealistic animation for movies and games, etc. Initially, 3D facial modeling-based approaches for facial reenactment were proposed because of their ability to capture the geometry and movement accurately and for improved photorealism in reenacted faces. Thies et al. presented the first real-time facial expressions transfer method from an actor to a target person. A commodity RGB-D sensor was used to track and reconstruct the 3D model of a source and target actor. For each frame, the tracked deformations of the source face were applied to the target face model. Later, the altered face was blended onto the original target face while preserving the facial appearance of the target face model. Face2Face is an advanced form of facial reenactment technique as presented in. This method works in real-time and can alter the facial movements of generic RGB video streams, e.g., YouTube videos, using a standard webcam. The 3D model reconstruction approach was combined with image rendering techniques to generate the output. This creates a convincing and instantaneous re-rendering of the target actor with a relatively simple home setup. This work was further extended to control a person's facial expressions in a target video based on intuitive hand gestures using an inertial measurement unit [7] .

GANs have been successfully applied for facial reenactment due to their ability to generate photo-realistic images [7] .

Pix2pixHD produces high-resolution images with better fidelity by combining multi-scale conditional GANs (cGAN) architecture with perceptual loss. Kim et al. proposed an approach that allows the complete reanimation of portrait videos by an actor, such as changing head pose, eye gaze, and blinking, rather than just modifying the facial expression of the target identity and thus producing photorealistic dubbing results. At first, a face reconstruction approach was used to obtain a parametric representation of the face and illumination information from each video frame to produce a synthetic rendering of the target identity. This representation was then fed to a render-to-video translation network based on the cGAN to predict the synthetic rendering into photo-realistic video frames. This approach requires training the videos for target identity. Wu et al. proposed ReenactGAN, which encodes the input facial features into a boundary latent space. A target-specific transformer was used to adapt the source boundary space according to the specified target, and later the latent space was decoded onto the target face. GANimation employed a dual cGAN generator conditioned on emotion action units (AU) to transfer facial expressions. The AU-based generator used an attention map to interpolate between the reenacted and original images. Instead of relying on AU estimations, GAN notation used facial landmarks along with the self-attention mechanism for facial reenactment. This approach introduced a triple consistency loss to minimize visual artifacts but requires the images to be synthesized with a frontal facial view for further processing. These models require a large amount of training data for the target identity to perform [7] .

3.3.3 Lip reconstruction (lip-sync):

The Lip-syncing approach involves synthesizing a video of a target identity such that the mouth region in the manipulated video is consistent with arbitrary audio input (Fig3.3).

A key aspect of synthesizing a visual speech is the movement and appearance of the lower portion of the mouth and its surrounding region. To convey a message more effectively and naturally, it is important to generate proper lip movements and expressions. From a scientific point of view, lip-syncing has many applications in the entertainment industry, such as making audio-driven photorealistic digital characters in films or games, voice-bots, and dubbing films in foreign languages. Moreover, it can also help hearing-impaired persons understand a scenario by lip-reading from a video created using genuine audio [7] .

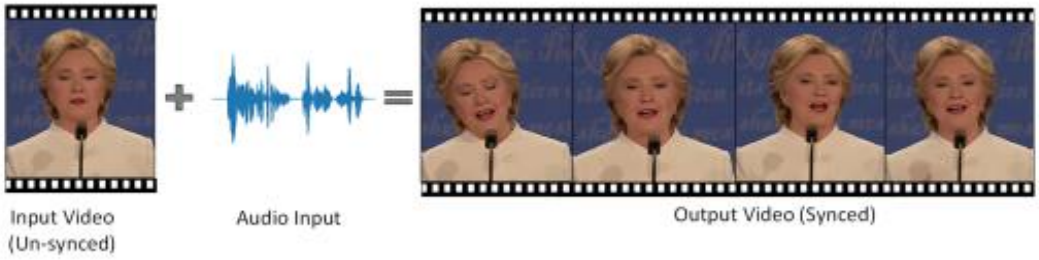

FIGURE 3.3: A visual representation of lip-syncing of an existing video to an arbitrary audio clip

Existing works on lip-syncing require the reselection of frames from a video or transcription, along with target emotions, to synthesize the lip's motion. These approaches are limited to a dedicated emotional state or don't generalize well to unseen faces. However, the DL models can learn and predict the movements from audio features. Suwajanakorn proposed an approach to generate a photo-realistic lip-synced video using a target's video and an arbitrary audio clip as input. The recurrent neural network (RNN) based model was employed to learn the mapping between audio features and mouth shape for every frame and later used frame reselection to fill in the texture around the mouth based on the landmarks. This synthesis was performed on the lower facial regions, i.e., mouth, chin, nose, and cheeks. This approach applied a series of post-processing steps, such as smoothing jaw location and re-timing the video to align vocal pauses or talking head motion, to produce videos that appear more natural and realistic. In this work, Barak Obama was considered as a case study due to the sufficient availability of online video footage [7].

3.3.4 Face Synthesis and Attribute Editing :

Facial editing in digital images has been heavily explored for decades. It has been widely adopted in the art, animation, and entertainment industry. However, lately, it has been exploited to create deep fakes. Face manipulation can be broadly grouped into two categories: face generation and face attribute editing. Face generation involves the synthesis of photorealistic images of a human face that doesn't exist in real life. In contrast, face attribute editing involves altering the facial appearance of an existing sample by modifying the attribute-specific region while keeping the irrelevant regions unchanged. Face attribute editing includes removing/wearing eyeglasses, changing viewpoint, skin retouching (e.g., smoothing skin, removing scars, and minimizing wrinkles), and even some higher-level modifications, such as age and gender, etc. Increasingly, people have been using commercially available AI-based face editing and mobile applications such as FaceApp to automatically alter the appearance of an input image. The tremendous evolution in deep generative models has made them widely adopted tools for image synthesis and editing[7] .

Generative deep learning models, i.e., GAN and VAE, have been successfully used to generate photorealistic fake human face images. In facial synthesis, the objective is to generate non-existent but realistic-looking faces. Face synthesis has enabled a wide range of beneficial applications, like automatic character creation for video games and 3D face modeling industries. AI-based face synthesis could also be used for malicious purposes, as the synthesis of photorealistic fake images for social network accounts with a false digital identity to spread misinformation. Several approaches have been proposed to generate realistic-looking facial images that humans cannot recognize as to whether they are real or synthesized.(Fig 3.4) shows synthetic facial images and the improvement in their quality between 2014 and 2019 that are nearly indistinguishable from real photographs [7] .

Figure 3.4: Increasingly improving improvements in the quality of synthetic faces, as generated by variations on GANs. In order, the images are from papers by Goodfellow et al. (2014) , Radford et al. (2015) , Liu et al. (2016) , Karras et al. (2017) , and Style-based (2018 , 2019)

Since the emergence of GAN in 2014, significant efforts have been made to improve the quality of synthesized images. The images generated using the first GAN model were low-resolution and not very convincing. DCGAN was the first approach that introduced a deconvolution layer in the generator to replace the fully connected layer, which achieved better performance in synthetic image generation. Liu et al. proposed CoGAN, based on VAE, for learning joint distributions of two-domain images. This model trained a couple of GANs rather than a single one, and each was responsible for synthesizing images in one domain. The size of generated images still remained relatively small, e.g., 64×64 or 128×128 pixels [7] .

3.3.5 Audio Deepfakes

AI-synthesized audio manipulation is a type of deep fake that can clone a person's voice and depict that voice saying something outrageous that the person never said. Recent advancements in AI-synthesized algorithms for speech synthesis and voice cloning have shown the potential to produce realistic fake voices that are nearly indistinguishable from genuine speech. These algorithms can generate synthetic speech that sounds like the target speaker based on text or utterances of the target speaker, with highly convincing results [7].

The synthetic voice is widely adapted for developing different applications, such as automated dubbing for TV and film, chatbots, AI assistants, text readers, and personalized synthetic voices for vocally handicapped people. Aside from this, synthetic/fake voices have become an increased threat to voice biometric systems and are being used for malicious purposes, such as political gains, fake news, fraudulent scams, etc [7] .

More complex audio synthesis could be combining the power of AI and manual editing. For example, neural network-powered voice synthesis models, such as Google's Tacotron, Wavenet or AdobeVoco, can generate realistically and convincing fake voices that resemble the victim's voice as the first step. Later on, audio editing software, e.g., Audacity, can be used to combine the different pieces of original and synthesized audios to make more powerful audios [7] .

AI-based impersonation is not limited to visual content; recent advancements in AI-synthesized fake voices are assisting the creation of highly realistic deep fakes videos. Recent developments in speech synthesis have shown their potential to produce realistic and natural audio deep fakes, exhibiting real threats to society. Combining synthetic audio content with visual manipulation can significantly make deep fake videos more convincing and increase their harmful impact. Until now, however, these synthesized speeches lack some aspects of voice quality, like expressiveness, roughness, breathiness, stress, emotion, etc., specific to a target identity. The AI research community has made some efforts to produce human-like voice quality with high speaker similarity. This section lists the latest progress in speech synthesis and describes the alarming outcomes in speech synthesis and the potential threat of stealing a voice identity [7] .

Speech synthesis refers to a technology that can generate speech from a given input, i.e., text-to-speech (TTS) or voice conversion (VC). TTS is a decades-old technology that can synthesize the natural-sounding voice of a speaker from a given input text and thus enables a voice to be used for better human-computer interaction. VC is another technique that modifies the audio waveform of a source speaker to make it sound like the target speaker's voice while keeping the linguistic content unchanged. The latest speech synthesis initiatives raise more concerns about the reliability of the speech/audio [7] .

Overall, important developments in speech synthesis have been done using the methods of speech concatenation or parameterization. The concatenative TTS systems are based on separating high-quality recorded speech into small fragments followed by concatenation into a new speech. In recent years, this method has become outdated and unpopular as it is not scalable and consistent. In contrast, parametric models emphasize extracting acoustic features from the given text inputs and converting them into an audio signal using the vocoders. Interesting outcomes of parametric TTS due to improved speech parameterization performance, vocal tract modeling, and the implementation of deep neural networks evidently show the future of artificial speech production. (Fig3.5) shows the principal design of modern TTS methods [7] .

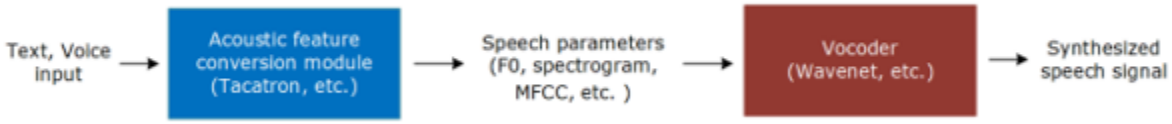

FIGURE 3.5: Workflow diagram of the latest parametric TTS systems

3.4 Conclusion

The privacy of an individual's life is extremely important and constitutes one of the most prominent problems of our time. Therefore, researchers in this field are trying through many efforts and scientific research to solve it. In this section, we have reviewed the most important and prominent techniques that would contribute to the protection of information and prevent its spread in illegal ways.

4.1 Introduction

The widespread use of computer vision technology in society forces the automatic processing of large-scale visual data that often contains personal and sensitive data that should not be revealed publicly. As stipulated by the General Data Protection Regulation (GDPR), an individual's consent to any use of his personal data is necessary. However, if the data does not allow the identification of an individual, companies are free to use this data without consent. This is considered dangerous and unacceptable to all people, especially as the risk of using this data for bad things increases daily.
As an example of this problem, recently, the popular data set for re-identification of the person, the duke MTMC data set was discontinued, for privacy reasons. However, the great need today for technology and surveillance cameras in order to ensure the protection of people and property and to avoid any actions that would threaten national security, made it necessary to search for appropriate solutions to preserve privacy and hide the identity of images without reducing the quality of the image.
To effectively anonymize images, we need a powerful model to replace the original face without destroying the existing data distribution; the output must be a realistic face that fits the given situation.

4.2 Objectif

Through this project we propose a model for anonymizing (or de-identifying) photos and videos by replacing the original face and removing its identifying properties, while retaining the necessary features without destroying the existing data distribution; That is: the output must be a realistic face of the human observer. And the main condition in this solution that we have provided is that it should not, nor in any way, identify the people in the photos. Our proposed approach can be used to mask computer vision data sets while retaining information necessary for tasks such as detection, identification, or tracking.

We rely entirely on Conditional Generative Adversarial Networks (CGAN) to generate anonymous images and videos that look realistic.

In GAN-based methods, the image generation process is usually controlled by a random noise vector file to generate various outputs. In such a random process is not suitable for anonymity purposes, where we need guarantees that the identity has actually changed from input to output. To address this issue, we Suggest a new identifier for identity control.

Our CIAGAN model must guarantee the following important properties that an anonymization system should have:

- **Realistic:** output images must look realistic in order to be used by state-of-the-art detection and recognition systems.
- **New identities:** the generated images must contain only new identities not present in the training set.
- **Anonymization:** the produced output must hide the identity of the person in the original image. Essentially, we are generating a new fake identity out of the input image.
- **Control:** the fake identity of the generated images is governed by a control vector, so we have full control over the real person-fake identity mapping.
- **Temporal consistency:** temporal consistency and pose preservation in videos should be ensured.

If we maintain the above terms, we guarantee the anonymity of photos and videos and protect data privacy. At the same time, our method ensures that the detectors will be able to use the anonymized data.

4.3 The architecture of our solution

In this section, we will talk about our approach to anonymizing photos and videos. The proposed **"Conditional Identity Anonymization Generative Adversarial Network"** (CIAGAN) leverages the power of generative adversarial networks to produce photorealistic images. In order to control and ensure the identity creation process. For the rest of the section, we'll be referring specifically to facial anonymization, although the method is directly applicable to the entire body.

4.3.1 Method overview

The important components of our method are as follows:

4.3.1.1 Conditional generative adversarial network

CGAN is a type of GAN that involves the conditional generation of images by a generator model. GANs rely on a generator that learns to generate new images, and a discriminator that learns to distinguish synthetic images from real images.
In cGANs, a conditional setting is applied, meaning that both the generator and discriminator are conditioned on some sort of auxiliary information (such as class labels or data) from other modalities. As a result, the ideal model can learn multi-modal mapping from inputs to outputs by being fed with different contextual information [8] .

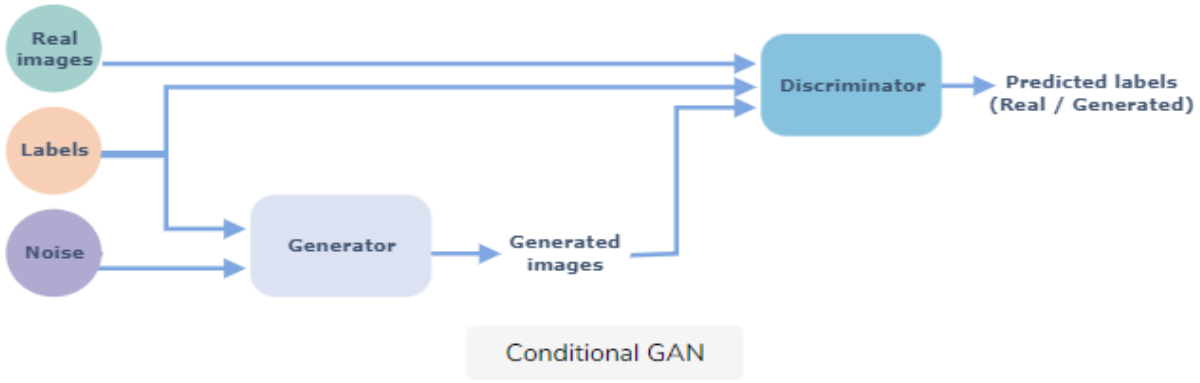

FIGURE 4.1 : Conditional generative adversarial network Standard architecture

We aspire to a great extent to take advantage of the large generating power of the to achieve realistic results because it is important that we can apply standard systems for detection and tracking without losing the accuracy and quality of the images. We achieve pose preservation by conditioning on the landmark representation. We train the conditional GAN in an adversarial game-theoretical way, where the discriminator judges the realism of the images generated by the generator [8] .

4.3.1.2 Pose preservation and temporal consistency:

We have used a landmark-based representation of the input face it ensures pose preservation which is especially useful for, e.g., tracking; and it provides a simple but efficient way to maintain temporal consistency when working with videos [9] .

4.3.1.3 Identity guidance discriminator

We propose a novel module that controls the identifying characteristics that the generator injects to create the new image. The identity discriminator and the generator play a collaboration game where they work together to achieve their common goal of generating realistic anonymized images. We now provide a more detailed description of the three modules of our method [9] .

4.3.2 Conditional generative adversarial networks

4.3.2.1 GAN framework

A generative adversarial network consists of two parts as Generator and Discriminator. The generator learns to generate the probable data which seems from the training data and on the other hand discriminator tries to differentiate the generator's fake data from real data. The discriminator penalizes the generator for implausible results [10] .

The Generator tries to fool the discriminator into thinking that the generated images are real and the discriminator tries to differentiate between real and fake images. Random noise is fed into the Generator that transforms it into a "fake image". The Discriminator is fed from both the training set images and the fake images coming from the generator and it has to tell them apart [10] .

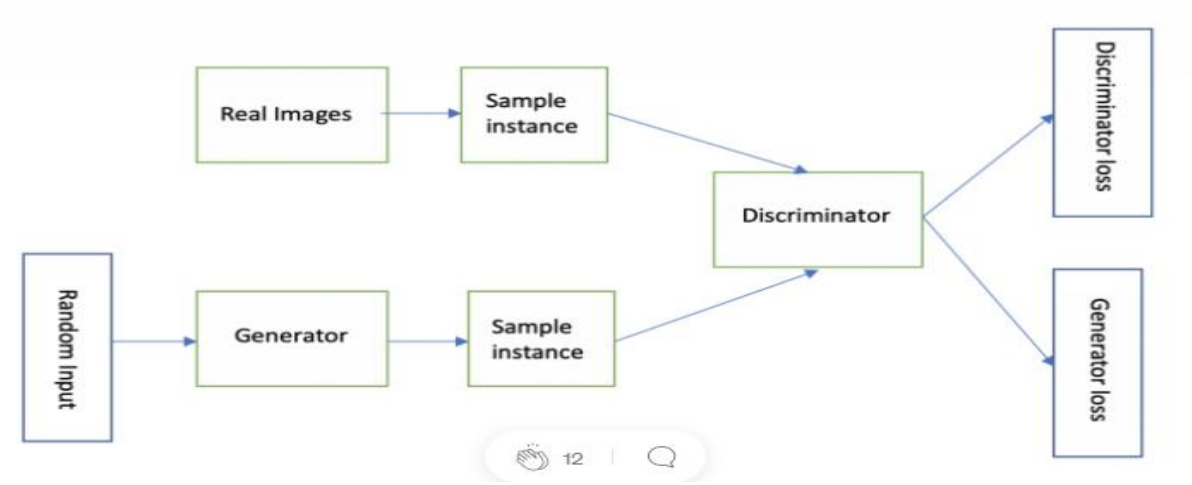

FIGURE 4.2 : Generative adversarial network architecture

It is well-known that GAN training is hard and requires many tricks .In this work, we choose to train CIAGAN with the **LSGAN loss function** . The idea of using least-squares loss function for GAN training is simple yet powerful: the least-squares loss function is able to move the fake samples toward the decision boundary, as it penalizes also samples that are correctly classified but still lie far away from real samples. This is opposed to the cross-entropy loss that mostly penalizes wrongly classified samples. Based on this property, LSGANs are able to generate samples that are closer to real data.

4.3.2.2 CIAGAN Training

In this work we will train our Conditional Identity Anonymization Generative Adversarial Network with the **loss function GAN (LSGAN) :**

4.3.2.2.1 Least Square Adversial Network (LSGAN)

In a normal GAN, the discriminator uses cross-entropy loss function which sometimes leads to vanishing gradient problems. Instead of that LSGAN uses the least-squares loss function for the discriminator. It has the desire to provide a signal to the generator about the fake samples that are far from the discriminator model's decision boundary for classifying them as real or fake. It can be implemented with a minor change to the output layer of the discriminator layer and the adoption of the least-squares, or loss function [11] .

4.3.2.2.2 LSGAN Architecture

- **The Generator:** will take as input high dimensional noise (in the sense that it has higher dimension that the images) and the cropped image [11]. It will join (actually simply concatenate) and process them together.

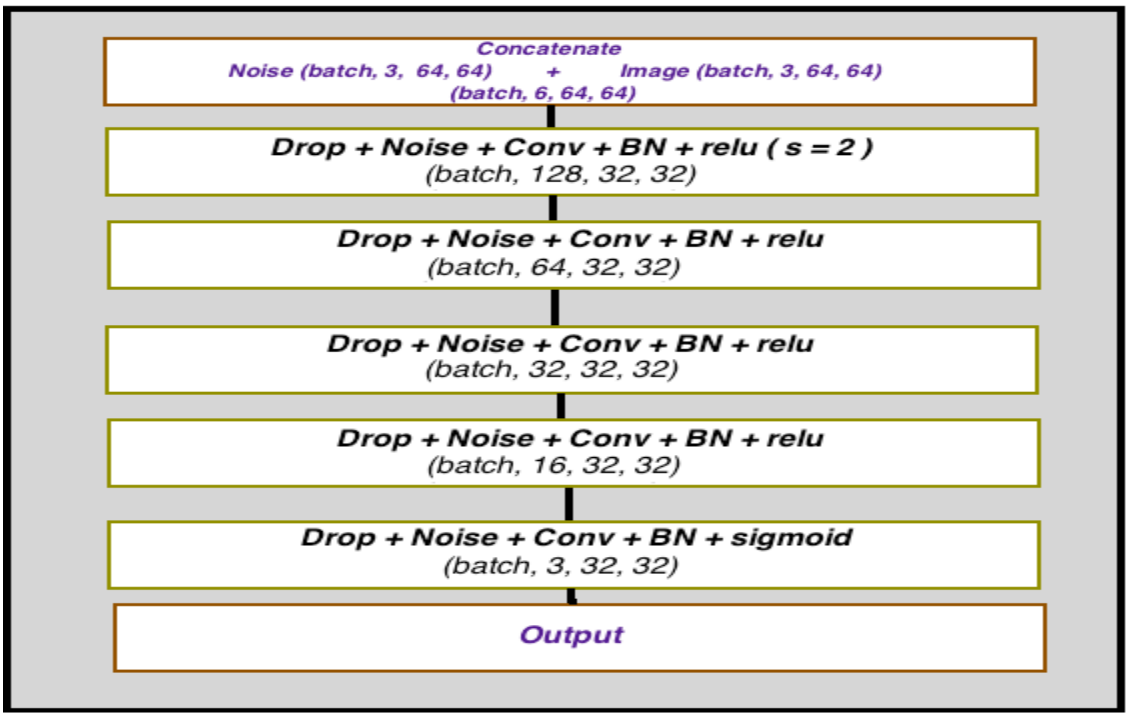

FIGURE 4.3 : The architecture of the Least Square Adversial Network generator

After the concatenation of input noise and cropped images, several convolution layers are stacked and we reinject noise at each layer, while applying a small Dropout (10%). The activation is (Leaky) relu as suggested in the DCGAN paper, except at the last layer where I used sigmoid since the original images are composed of real numbers between 0 and 1 [11] .

- **The Discriminator:** will take as input the full reconstructed image, i.e., the cropped image with either the true center or the generated center. This will also help to avoid border artifacts since the Discriminator will see the whole image and not only the center, thus giving gradient information to the Generator that the whole image should be consistent [11] .

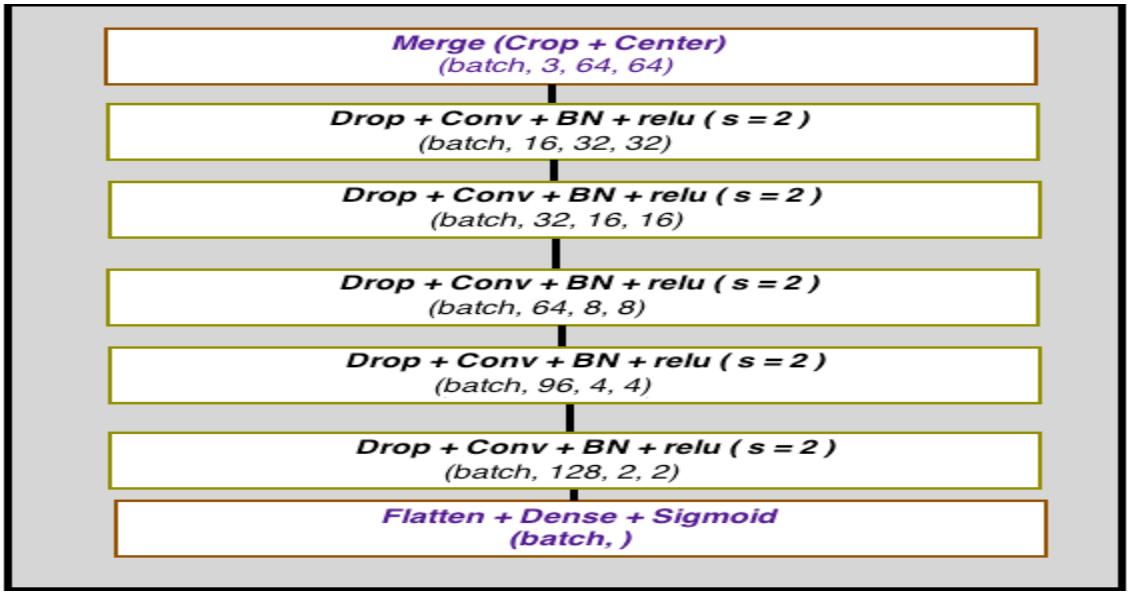

FIGURE 4.4 : The architecture of the Least Square Adversial Network discriminator

4.3.3 Pose preservation and temporal consistency

In order to work on videos, any de-anonymization pipeline must ensure the temporal consistency of generated images on the video sequence that why our input it should be in the form of :

4.3.3.1 Masked background image

We only want to generate the face region of an image and inpainting it to the original image background. This allows our algorithm to focus just on creating faces (and not that of background) . and at the same time guarantees that we do not have background changes that could interfere with the detection or tracking algorithms. To achieve this, we provide the generative model with the masked background image together with the landmark image. The masked background image still contains the forehead region of the head. Once the generator has access to this information, it can learn to match the skin appearance of the generated face to the forehead skin color. This leads to overall better quality of visual results. In cases there are multiple faces in the same image, we detect each face on the image and sequentially apply our anonymization framework.

We make sure to keep the background image because we need to generate a third face that doesn't look overly artificial. That means we will blend the face with the background.

FIGURE 4.5 : Example of a Masked image

4.3.3.2 Partial landmarks image :

1- We do not want the appearance of the input face to leak to the new face. That is why we completely block the face with a mask.

We show the frame of the face in order to express the oriental because it is very important for faces and bodies in order to know what kind of pose, they are in:

- **Mouth for expressions:**

The mouth we also keep it in order to keep expression (if the person was smiling, we want to keep that expression in the output image).

- **Nose and frame for orientation:**

The bridge of the nose to infer the orientation of the face .

- **Free temporal consistency:**

We get a free temporal consistency because landmarks are consistent in time by nature, so if we use this as an input, we will achieve some form of temporal consistency.

2- Face landmark images. Featured image contains a scattered representation of the face with little identity information left, avoid identity leakage.

3- The generator is conditioned on the face shape, which allows us to keep the input in the output. This is Particularly important because we intend to use the generated images and videos as inputs to computer vision algorithms, ensuring that the method will not alter the position of the face or body unknown is very useful. To hide as much identity as possible.

To conceals much as possible and ,at the same time ,preserve mode, instead of using of all 68 landmarks. we use only the silhouette of the face , mouth and bridge of the nose. That allows the grid to choose several types of facial features, such as eye distance or eye shape, at the same time while preserving expressions that depend on the mouth area, e.g. Smiling or laughing [12] .

FIGURE 4.6 : Example Partial landmarks image

4.3.4 Identity Discriminator

We have to generate an output that is not detectable as the initial identity or the celebrity identity. So, we should have an Identity Discriminator which is another neural network that is trained for reidentification or real images.

Its goal is to bring the embedding of the newly generated identity closer to the training ID embedding.

4.3.4.1 Identity Discriminator Architecture

Similarity learning does not include an output layer that assigns class probability. Instead, a similarity model's output layer produces an embedding of the input image into a representative feature space. A similarity learning model has two identical neural networks called Siamese neural networks [13] . These networks are trained with back-propagation to optimize a ranking loss function. Each neural network produces an embedding through back-propagation, and the network's weights are updated to minimize the loss. A similarity learning model is trained to learn a representative feature vector of an object from an image signal. The feature vector is generated so that dis/similar real-world objects are dis/similar in the embedding space. There are a significant number of loss functions adopted in experiments for training similarity learning models, namely contrastive loss, triplet loss, and Proxy-NCA [13] .

A Siamese network consists of two identical neural networks, each taking one of the two input images. The last layers of the two networks are then fed to a contrastive loss function , which calculates the similarity between the two images. I have made an illustration to help explain this architecture [13] .

The objective of the Siamese architecture is not to classify input images, but to differentiate between them. So, a classification loss function (such as cross entropy) would not be the best fit. Instead, this architecture is better suited to use a contrastive function. Intuitively, this function just evaluates how well the network is distinguishing a given pair of images [13] .

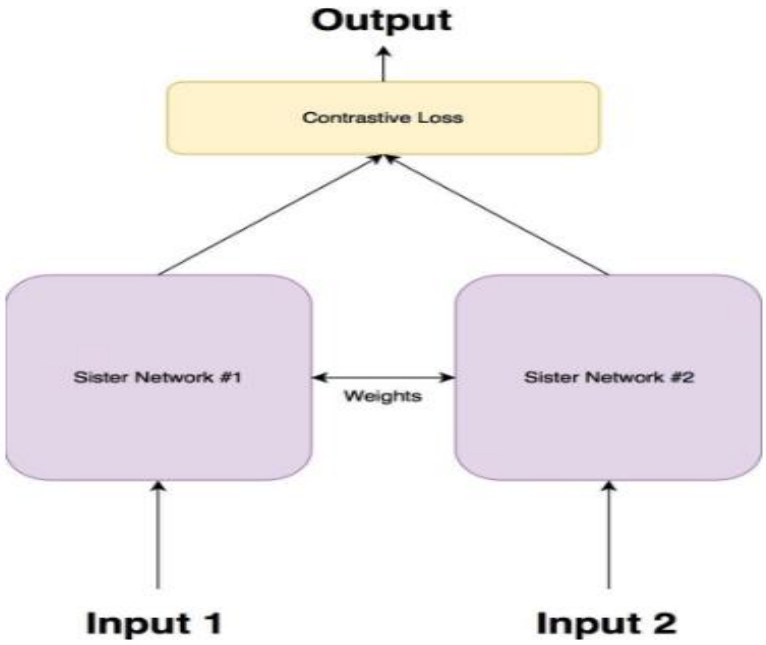

FIGURE 4.7 : Siamese network architecture

4.3.4.2 Identity discriminator training:

Identity discriminator is designed as a siamese neural network pretrained using Proxy-NCA loss. Pretraining is done using real images, where the discriminator is trained to bring together features coming from the images that belong to the same identity. During the GAN training, we finetune the siamese network using contrastive loss. During this fine-tuning step, we allow the siamese network to bring together the ID representation of the fake images and the real images. The identity discriminator and the generator are trained jointly in a collaborative manner. The identity discriminator's goal is to provide a guidance signal to the generator to guide it towards creating images whose representational features are similar to those of a particular identity [13].

4.3.5 System work steps

4.3.5.1 The input of our model

The initial input that we're feeding to our neural network is going to be the original detected face. There is no guarantee here that we're fully anonymizing the image because if there is a face that is not detected as a face, then the generator will not anonymize it.

We don't want to present to our neural network the whole picture like we said before but only part of it, because if we present the full picture, there is a strong chance that some identity will leak to the generated anonymized image version.

We'll cover the face and let the hair and the forehead pass through the **generator**. In order to give the **generator** an idea about the pose of the face, we are going to also pass the face through facial landmarks so that it can extract some facial elements, or more precisely, where are the facial elements such as the bridge of the nose, the corner of the mouth, and the shape of the face [9] .

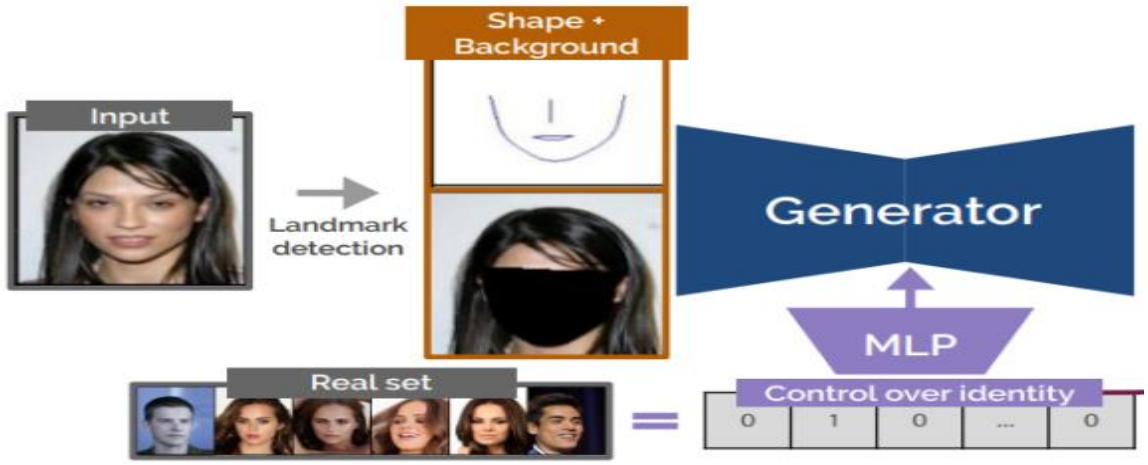

FIGURE 4.8 : The input to the generator

4.3.5.1.1 Generator Architecture

Encoder-decoder neural network architecture also known as U-Net where VGG11 neural network without fully connected layers as its encoder. Each blue rectangular block represents a multi-channel features map passing through a series of transformations. The height of the rod shows a relative map size (in pixels), while their widths are proportional to the number of channels (the number is explicitly subscribed to the corresponding rod). The number of channels increases stage by stage on the left part while decrease stage by stage on the right decoding part [14] .

The arrows on top show transfer of information from each encoding layer and concatenating it to a corresponding decoding layer [14] .

- The encoder is the first half in the architecture diagram (Figure 4.9). It usually is a pre-trained classification network like VGG/ResNet where you apply convolution blocks followed by a maxpool downsampling to encode the input image into feature representations at multiple different levels .
- The decoder is the second half of the architecture. The goal is to semantically project the discriminative features (lower resolution) learnt by the encoder onto the pixel space (higher resolution) to get a dense classification. The decoder consists of upsampling and concatenation followed by regular convolution operations.
- Bottlenecks in Neural Networks are a way to force the model to learn a compression of the input data. The idea is that this compressed view should only contain the "useful" information to be able to reconstruct the input (or segmentation map) [14] .

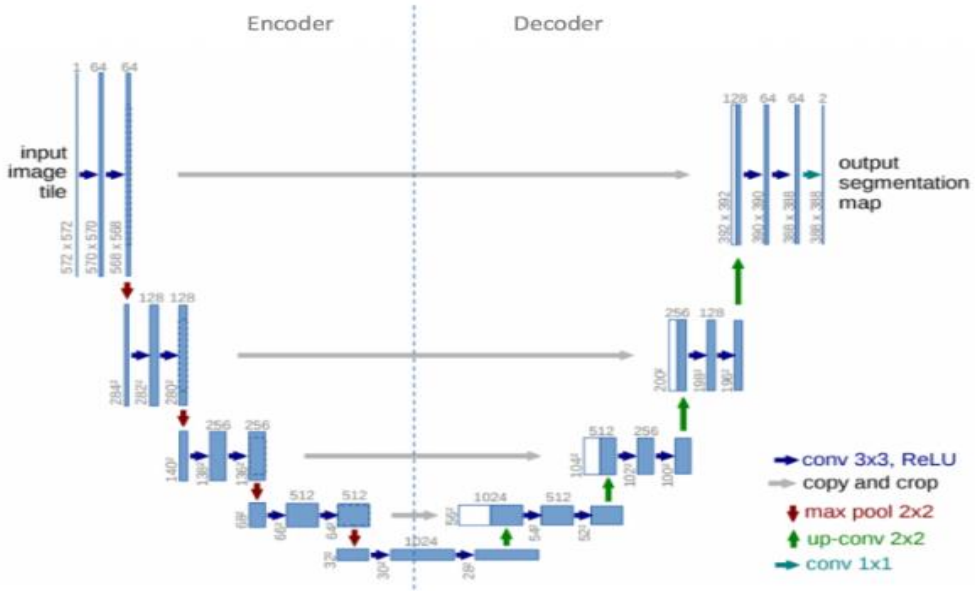

FIGURE 4.9 : Encoder-decoder neural network architecture

U-net architecture. Blue boxes represent multi-channel feature maps, while boxes represent copied feature maps. The arrows of different colors represent different operations.

4.3.5.2 The process

Losses1: GAN loss

- First, we are going to use GAN to generate realistically looking images.
- Our Gan generate an output and there is a discriminator that is going to judge whether that output looks like reel face or not.

But without further losses the network will over fit and the network will simply do a reconstruction to return a very similar image to our input image.

The entire variability in image generation is provided by the landmark input, the network quickly overfits on the training set, effectively doing only image reconstruction. By doing so, it generates faces that are very similar to those in the training dataset, forfeiting the final anonymization goal.

To solve this problem, we introduce a novel identity guidance discriminator. More precisely, for every given real image, we randomly choose
the desired identity of its corresponding generated image.This identity – represented as on one hot-vector – is given as input to a transposed convolutional neural network.

Losses2: ID loss

- Second, the Identity loss compare the output image with the training set and it looks as if this identity can be recovered or not therefore whether the identity of the output is the same of the input given all the training set that we have .

4.3.5.2.1 Discriminator Architecture

For the discriminator we use a transposed convolutional neural network containing a fully connected layer followed by multiple transposed convolutional layers.

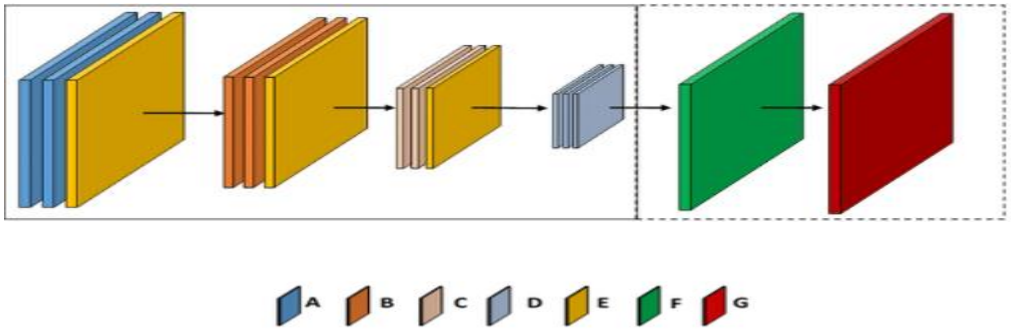

FIGURE 4.10 : CIAGAN Discriminator Architecture
The fully-convolutional network (FCN). A, B, C and D are convolutional layers; E is a pooling layer; F is a transposed convolutional layer or unpooling layer ; G is a loss layer.

4.3.5.2.2 How identity discriminator works:

Identity guidance

Input

- Identity discriminator: is going to have an identity guidance so the input to the bottleneck is going to be a one hot vector encoding of a random ID of the training set.
- Our training set. consists of several celebrity images and we have a fixed set of ID's that we can use in order to anonymize an image.
- We take one of these identities at random and we pass this identity control which is this one hot factor encoding through an MLP and we obtain a representation can be concatenated with the bottleneck over the generator and this is going to be a new identity information.
- The decoder has part of the embedding that comes from the generator and has the characteristics of the old identity and it also has the representation that comes from the MLP which contains the new identity information .
- The decoder uses this encoded information of the initial ID and the new ID of the random training and it's going to mix them.

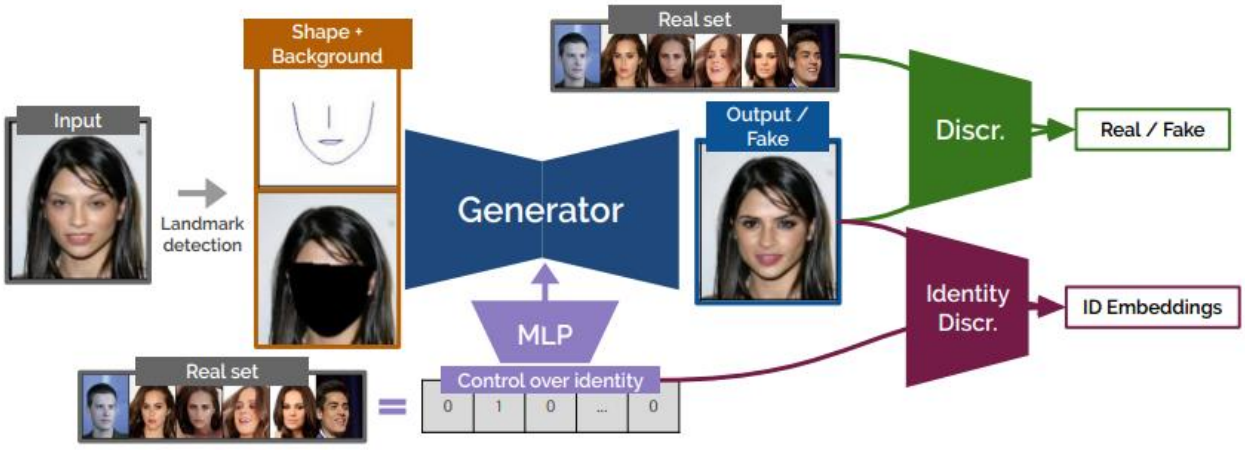

FIGURE 4.11 : The global architecture of our solution (CIAGAN architecture)

4.3.5.3 The Output of our System

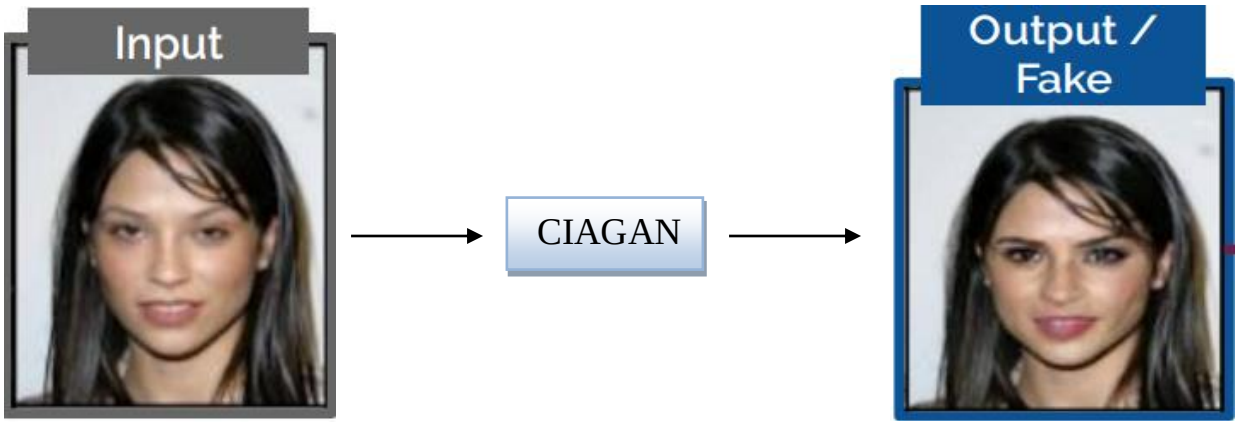

FIGURE 4.12 : The input and the output of our CIAGAN

4.4 The advantages of CIAGAN

1- Scientists have done many types of research and studies on how to remove sensitive information from the image, including the face. Among the most prominent techniques used were **:**

- ✓ **Face blurring:** is a computer vision method used to anonymize faces in images and video. An example of face blurring and anonymization can be seen in Figure 1 above , the face is blurred, and the identity of the person is indiscernible.

FIGURE 4.13 : Face blurring Example

- ✓ **Face pixelization:** is the term used in computer graphics to describe blurry sections or fuzziness in an image due to visibility of single-colored square display elements or individual pixels.

FIGURE 4.14 : Face pixelization Example

- ✓ **Edge detection:** is an image processing technique for finding the boundaries of objects within images. It works by detecting discontinuities in brightness. Edge detection is used for image segmentation and data extraction in areas such as image processing, computer vision, and machine vision. It is used to hide some privacy-sensitive information about the image.

FIGURE 4.15 : Edge detection Example

- ✓ **Face silhouette:** is the image of a person, animal, object or scene represented as a solid shape of a single colour, usually black, with its edges matching the outline of the subject. The interior of a silhouette is featureless, and the silhouette is usually presented on a light background, usually white, or none at all. It is also used to hide the identity of images in order not to identify their owners.

FIGURE 4.16 : Face silhouette Example

However, these techniques are very simple and do not guarantee 100% the removal of all privacy-sensitive information. Often, these methods make faces undetectable and therefore unusable for vision tasks such as detection and tracking.

2- Our project relies heavily on creating faces that look real, and it is difficult to distinguish that they are fake faces that do not belong to any human being and were made only by GAN.

3- Our method, unlike many other techniques, provides completely different images from the source image, but also when changing the control vector, our network is able to provide more diverse pictures.

4- Our CIAGAN works on only “the landmark images”, but not the whole faces, which is why it does a better job than it would have if it had used the full face.

5- The source image can be anonymized based on different identities (e.g., long or short hair, dark or light skin.....).

6- Neither the dataset image nor the source image features can be distinguished.

4.5 The disadvantages of CIAGAN

1- One of the weaknesses of our model is that it relies heavily on defining the face and its characteristics completely so that it can be anonymized. It means that if the faces are not clear, or not opposite to the camera, or if they make extreme movements, they cannot be detected and therefore cannot be anonymized.

2- In a few cases, when the video contains many faces at the same time, there may be an imbalance in the faces and difficulty in centering.

3- Our method proves its ability to generate objectively good images for a diversity of backgrounds and poses. However, it still struggles in several challenging scenarios. These issues can impact the generated image quality, but, by design, our model ensures the removal of all privacy-sensitive information from the face.

FIGURE 4.17 : Failure Cases of CIAGAN – example 1

4- Handling non-traditional poses can cause our model to generate corrupted faces. We use a sparse pose estimation to describe the face pose, but there is no limitation in our architecture to include a dense pose estimation. A denser pose estimation would, most likely, improve the performance of our model in cases of irregular poses. However, this would set restrictions on the pose estimator and restrict the practical use case of our method.

FIGURE 4.18 : Failure Cases of CIAGAN – example 2

4.6 Conclusion

In this part, we have talked about the architecture of our model, its components, characteristics and how each part works, we present also in this section the advantages and the disadvantages of our solution. We've explained what our model input is, and what process is going on inside. We've also explained how each of the elements we saw earlier contributes to hiding the identity of the images perfectly so that no one can identify them.

5.1 Introduction

In this chapter, we will start with an overview of the tools used to realize our system and will talk about its evaluation. We used the Evaluation Metrics as a recall and the Freckled Inception Distance Score, and of course, we are going to express the results obtained after tests in the form of tables.

5.2 The Development environment

We developed this application with the Python language and we used Google Colab as a development tool.

5.2.1 Python Language

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development. Also, since it is extensible and portable, Python can be used to perform cross languages tasks. The adaptability of Python makes it easy for data scientists and developers to train machine learning models. Fast code tests: Python provides a lot of code review and test tools.

5.2.2 Google Golab

Colaboratory, or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.

5.2.3 Hardware environment

Processor: Intel(R) Celeron(R) CPU N3060 @ 1.60GHz 1.60 GHz
Installed RAM: 4.00 GB
System type : 64-bit operating system, x64-based processor

5.3 Python libraries used in the project

5.3.1 SSH: Single Stage Headless Face Detector

Face detection is a crucial step in various problems in volving verification, identification, expression analysis, etc. From the Viola-Jones detector to recent work by Hu et al., the performance of face detectors has been improved dramatically. However, detecting small faces is still considered a challenging task. The recent introduction of the WIDER face dataset, containing a large number of small faces, exposed the performance gap between humans and current face detectors. The problem becomes more challenging when the speed and memory efficiency of the detectors are taken into account. The best-performing face detectors are usually slow and have high memory oot-prints) partly due to the huge number of parameters as well as the way robustness to scale or incorporation of context are addressed [15] .

We introduced a single-point headless (SSH) face detector. Unlike the proposed two-stage classification detector, SSH detects faces directly from the early convolutional layer in the classification network in a single-stage manner. SSH has no headers. In other words, it can achieve the most advanced results while removing the "head" of its underlying classification network—that is, the fully connected layer containing a large number of parameters in VGG-16 (**Note**: In fact, it is the removal of the three VGG A fully connected layers). In addition, SSH does not rely on image pyramids to detect faces at various scales but is designed to be scaled unchanged. We also detect faces with different scales from different layers in a single forward pass of the network. These attributes make SSH fast and lightweight [15] .

FIGURE 5.1 : Detected faces by SSH face detector

In our method SSH is able to detect various face sizes in a single CNN feed-forward pass and without employing an image pyramid in ~ 0.1 second for an image with size 800×1200 on a GPU.

5.3.2 Histogram of Oriented Gradients (HOG)

The Histogram of Oriented Gradients, also known as HOG, is a feature descriptor like the Canny Edge Detector, SIFT (Scale Invariant, and Feature Transform). It is used in computer vision and image processing for object detection. The technique counts occurrences of gradient orientation in the localized portion of an image. This method is quite similar to Edge Orientation Histograms and Scale Invariant Feature Transformation (SIFT). The HOG descriptor focuses on the structure or shape of an object. It is better than any edge descriptor as it uses magnitude as well as the angle of the gradient to compute the features. For the regions of the image, it generates histograms using the magnitude and orientation of the gradient [16] .

We know the basic principle of Histogram of Oriented Gradients. We will be moving into how we calculate the histograms and how these feature vectors, that are obtained from the HOG descriptor, are used by the classifier such as SVM to detect the concerned object.

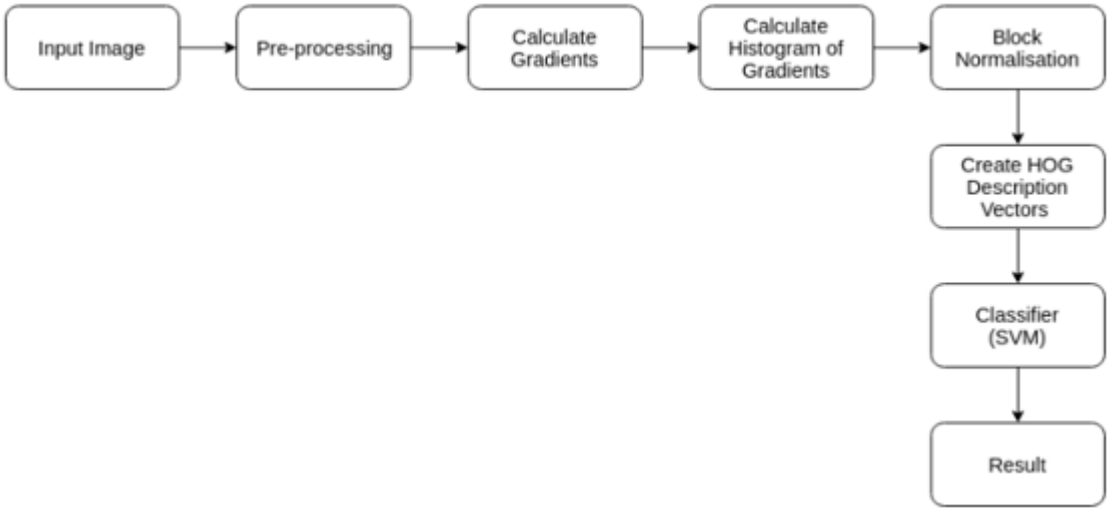

FIGURE 5.2 : Steps for Object Detection with HOG

5.3.3 Shape_predictor_68_face_landmarks

Shape_predictor() is a tool that takes in an image region containing some object and outputs a set of point locations that define the pose of the object.
a landmark's facial detector with pre-trained models, the dlib is used to estimate the location of 68 coordinates (x, y) that map the facial points on a person's face like image below. These points are identified from the pre-trained model where the iBUG300-W dataset was used.

FIGURE 5.3 : Shape_predictor_68_face_landmarks

5.3.4 Torchvision:

It is a library for Computer Vision that goes hand in hand with PyTorch. It has utilities for efficient Image and Video transformations, some commonly used pre-trained models, and some datasets (torchvision does not come bundled with PyTorch, you will have to install it separately).

Installation

- ✓ Conda install torchvision -c pytorch (if you are using conda)
- ✓ Pip install torchvision (for the pip installation)

Advantages :

- Since it is an accompaniment to PyTorch, it automatically comes with the GPU support**.**
- Its development philosophy is to be simple in implementation (e.g: without an extensive argument set for its functions) . The developers have kept it separately from PyTorch to keep it lean and lightweight.
- (Comes with sample data-set (CelebA etc) , some commonly used pre-trained models (ResNet18, maskRCNN_resnet50 etc) and even sample starter codes for some of the typical AI/ Machine Learning Problems (Image Classification, Semantic Segmentation , Keypoint detection etc) — all inbuilt into its library
- It is developed and maintained by the Facebook AI team, and supported by the python community.

5.3.4 OpenCV :

It is an open-source library that is very useful for computer vision applications such as video analysis, CCTV footage analysis, and image analysis. OpenCV is written in C++ and has more than 2,500 optimized algorithms. When we create applications for computer vision that we don't want to build from scratch we can use this library to start focusing on real-world problems. Many companies are using this library today such as Google, Amazon, Microsoft, and Toyota. Many researchers and developers contribute. We can easily install it in any OS like Windows, Ubuntu and macOS.

OpenCV offers method to read video from camera, file or any video stream and write it to a file. We can get video frame by frame and process and save it to output file. In this tutorial we will read video from camera or file and apply some operations on it and then write to a new video file.

To read our video we use :

Video Capture method. It can be used with cameras connected to system, video files or network stream. First, we read and get its details.

```
import cv2
# Read video from file
   cap = cv2.VideoCapture('VIDEO_PATH')
# Read from webcam
   cap = cv2.VideoCapture(0) # 0 for default cam
# Read from network
   cap = cv2.VideoCapture('IP_HERE')
```

5.3.5 Numpy

It is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. It also has functions for working in the domains of linear algebra, the Fourier transform, and matrices.

NumPy was created in 2005 by Travis Oliphant. It is an open-source project, and you can use it freely.

We use the "np.pad function" for padding.

Padding: is the space between an image or cell contents and its outside border**.** In the image below, the padding is the yellow area around the content. In this case, padding goes completely around the contents: top, bottom, right, and left sides.

5.3.6 PyTorch

It is an open-source library used in machine learning library developed using Torch library for python program. It is developed by Facebook's AI Research lab and released in January 2016 as a free and open-source library mainly used in computer vision, deep learning, and natural language processing applications. Programmers can build a complex neural network with ease using PyTorch as it has a core data structure, Tensor, and multi-dimensional array like Numpy arrays. PyTorch use is increasing in current industries and the research community as it is flexible, faster, and easy to get the project up and running, due to which PyTorch is one of the top deep learning tools.

Why do we need PyTorch?

The PyTorch framework can be seen as the future of the deep learning framework. Many deep learning frameworks are getting introduced, and the most preferred frameworks are TensorFlow and PyTorch, but among all, PyTorch is emerging as a winner due to its flexibility and computation power. For machine learning and Artificial Intelligence enthusiast, PyTorch is easy to learn and will be very useful to build models.

PyTorch Components

Let's look into the five major components of PyTorch:

FIGURE 5.4 : PyTorch Components

In our project we use just Tensors library:

Tensors: Tensors are the multi-Dimensional array similar to the Numpy array, and Tensors are available in Torch as a torch. IntTensor, torch.FloatTensor, torch.CharTen etc. It is used a convolution neural network to develop image classification, object detection, and generative application. Using PyTorch, we can process images and videos to develop a highly accurate and precise computer vision model.

5.3.7 Pillow

This library provides support for various image formats including the popular JPEG and PNG formats. Another reason you would consider using Pillow is the fact that it is quite easy to use and very popular with Pythonistas. The package is a common tool in the arsenal of most data scientists who work with images.

It also provides various image processing methods as we will see in this piece. These techniques are very useful especially in augmenting training data for computer vision problems.

5.4 The Dataset used in the project

We perform experiments on 2 public datasets:

5.4.1 CelebFaces Attributes Dataset (CelebA)

It is a large-scale face attributes dataset with more than **200K** celebrity images, each with **40** attribute annotations. The images in this dataset cover large pose variations and background clutter. CelebA has large diversities, large quantities, and rich annotations, including

- **10,177** number of **identities**,
- **202,599** number of **face images**, and
- **5 landmark locations**, **40 binary attributes** annotations per image.

The dataset can be employed as the training and test sets for the following computer vision tasks: face attribute recognition, face recognition, face detection, landmark (or facial part) localization, and face editing & synthesis [17] .

Sample Images

FIGURE 5.5 : CelebA dataset sample images

5.4.2 Labeled Faces in the Wild (LFW) :

Labeled Faces in the Wild (LFW) is a database of face photographs designed for studying the problem of unconstrained face recognition. This database was created and maintained by researchers at the University of Massachusetts, Amherst (specific references are in Acknowledgments section). 13,233 images of 5,749 people were detected and centered by the Viola Jones face detector and collected from the web. 1,680 of the people pictured have two or more distinct photos in the dataset. The original database contains four different sets of LFW images and also three different types of "aligned" images. According to the researchers, deep-funneled images produced superior results for most face verification algorithms compared to the other image types. Hence, the dataset uploaded here is the deep-funneled version [18] .

The dataset consists of 6, 000 pair images, split in 10 different splits, where half of the pairs contains images of same identity, and the remaining pairs consist of images that have different identities.

- ✓ Face verification and other forms of face recognition are very different problems. For example, it is very difficult to extrapolate from performance on verification to performance on 1:N recognition.
- ✓ Many groups are not well represented in LFW. For example, there are very few children, no babies, very few people over the age of 80, and a relatively small proportion of women. In addition, many ethnicities have very minor representation or none at all.
- ✓ While theoretically LFW could be used to assess performance for certain subgroups, the database was not designed to have enough data for strong statistical conclusions about subgroups. Simply put, LFW is not large enough to provide evidence that a particular piece of software has been thoroughly tested.
- ✓ Additional conditions, such as poor lighting, extreme pose, strong occlusions, low resolution, and other important factors do not constitute a major part of LFW. These are important areas of evaluation, especially for algorithms designed to recognize images "in the wild".

For all of these reasons, we would like to emphasize that LFW was published to help the research community make advances in face verification, not to provide a thorough vetting of commercial algorithms before deployment.

Sample Images

FIGURE 5.6 : Labeled Faces in the Wild dataset sample images

5.5 Evaluation Measures

5.5.1 Evaluation Metrics

They are used to measure the quality of the statistical or machine learning model. Evaluating machine learning models or algorithms is essential for any project. There are many different types of evaluation metrics available to test a model.

Different types of metrics are used like **"**Recall" and "**The Frechet Inception Distance Score"** and **"Inception Score"** [19] .

5.5.1.1 Recall

Recall-Evaluation method is:

- defined as the fraction of positive cases that are correctly identified.
- defined as the percentage of true positive cases versus all the cases where the prediction is true.
- defined as the percentage of correct predictions out of all the observations.

In an imbalanced classification problem with two classes, recall is calculated as the number of true positives divided by the total number of true positives and false negatives [20] .

- **Recall = TruePositives / (TruePositives + FalseNegatives)**

The result is a value between 0.0 for no recall and 1.0 for full or perfect recall.

Recall score is used to measure the model performance in terms of measuring the count of true positives in a correct manner out of all the actual positive values. Precision-Recall score is a useful measure of success of prediction when the classes are very imbalanced [20] .

In our project we use the standard Recall@1 evaluation metric for re-identification:

It measures the ratio of samples whose nearest neighbor is from the same class. The metric can take values from 0 to 100 with 0 showing perfect deidentification rate and 100 showing perfect identification rate. Note that in a balanced dataset, a random classifier will produce (on average) a Recall@1 of 1/|C| where C is the number of classes.

5.5.1.2 The Frechet Inception Distance Score

We evaluate the visual quality of the generated images quantitatively using the Frechet Inception Distance (FID) , a metric that compares the statistics of generated samples to those of real samples. The lower the FID, the better, corresponding to more similar real and generated samples [21] .

The Frechet Inception Distance score, or FID for short, is a metric that calculates the distance between feature vectors calculated for real and generated images (a metric used to assess the quality of images created by a generative model) [21] .

The score summarizes how similar the two groups are in terms of statistics on computer vision features of the raw images calculated using the inception v3 model used for image classification. Lower scores indicate the two groups of images are more similar, or have more similar statistics, with a perfect score being 0.0 indicating that the two groups of images are identical [21] .

The FID score is used to evaluate the quality of images generated by generative adversarial networks, and lower scores have been shown to correlate well with higher quality images [21] .

The FID score is then calculated using the following equation taken from the paper:

$$d^2 = ||mu\_1 - mu\_2||^2 + Tr(C\_1 + C\_2 - 2*sqrt(C\_1*C\_2))$$

- ✓ The score is referred to as *d^2*, showing that it is a distance and has squared units.
- ✓ The “mu_1” and “mu_2” refer to the feature-wise mean of the real and generated images, e.g., 2,048 element vectors where each element is the mean feature observed across the images.
- ✓ The C1 and C2 are the covariance matrix for the real and generated feature vectors, often referred to as sigma.
- ✓ The ||mu_1 – mu_2||^2 refers to the sum squared difference between the two mean vectors.
- ✓ *Tr* refers to the trace linear algebra operation, e.g., the sum of the elements along the main diagonal of the square matrix.
- ✓ The sqrt is the square root of the square matrix, given as the product between the two covariance matrices.

The use of activations from the Inception v3 model to summarize each image gives the score its name of “Frechet Inception Distance.”

A lower FID indicates better-quality images; conversely, a higher score indicates a lower-quality image and the relationship may be linear [22] .

The authors of the score show that lower FID scores correlate with better-quality images when systematic distortions were applied such as the addition of random noise and blur.

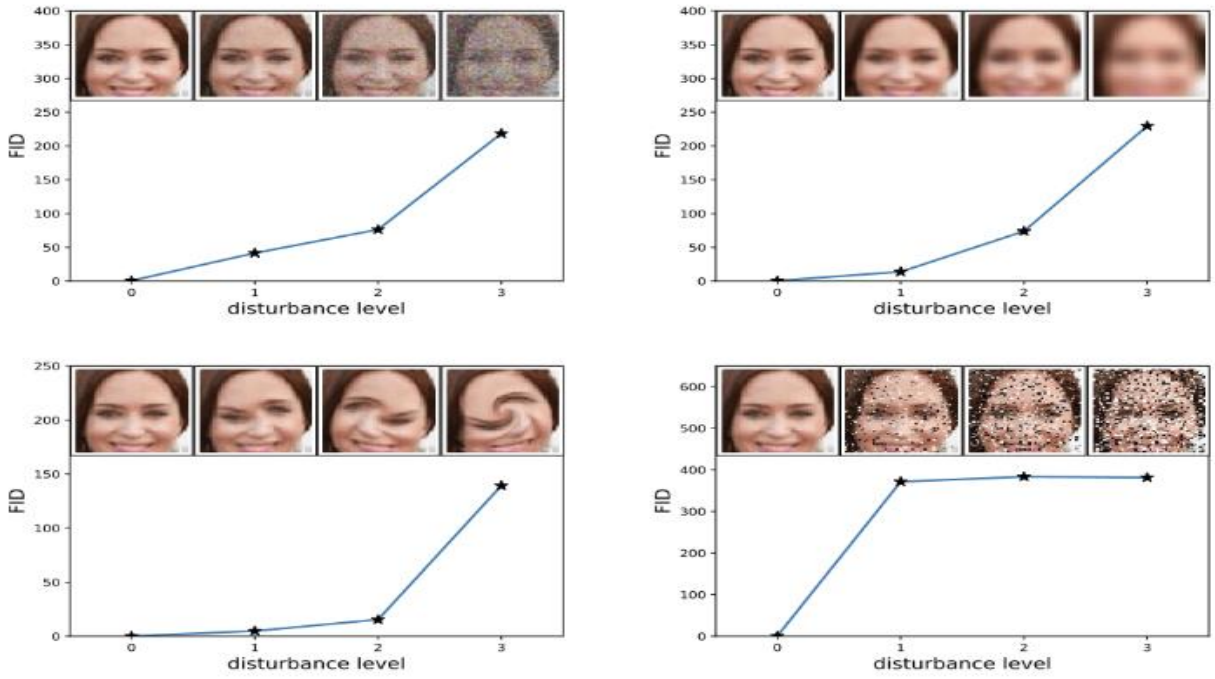

FIGURE 5.7 : Example of How Increased Distortion of an Image Correlates with High FID Score.

In Table 5.1, we show several variants of our model. Siamese indicates our full model, with a siamese identity discriminator, and using landmarks as input. Classification indicates replacing the siamese identity discriminator by a classification network. As we can see, the results of the detection drop more than 35 percentage points (pp). We also show what happens if instead of landmarks, entire face images are provided as input. In these cases, the detection rate drops 1.6pp, and the FID score increases, showing that the faces are both more difficult to be detected, and have lower visual quality.

Models	Detection (↑)	Recall@1 (↓)	FID (↓)
Siamese	99.9	1.3	2.1
Classification	64.6	0.4	63.2
Faces	98.3	1.1	6.5

Table 5.1: Ablation study of our model

In (Table 5.1) First row presents the result of our model, second row shows the result of the model where the siamese identity guidance network is replaced with a classification network, while the third row shows the result of the model where the generator accepts full face images instead of landmarks.

5.5.1.3 Inception Score

The Inception Score, or IS for short, is an objective metric for evaluating the quality of generated images, specifically synthetic images output by generative adversarial network models.

Calculating the Inception Score

We can construct an estimator of the Inception Score from samples x (i) by first constructing an empirical marginal class distribution,

$$\hat{p}(y) = \frac{1}{N}\sum_{i=1}^{N} p(y|\mathbf{x}^{(i)}),$$

where N is the number of sample images taken from the model. Then an approximation to the expected KL divergence can be computed by

$$\mathrm{IS}(G) \approx \exp(\frac{1}{N}\sum_{i=1}^{N} D_{KL}(p(y|\mathbf{x}^{(i)} \parallel \hat{p}(y))).$$

Inception Score on CIFAR-10 We train LSGANs and DCGANs with the same network architecture on CIFAR-10 and use the models to randomly generate 50, 000 images for calculating the inception scores. The evaluated inception scores of LSGANs and DCGANs are shown in (Table5.2). As we observe that the inception scores vary for different trained models, the reported inception scores in (Table5.2) are averaged over 10 different trained models for both LSGANs and DCGANs. For this quantitative evaluation of inception score, LSGANs show comparable performance to DCGANs.

Method	Inception Score
DCGAN	6.22
LSGAN (ours)	**6.47**

Table 5.2: Inception scores on CIFAR-10

We use FaceNet identification model, pre-trained on two public datasets: VGGFace2 and CASIA-Webface. The main evaluation metric is the true acceptance rate: the ratio of true positives for a maximum 0.001 ratio of false positives. We present the results in Table 5.3.

The network evaluated in real faces reaches a score of almost 0.99, nearly perfect identification achieves an impressive anonymization performance by obtaining a score of less than 0.04 using the networks trained in both datasets. CIAGAN improves on this result and lowers the identification rate to 0.034 VGCFace2 and 0.019 CASIA thus improving anonymization. On average, CIAGAN shows a 10.5% better de-identification rate on the first dataset, and a 45.7% better de-identification rate on the second dataset, while keeping a high detection rate of 99.13%. The average performance of 2.65% true positive rate shows that even a near flawless system would completely fail to find the true identities in our CIAGAN-processed data, showing the strength of our method in achieving image anonymization.

VGGFace2: provides annotation to enable evaluation on two scenarios: face matching across different poses, and face matching across different ages.

De-ID method	**VGGFace2 (↓)**	**CASIA (↓)**
Original	0.986 ±0.010	0.965 ±0.016
Gafni et al. [7]	0.038 ±0.015	0.035 ±0.011
Ours	0.034 ± 0.016	0.019 ± 0.008

Table 5.3 : Comparisons with SOTA in LWF dataset. Lower (↓) identification rates imply better anonymization.

5.6 Discussions and comments about the results

5.6.1 Detection and Recognition

To evaluate the performance of the detectors, we use the percentage of detected faces. We evaluate two important capabilities that an anonymization method should have:

- ✓ high detection rate
- ✓ low identification rate.

In our method we do not want a trained system to be able to find the identity of the new generated face, but at the same time, we still want a face detector to have a high detection rate to this face.

In Table 5.4 , we show the detection and identification results of our method compared to the other methods on the CelebA dataset .

The detection rate of classical HOG and deep learning-based SSH detectors in our anonymized images is at almost 100%. Blurring methods have a much lower detection rate in the images, while faces in the pixelized images are not detectable at all.

FaceNet: is a face recognition pipeline that learns mapping from faces to a position in a multidimensional space where the distance between points directly correspond to a measure of face similarity.

Models	Detection (↑)		Identification (↓)	
	Dlib	SSH	PNCA	FaceNet
Original	100	100	70.7	65.1
Pixelization 16 by 16	0.0	0.0	0.3	0.3
Pixelization 8 by 8	0.0	0.0	0.4	0.3
Blur 9 by 9	90.6	38.6	16.9	57.2
Blur 17 by 17	68.4	0.3	1.9	0.5
Ours	99.9	98.7	1.3	1.0

Table 5.4 : Results of common existing detection and recognition pre-trained methods. Lower (↓) results imply a better anonymization. Upper (↑) results imply a better detection.

- We generated the landmarks and the masks using the Dlib-ml library.
- We train our network on 128 × 128 resolution images and use an encoder-decoder U-Net architecture for the generator.
- The identity vector is parametrized by a transposed convolutional neural network containing a fully connected layer followed by multiple transposed convolutional layers.
- The features coming from the landmark and the identity branches are concatenated in the generator's bottleneck.
- For the discriminator, we use a standard convolutional neural network that has the same architecture as the identity guidance network.
- Our model generates a realistic face.
- To evaluate our project, we relied on evaluation metrics.
- Our project still needs some additions, but it has proven very effective compared to other projects that dealt with the same problem.
- Our Model is not able to find the identity of the new created face. The percentage of finding the face does not exceed 1.0% with FaceNet and 1.3% with PNCA, but at the same time, it can detect the face with a rate of 99.9% with Dlib and 38.6 with SSH .We can consider these results to be excellent, unlike other techniques
- Except for one case, which is when there are non-traditional poses so our model can generate corrupted faces.
- If there are many faces in different poses in the same video, this can cause the fabricated faces to be misaligned in the anonymized video.

5.7 Comparison

We will compare here the de-identification (anonymization) power of our model and the percentage of image detection with other methods such as Live face deidentification in video model.

5.7.1 Qualitative comparison

5.7.1.1 Live face deidentification in video

It is a live anonymization video based on a different method than the one we used on this project.

The image in the first column is the source image. In the first row, we show the generated images from the framework of live face deidentification in video, while in the second image, we show the generated images from CIAGAN (our work).

We qualitatively compare our results with those of the live anonymization video. We see that our method not only provides images which are more dissimilar to the source image, but by changing the control vector, our network is able to provide much more diverse images than in **live anonymization video** we can see that the features of the image source are very clear in the anonymized image and did not differ much, they can be distinguished simply.

FIGURE 5.8: Qualitative comparison with "Live face deidentification in video"

5.7.1.2 Visual quality of the results

FIGURE 5.9: Qualitative comparison with « Live face deidentification in video) on temporal consistency

From left to right: original frames; faces generated by our model using faces as input; faces generated by our model using landmarks as input, faces generated by live face deidentification in video.

We demonstrate the temporal consistency of our method and compare the results with those of face deidentification in video .

We see that the pose is preserved in all cases, yielding excellent temporal consistency. At the same time, we see that the CIAGAN version which works with landmarks produces better looking images than the version trained on full faces.

5.7.1.2 Face swapping

We already introduced a novel identity guidance network that guides the generator to produce images with similar features to those of a given identity.

One might argue that by doing so, the generator is learning to only do face swapping, replacing the face of the chosen identity with the landmarks of the source image. We show that this is not the case, by evaluating the identification rate of our generated images on the training set of the real images. We set the labels of the generated images to the labels of their desired identity. If the generator were learning to do only face swapping, then a recognizer would be able to correctly identify all generated images.

However, we show that this is not the case. Neither FaceNet nor our model trained in P-NCA are able to achieve a higher recognition rate than a random guesser.

Additionally, in Fig. 5.10, we present a qualitative experiment, where the first image of each row contains the source images while the first image in each column is a randomly chosen image from the desired identities. The other images are generated. We see that the generated images take high-level characteristics of their given identities (such as race or sex) but differ greatly from the real images of those identities.

FIGURE 5.10: Generated faces of our model, where a source image is anonymized based on different identities.

5.8 Project criticism

Our method suffers from some weaknesses because it needs the original faces in the beginning to be able to anonymize them because no face can be hidden unless it is detected. It depends entirely on the discovery and identification of landmarks, and the face must be clear against the camera so that it can be changed in an excellent way. In future work, we plan to work on the whole picture and eliminate the need for landmarks in order to be able to deal with extreme situations.

Failure Cases of our model because of some extreme situation :

FIGURE 5.11: Failure Cases of CIAGAN – example 3

5.9 Conclusion

We discuss in this chapter the environment of development , the libraries of python and the dataset we used in this project, we also discuss the results we obtained after the evaluation metric we did.

6.1 Conclusion

The life of the individual is a personal matter, so we must respect it and not endanger it through the bad use of photos and videos, especially since the method of obtaining them and accessing sensitive information after the large spread of surveillance cameras became easy, so it was necessary to search for effective solutions all the time to fight this phenomenon. Through our research, we were able to design a CIAGAN model that depends on "conditional generative adversarial networks" and "Identity guidance", which ensures that the detected faces are changed to other unrecognizable realistic faces. The topic we discussed in our research is a difficult topic that constitutes a major obstacle for the owners of companies and shops who want to use surveillance cameras to maintain their security, which prompted some scientists to raise the problem, some good research appeared in the last two years but it is only the beginning and there is still a lot of work left to eliminate this problem once and for all.

6.2 Future Work

In the future, we look forward to developing our project more and including other features to make it more effective, such as changing the voice of the speaker in the videos and replacing it with a fake voice in order to give greater privacy to this person, or defining the entire body, not just the face, and changing its shape to another fake body. But what really matters in our future work is to eliminate the flaws that our model suffers from, so we will try to eliminate the need for full facial features in order to be able to deal with extreme situations in which the face is unclear.

Bibliography

[1] George Danezis, Josep Domingo-Ferrer, Marit Hansen, Jaap-Henk Hoepman, Daniel Le Metayer, Rodica Tirtea, and Stefan Schiffner. Privacy and data protection by design-from policy to engineering. *arXiv preprint arXiv:1501.03726*, 2015.

[2] Priyank Jain, Manasi Gyanchandani, and Nilay Khare. Differential privacy: its technological prescriptive using big data. *Journal of Big Data*, 5(1):1–24, 2018.

[3] Balaji Raghunathan. *The complete book of data anonymization: from planning to implementation*. CRC Press, 2013.

[4] Pascal Birnstill, Daoyuan Ren, and Jürgen Beyerer. A user study on anonymization techniques for smart video surveillance. In *2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2015.

[5] Dorothy Elizabeth Robling Denning. *Cryptography and data security*. Addison-Wesley Longman Publishing Co., Inc., 1982.

[6] Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. How does data augmentation affect privacy in machine learning? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10746–10753, 2021.

[7] Momina Masood, Marriam Nawaz, Khalid Mahmood Malik, Ali Javed, and Aun Irtaza. Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward. *arXiv preprint arXiv:2103.00484*. 2021.

[8] Stiven MORVAN, Colin TREAL, and Johan SORETTE. Methods applicable on cgan for improving performance related to image translation applications. 2019.

[9] Maxim Maximov, Ismail Elezi, and Laura Leal-Taixé. Ciagan: Conditional identity anonymization generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5447–5456, 2020.

[10] Yu Liu, Duc Nguyen, Nikos Deligiannis, Wenrui Ding, and Adrian Munteanu. Hourglass-shapenetwork based semantic segmentation for high resolution aerial imagery. *Remote Sensing*, 9:522, 05 2017.

[11] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2017.

[12] Adrian Rosebrock. Facial landmarks with dlib opencv and python-pyimagesearch. *PyImageSearch*, 2017.

[13] Davide Chicco. Siamese neural networks: An overview. *Artificial Neural Networks*, pages 73–94, 2021.

[14] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[15] Lei Shi, Xiang Xu, and Ioannis A Kakadiaris. Ssfd: A face detector using a single-scale feature map. In *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–10. IEEE, 2018.

[16] D Sangeetha and P Deepa. Efficient scale invariant human detection using histogram of oriented gradients for iot services. In *2017 30th International Conference on VLSI Design and 2017 16th International Conference on Embedded Systems (VLSID)*, pages 61–66. IEEE, 2017.

[17] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018):11, 2018.

[18] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[19] Hercules Dalianis. Evaluation metrics and evaluation. In *Clinical text mining*, pages 45–53. Springer, 2018.

[20] Walid Magdy and Gareth JF Jones. Pres: a score metric for evaluating recall-oriented information retrieval applications. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 611–618, 2010.

[21] Jason Brownlee. How to implement the frechet inception distance (fid) for evaluating gans. *Retrieved December*, 5:2019, 2019.

[22] Michael Soloveitchik, Tzvi Diskin, Efrat Morin, and Ami Wiesel. Conditional frechet inception distance. *arXiv preprint arXiv:2103.11521*, 2021.